

Proceedings of the
16th Annual Conference of the
European Association for Machine Translation
EAMT 2012

Trento | Italy, May 28th - 30th 2012



Edited by Mauro Cettolo, Marcello Federico, Lucia Specia, Andy Way

EAMT 2012

Proceedings of the 16th Annual Conference of the
European Association for Machine Translation

Trento | Italy, May 28th - 30th 2012

Edited by

Mauro Cettolo, Marcello Federico, Lucia Specia, Andy Way



Table of Contents

Foreword.....	IX
Message from the Conference Chair	XI
Message from the Programme Chairs	XIII
Committees.....	XV
Sponsors.....	XIX
<i>Invited Talk:</i>	
<i>The Unavoidable Adoption of Machine Translation</i>	<i>XXI</i>
Donald A. DePalma	

Oral Session 1 – User Papers

<i>From Subtitles to Parallel Corpora</i>	<i>3</i>
M. Fishel, Y. Georgakopoulou, S. Penkale, V. Petukhova, M. Rojc, M. Volk, A. Way	
<i>Building English-Chinese and Chinese-English MT engines for the computer software domain.....</i>	<i>7</i>
M. Khalilov, R. Choudhury	
<i>Statistical Machine Translation prototype using UN parallel documents</i>	<i>12</i>
B. Pouliquen, C. Mazenc, C. Elizalde, J. Garcia-Verdugo	
<i>User Evaluation of Interactive Machine Translation Systems</i>	<i>20</i>
V. Alabau, L. A. Leiva, D. Ortiz-Martínez, F. Casacuberta	

Oral Session 2 – Research Papers

<i>Translate, Predict or Generate: Modeling Rich Morphology in Statistical Machine Translation</i>	<i>27</i>
A. El Kholly, N. Habash	
<i>Exploiting Shared Chinese Characters in Chinese Word Segmentation Optimization for Chinese-Japanese Machine Translation.....</i>	<i>35</i>
C. Chu, T. Nakazawa, D. Kawahara, S. Kurohashi	
<i>Hebrew Morphological Preprocessing for Statistical Machine Translation</i>	<i>43</i>
N. Singh, N. Habash	

Poster Session 1 – User and Project Papers

User Papers:

<i>Building Translation Awareness in Occasional Authors: A User Case from Japan</i>	53
M. Tatsumi, A. Hartley, H. Isahara, K. Kageura, T. Okamoto, K. Shimizu	
<i>Efficiency-based evaluation of aligners for industrial applications</i>	57
A. Toral, M. Poch, P. Pecina, G. Thurmair	
<i>Evaluation of Machine-Translated User Generated Content: A pilot study based on User Ratings</i>	61
L. Mitchell, J. Roturier	
<i>A Machine Translation Toolchain for Polysynthetic Languages</i>	65
P. Homola	
<i>EASTIN-CL: A multilingual front-end to a database of Assistive Technology products</i>	69
G. Thurmair, A. Agnoletto, V. Gower, R. Rozis	
<i>Towards the Integration of MT into a LSP Translation Workflow</i>	73
D. Vilar, M. Schneider, A. Burchardt, T. Wedde	
<i>Context-Aware Machine Translation for Software Localization</i>	77
V. Muntés-Mulero, P. Paladini Adell, C. España-Bonet, L. Màrquez	

Project Papers:

<i>Virtus™: Translation for Structured Data</i>	81
<i>MOLTO - Multilingual On-Line Translation</i>	82
<i>AIDA: Automatic Identification and Glossing of Dialectal Arabic</i>	83
<i>CESAR - Central and South-East European Resources</i>	84
<i>BOLOGNA - Bologna Translation Service</i>	85

Poster Session 2 – Project Papers

<i>ACCEPT - Automated Community Content Editing PorTal</i>	89
<i>PANACEA - Platform for Automatic, Normalised Annotation and Cost-Effective Acquisition of Language Resources for Human Language Technologies</i>	90
<i>ATLAS - Automatic Translation into Sign Languages</i>	91
<i>FAUST - Feedback Analysis for User adaptive Statistical Translation</i>	92

<i>EU-BRIDGE - Bridges Across the Language Divide</i>	93
<i>GF Eclipse Plugin: an IDE for grammar development in GF.....</i>	94
<i>CrossLang Moses SMT Production System</i>	95
<i>Embedding Machine Translation in ATLAS Content Management System.....</i>	96
<i>TTC - Terminology Extraction, Translation Tools and Comparable Corpora</i>	97
<i>Confident MT - Estimating Translation Quality for Improved Statistical Machine Translation</i>	98
<i>PET: a Tool for Post-editing and Assessing Machine Translation.....</i>	99
<i>LetsMT! - Do-It-Yourself Machine Translation Factory on the Cloud</i>	100

Oral Session 3 – Research Papers

<i>Cross-lingual Sentence Compression for Subtitles.....</i>	103
W. Aziz, S. C. M. de Sousa, L. Specia	
<i>Can Automatic Post-Editing Make MT More Meaningful?</i>	111
K. Parton, N. Habash, K. McKeown, G. Iglesias, A. de Gispert	
<i>Evaluating User Preferences in Machine Translation Using Conjoint Analysis</i>	119
K. Kirchhoff, D. Capurro, A. Turner	

Poster Session 3 – Research and Project Papers

Research Papers:

<i>Cascaded Phrase-Based Statistical Machine Translation Systems.....</i>	129
D. Tufiş, S.D. Dumitrescu	
<i>Hybrid Parallel Sentence Mining from Comparable Corpora</i>	137
D. Ştefănescu, R. Ion, S. Hunsicker	
<i>Domain Adaptation of Statistical Machine Translation using Web- Crawled Resources: A Case Study.....</i>	145
P. Pecina, A. Toral, V. Papavassiliou, P. Prokopidis, J. van Genabith	
<i>Relevance Ranking for Translated Texts</i>	153
M. Turchi, J. Steinberger, L. Specia	

<i>Automatic Tune Set Generation for Machine Translation with Limited In-domain Data</i>	161
J. Chen, J. Devlin, H. Cao, R. Prasad, P. Natarajan	
<i>Domain Adaptation in SMT of User-Generated Forum Content Guided by OOV Word Reduction: Normalization and/or Supplementary Data?</i>	169
P. Banerjee, S. K. Naskar, J. Roturier, A. Way, J. van Genabith	
<i>Long-distance reordering during search for hierarchical phrase-based SMT</i>	177
F. Braune, A. Gojun, A. Fraser	
<i>Mixture-Modeling with Unsupervised Clusters for Domain Adaptation in Statistical Machine Translation</i>	185
R. Sennrich	
<i>Extending CCG-based Syntactic Constraints in Hierarchical Phrase-Based SMT</i>	193
H. Almaghout, J. Jiang, A. Way	
<i>Project Papers:</i>	
<i>MosesCore - Moses Open Source Evaluation and Support Co-ordination for OutReach and Exploitation</i>	201
<i>MateCat - Machine Translation Enhanced Computer Assisted Translation</i>	202
<i>SUMAT - An online service for SUBtitling by MACHine Translation</i>	203
<i>TransLectures - Transcription and Translation of Video Lectures</i>	204
<i>ACCURAT - Analysis and Evaluation of Comparable Corpora for Under Resourced Areas of Machine Translation</i>	205
<i>CoSyne - a Project on Multilingual Content Synchronization with Wikis</i>	206
<i>LT-Innovate - The Forum for Europe's Language Technology Industry</i>	207
<i>TOSCA-MP - Task-oriented search and content annotation for media production</i>	208
<i>Organic.Lingua - Demonstrating the Potential of a multilingual Web portal for Sustainable Agricultural & Environmental Education</i>	209

Poster Session 4 – Research Papers

<i>Flexible finite-state lexical selection for rule-based machine translation</i>	213
F. M. Tyers, F. Sánchez-Martínez, M. L. Forcada	
<i>Statistical Post-Editing of Machine Translation for Domain Adaptation</i>	221
R. Rubino, S. Huet, F. Lefèvre, G. Linarès	

<i>Crowd-based MT Evaluation for non-English Target Languages</i>	229
M. Paul, E. Sumita, L. Bentivogli, M. Federico	
<i>Readability and Translatability Judgments for “Controlled Japanese”</i>	237
A. Hartley, M. Tatsumi, H. Isahara, K. Kageura, R. Miyata	
<i>A Phrase Table without Phrases: Rank Encoding for Better Phrase Table Compression</i>	245
M. Junczys-Dowmunt	
<i>Creating Term and Lexicon Entries from Phrase Tables</i>	253
G. Thurmair, V. Aleksić	
<i>WIT³: Web Inventory of Transcribed and Translated Talks</i>	261
M. Cettolo, C. Girardi, M. Federico	
<i>A Hybrid System for Patent Translation</i>	269
R. Enache, C. España-Bonet, A. Ranta, L. Màrquez	

Oral Session 4 – Research Papers

<i>Hierarchical Sub-sentential Alignment with Anymalign</i>	279
A. Lardilleux, F. Yvon, Y. Lepage	
<i>Adjunct Alignment in Translation Data with an Application to Phrase-Based Statistical Machine Translation</i>	287
S. Arnoult, K. Sima’an	
<i>LTG vs. ITG Coverage of Cross-Lingual Verb Frame Alternations</i>	295
K. Addanki, C. Lo, M. Saers, D. Wu	

Oral Session 5 – Research Papers

<i>Learning Machine Translation from In-domain and Out-of-domain Data</i>	305
M. Turchi, C. Goutte, N. Cristianini	
<i>Discriminative Reordering Extensions for Hierarchical Phrase-Based Machine Translation</i>	313
M. Huck, S. Peitz, M. Freitag, H. Ney	
<i>Pivot-based Machine Translation between Statistical and Black Box systems</i>	321
A. Toral	

Foreword

The European Association for Machine Translation (EAMT) organised its first Workshop/ Conference back in 1996, in Austria. Up until 2009, when I became EAMT President, events had been held in Denmark, Switzerland, the Czech Republic, Slovenia, the UK, Hungary, Ireland, Malta, Norway and Germany.

When I took over, I was very keen as EAMT President to see our conferences take place in countries that we hadn't visited before. In 2009, we went to Spain, France in 2010, and last year we went to one of the Benelux countries, namely Belgium. All three events were fantastically organised, and proved to be very successful.

This year, we are continuing this trend. I am very pleased that this year we are holding the EAMT annual conference for the first time in Italy, where MT has thrived for quite some time now. I am also pleased to say that this is the first EAMT conference held since I became President of the International Association of MT (IAMT), a role I am honoured to fulfil.

This is the 16th Annual Meeting of the EAMT, which as an organisation continues to grow and thrive. The numbers of student, individual, institutional and corporate members continue to rise, partly due to improved membership packages, but also because of the range of new initiatives that the Association has recently undertaken, including the Best PhD Thesis Award, the database version of the MT Compendium, sponsoring R&D activities, an extension of our activities to the MENA region, Best Paper Award etc. Note also that since its inception in 1997, the EAMT has not raised its Membership rates, and we will continue to hold the cost of membership for 2012. Joining us really is great value, especially in a year like 2012, where more than one IAMT-affiliated event takes place (EAMT here, and AMTA later in the year in San Diego: <http://amta2012.amtaweb.org/>), especially now that with the help of the Presidents of the other regional associations, Alon Lavie and Hitoshi Isahara, we have arranged for conference discounts to benefit all IAMT members, no matter which regional association you have joined.

As last year, I would like to thank my colleagues on the EAMT Committee, who continue to provide me with invaluable support. They work tirelessly on behalf of all of us, and we are all very fortunate to have such a strong body of colleagues representing our Association.

In addition to all this, EAMT conferences continue to improve in quality, with the result that ever larger audiences have been attracted to our events, to the extent that the annual EAMT Conference is now a must for many protagonists in the field, and not just from Europe. This 16th Conference is no exception, and in particular I would like to thank my Programme Co-Chair Lucia Specia, together with the overall Conference Chair, Marcello Federico, for helping me assemble a very attractive programme, comprising of Research and User tracks, poster sessions, and a terrific Invited Speaker in Don DePalma. As in the past two years, a special session has been

organised where some prominent FP7 projects are featured, so this too will be a really interesting session.

Last but not least, I would especially like to thank our local organizers, Marcello Federico and Mauro Cettolo, who very generously volunteered to hold the meeting in Trento. We are very grateful to Marcello and his team for their excellent organization of this event.

Finally, thanks to all of you for coming. I hope you all enjoy the conference, that you benefit from the excellent programme that has been assembled, and that you go away from here having made new friends.

Andy Way
Director of Language Technology

Applied Language Solutions,
Delph, Saddleworth, UK

President of the EAMT

andy.way@appliedlanguage.com

Message from the Conference Chair

It is a great pleasure to welcome you at the Fondazione Bruno Kessler (FBK) for the 16th Conference of the European Association for Machine Translation. This is the first time that the EAMT annual conference has been organized in Italy and I'm very thankful to the Board of EAMT for giving me the opportunity to host the 2012 edition in Trento.

Given the increasing popularity of the EAMT conference over the last few years, the board of EAMT decided to organize the 2012 conference in two and a half days instead of two. This choice was indeed rewarded by an unexpectedly high number of paper submissions this year, 30% more than in 2011. Hence, I really hope you will enjoy the technical program and will find yourself comfortable with the conference venue.

The conference will be held at the technological and scientific hub of FBK, on the hill of Povo, a suburb of Trento. Morning sessions and coffee breaks will be hosted in the conference room of the main building. Lunches will be instead served in the large hall of the North building, where also the afternoon poster sessions will take place. In the same building, there will be extra rooms available for informal meetings as well as a cafeteria.

I hope you will enjoy the two social events that we have organized: the welcome reception in the Sass underground archaeological area in Trento, and the conference banquet along the lake in Riva del Garda. I hope these two occasions will give you a taste of the architectural, cultural and natural beauty of Trentino and Italy.

Nothing of this conference could have been organized without good teamwork. Hence, I wish to thank the Program co-Chairs, Lucia Specia and Andy Way, for managing the unexpectedly high number of submissions and for arranging the conference program. Thank you also to the staff at FBK, which worked with impressive professional dedication to set-up this event. In particular, I wish to express my gratitude to Mauro Cettolo, Local Organization Chair, Silvia Malesardi, venue and the social events, Francesca Guerzoni, website, Moira Osti, marketing, Luigi Massimiliano Cordisco, correspondence, Barbara Gazzoli and Adalberto Bragagna, editing of proceedings, and to our student volunteers, Prashant Mathur, Jose Camargo de Souza, and Nick Ruiz.

The organization of a conference is the sum of many important parts. Among them there are also the sponsors, which generously supported EAMT 2012. In particular, I would like to acknowledge support of the Superintendence for Cultural and Archaeological Heritage of the Province of Trento, for hosting the welcome reception, and of Springer, for sponsoring the best paper award. Last but not least, I wish to publicly thank our silver sponsor, Microsoft Translator, and our bronze sponsor Virtus.

I wish you a very successful conference and pleasant stay in Trento!

Marcello Federico, Fondazione Bruno Kessler, IT

EAMT-2012 Conference Chair

Message from the Programme Chairs

It is a great pleasure for us to welcome you to the 16th Conference of the European Association for Machine Translation (EAMT) in Trento. We have been happy to serve as programme co-chairs of a conference that has become the yearly reference conference for European machine translation developers, researchers and users, and keeps growing year by year. A sign of this growth is that the conference was extended from 2 to 2 ½ days in order to keep to the single track format – which makes EAMT events very homely for regulars and newcomers alike.

As in previous years, the conference has three main tracks: (i) a research track, where researchers report about significant research results in any aspect of machine translation and related areas, with a substantial evaluation component, (ii) a user track, where users report their experiences with machine translation in business, government, or NGOs, and (iii) a projects track to publicize EU and international projects and initiatives. We also introduced a technology showcase for product demonstrations in order to encourage participation from developers/industry. In order to encourage submissions for the user track, we changed the format of these submissions: short papers with 2-4 pages. For projects/product demonstrations, both submissions only required a 1-page abstract.

We received a record number of submissions - a total of 102 papers: 57 in the research track, 17 in the user track, and 28 project/product descriptions. Most of the latter were accepted, but were reformulated by the project participants to conform to the conference style-guide. As far as research and user papers are concerned, after double-blind review by at least three leading MT reviewers, 40 of them (54%) were accepted and found their way into the proceedings: 29 research papers (51%) – 12 for oral presentation and 17 for poster presentation – and 11 user papers (64%), 4 for oral presentation and 7 for poster presentation. Poster presenters will also have the opportunity to showcase their work in a one-minute poster booster oral session. As expected, submissions come mainly from Europe, with a large number of submissions also received from the US this year. We also received papers with authors from the Japan, Canada, Brazil, Hong Kong, India, Kazakhstan, Mexico and Singapore.

We are in debt to the members of the programme committee and to the secondary reviewers they appointed for some of their papers. As the number of papers received was even higher than usual, they had an unusually large workload: we especially thank them for their invaluable help, which most of them completed on time, which made our lives easier!

We hope that the reviewers' comments were useful and constructive and helped all authors: for those whose papers weren't accepted, by increasing their chance in a later submission somewhere else; and for those whose papers got in, to improve their manuscripts. We know we didn't give them a lot of time to do so, and we thank authors for sending their camera-ready versions on time. We hope that the resulting

selection of papers, which you have in your conference pack, truly represents the best of machine translation research, development and real-world usage.

As an opener, we will enjoy an invited talk by Don DePalma, from Common Sense Advisory, which we hope will appeal to both our research and our user audience. To close the conference, we will have a presentation by the winner of the EAMT Best Thesis Award, Abby Levenberg, completed under the supervision of Dr. Miles Osborne at the University of Edinburgh, on *Stream-based Statistical Machine Translation*.

We thank you all: authors, presenters, members of the programme committee, reviewers and secondary reviewers, and attendees, for helping us to make EAMT-2012 a success: we hope you enjoy the programme that we have prepared for you.

As these proceedings are being finalized, our job is almost finished, and the conference is now in good hands: those of the local organizers in Trento, headed by Marcello Federico. It has been great to work with them, and we send them a special thank you!

Lucia Specia, University of Sheffield, UK
Andy Way, Applied Language Solutions, UK
EAMT-2012 co-programme chairs

Committees

Conference Chair

Marcello Federico (FBK-irst, Italy)

Programme Chairs

Research Track

Lucia Specia (University of Sheffield, UK)

User Track

Andy Way (Applied Language Solutions, UK)

Local Organization Chair

Mauro Cettolo (FBK-irst, Italy)

Programme Committee

Research Track

Wilker Aziz (University of Wolverhampton, UK)

Loïc Barrault (Université du Maine, France)

Nicola Bertoldi (Fondazione Bruno Kessler, Italy)

Laurent Besacier (Université J. Fourier, France)

Alexandra Birch (University of Edinburgh, UK)

Hervé Blanchon (l'Université Pierre Mendès - Grenoble 2, France)

Ondrej Bojar (Charles University, Prague)

Ralf Brown (Carnegie Mellon University, USA)

Antal van den Bosch (Universiteit van Tilburg, Netherlands)

Bill Byrne (Cambridge University, UK)

Nicola Cancedda (Xerox Research Centre, France)

Michael Carl (IAI Saarbrücken, Germany)

Marine Carpuat (NRC Institute for Information Technology, Canada)

Helena Caseli (Universidade Federal de São Carlos, Brazil)

Marta Costa-jussà (Barcelona Media, Spain)

Jinhua Du (Xi'an University of Technology, China)

Marc Dymetman (Xerox Research Centre, France)

Andreas Eisele (European Commission, Luxembourg)

Mark Fishel (University of Zurich, Switzerland)

Declan Groves (Dublin City University, Ireland)
Barry Haddow (University of Edinburgh, UK)
Yifan He (Dublin City University, Ireland)
Jie Jiang (Applied Language Solutions, UK)
Patrik Lambert (Université du Maine, France)
David Langlois (Nancy University, France)
Alon Lavie (Carnegie Mellon University, USA)
Yanjun Ma (Dublin City University, Ireland)
Pavel Pecina (Dublin City University, Ireland)
Sergio Penkale (Applied Language Solutions, UK)
Juan Antonio Pérez-Ortiz (Universitat d'Alacant, Spain)
Daniele Pighin (Universitat Politècnica de Catalunya, Spain)
Maja Popović (DFKI, Germany)
Carlos Ramish (University of Grenoble, France)
Felipe Sánchez-Martínez (Universitat d'Alacant, Spain)
Kepa Sarasola (Euskal Herriko Unibertsitatea, Spain)
Christophe Servan (Université du Maine, France)
Khalil Sima'an (Universiteit van Amsterdam, Netherlands)
Michel Simard (National Research Council, Canada)
Sara Stymne (Linköping University, Sweden)
Jörg Tiedemann (Uppsala University, Sweden)
John Tinsley (Dublin City University, Ireland)
Jun-Ichi Tsujii (Microsoft Research Asia, China)
Marco Turchi (JRC, Italy)
Vincent Vandeghinste (Katholieke Universiteit Leuven, Belgium)
François Yvon (LIMSI/CNRS, France)

User Track

Juan Alberto Alonso (Lucy Software Ibérica, Spain)
Diego Bartolome (Tau You, Spain)
Anthony Clarke (CLS Communications AG., Switzerland)
David Clarke (WeLocalize, Ireland)
Heidi Depraetere (Cross Language, Belgium)
Mike Dillinger (Translation Optimization Partners, US)
Ray Flournoy (Adobe Systems, US)
Viggo Hansen (EAMT Executive Committee)
Manuel Herranz (PangeaMT, Spain)
Fred Hollowood (Symantec, Ireland)
Daniel Grasmick (Lucy Software, Germany)
Dorothy Kenny (Dublin City University, Ireland)
Bente Maegaard, CST (University of Copenhagen, Denmark)
Enda McDonnell (Alchemy Software, Ireland)
Nelson Ng (Ebay, US)
Sharon O'Brien (Dublin City University, Ireland)
Sergio Ortiz-Rojas (Prompsit Language Engineering, Spain)
Mirko Plitt (Autodesk, Switzerland)
Gema Ramirez Sanchez (Prompsit Language Engineering, Spain)

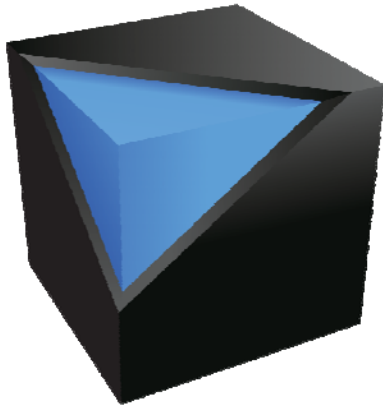
Adriane Rinsche (Language Technology Centre, UK)
Phil Ritchie (VistaTEC, Ireland)
Johann Roturier (Symantec, Ireland)
Reinhard Schaefer (University of Limerick, Ireland)
Dag Schmidtke (Microsoft Ireland, Dublin, Ireland)
Jörg Schütz (Biloom, Germany)
Svetlana Sheremetyeva (LanA Consulting ApS, Denmark)
Svetlana Sokolova (PROMT, Russia)
Gregor Thurmair (Linguatec, Germany)
Feiyu Xu (DFKI, Germany)
Elia Yuste (PangeaMT, Spain)
Ventsislav Zhechev (Autodesk, Switzerland)

Sponsors

SILVER SPONSOR

Microsoft®
Translator

BRONZE SPONSOR



VIRTUS™

Invited Talk

The Unavoidable Adoption of Machine Translation

Donald A. DePalma

Ph.D., Chief Strategy Officer & Founder of Common Sense Advisory, Inc.

There is an inevitability to machine translation that no business, government agency, or even language service provider can avoid. It's simply a matter of the huge volume of content that organizations large and small must translate to be relevant to their global constituencies. In this presentation, DePalma will review the current state of machine translation and related technologies from a business perspective, reviewing its evolution and increasing adoption among translation buyers and suppliers. He will discuss the drivers for, obstacles to, and major trends affecting the segment. He will also look at the future of machine translation and what that means for buyers and suppliers.

Oral Session 1 – User Papers

From Subtitles to Parallel Corpora

Mark Fishel,^γ Yota Georgakopoulou,^δ Sergio Penkale,^χ Volha Petukhova,^φ Matej Rojc,^ξ
Martin Volk,^γ Andy Way^χ

^γ Institute of Computational Linguistics, University of Zurich, Switzerland

^δ Deluxe Digital Studios, UK ^χ Applied Language Solutions Ltd., UK

^φ Human Speech and Language Technologies, Vicomtech, Spain

^ξ Laboratory for Digital Signal Processing, University of Maribor, Slovenia

^γ {fishel, volk}@cl.uzh.ch ^δ yota.georgakopoulou@bydeluxe.com

^χ {sergio.penkale, andy.way}@appliedlanguage.com

^φ vpetukhova@vicomtech.org ^ξ matej.rojc@uni-mb.si

Abstract

We describe the preparation of parallel corpora based on professional quality subtitles in seven European language pairs. The main focus is the effect of the processing steps on the size and quality of the final corpora.

1 Introduction

The present user study is a part of the SUMAT project,¹ which aims at developing an online machine translation (MT) service for subtitles. The project employs the paradigm of statistical MT, which means that large datasets are required for training translation models.

The training data was provided by professional subtitle companies, which create and translate subtitles for movies, TV shows and other video material; they are also the future users of the translation systems planned in the project.

In this paper we will focus on the preparation of parallel corpora on the basis of the provided data. We will describe in detail the problems that arose while producing ready, clean, usable datasets from raw subtitle files, discuss our solutions to those problems and their effect on the size and quality of the final datasets.

2 General Description

The project plans include translation between seven language pairs: English–Dutch, English–French, English–German, English–Portuguese, English–Spanish, English–Swedish and Serbian–Slovenian. Additional monolingual data was pro-

vided for language models, but in this paper we will focus on handling parallel data.

Previous work on subtitle translation (Armstrong et al., 2006; Volk et al., 2010) has demonstrated that subtitle-by-subtitle translation can be successful; there are also examples of sentence-based translation for subtitles (Tiedemann, 2009). Sentence-based translation can be linguistically motivated, but just like any other merging/splitting of the subtitles, it introduces additional pre-processing and post-processing steps, which are additional potential sources of error. In the SUMAT project we will compare the different approaches in terms of the final translation quality, but this user study is limited to subtitle-based processing only.

The subtitle companies provided the subtitle files with their original names (following a variety of naming conventions) and for the most part – in their original format. All files were accompanied by their genres and domains. Automatic processing therefore had to start with systematic file renaming, and subsequent format conversion; the following steps were language identification, document alignment, subtitle alignment and finally tokenization and lower-casing. All of these steps are described in more detail in the following sections.

3 Format Conversion

The subtitle files supplied by the subtitle companies included a text-based format, colloquially called *the .txt format* and several binary formats: STL,² 890,³ PAC⁴ and the o32/s32/x32 format group.⁵ We implemented file format converters for

²<http://www.ebu.ch>

³<http://www.cavena.se>

⁴<http://www.subtitling.com>

⁵<http://www.softelgroup.com>

Format	success rate	#files	#subs ($\cdot 10^3$)
TXT	99.6%	18 381	9 031.1
STL	99.9%	5 074	1 434.5
890	99.1%	1 469	269.2
PAC	98.2%	3 940	1 528.3
Total	99.4%	28 864	12 263.0

Table 1: Format conversion success rates in the raw dataset and the resulting number of files and subtitles after conversion.

all of them, except the latter group, which turned out to be too complicated to support without format specifications and was manually converted to .txt.

The binary formats supported text positioning, coloring, fonts, etc. Although the final translation systems have to preserve such formatting, it will be handled separately from translation. We thus discarded all formatting information in the training data and selected .txt as the common format.

The portion of the data in the .txt format thus required only encoding normalization; the main problem turned out to be the ambiguity of this format, as several different formats were grouped under a common name. All differences occurred in the subtitle time stamps: in addition to the usual index, starting and ending time, some files specified the subtitle duration (sometimes omitting the ending time), or preceded time codes with *TIMEIN* and *TIMEOUT*. A small amount of the files were missing some necessary information (e.g. only the starting time with no duration or ending time).

In contrast, the binary formats have a fixed text encoding. The main problem was caused by the formats without open specifications (890 and PAC), which have custom encoding tables for non-ASCII characters (diacritics, specified after the “carrier” letters, custom characters like non-Latin letters, copyright symbols, custom quote marks, etc.) and were reverse-engineered to implement format conversion.

Table 1 presents format frequencies in the dataset, conversion success rates and results; only 0.6% of the files were lost during this step.

4 Language Identification

Automatic language identification was required to check whether every subtitle file indeed contained subtitles in the specified language pair and to steer

document alignment.

We performed language identification using the *Lingua::Ident* package,⁶ which implements a character trigram probability-based algorithm. The *OpenSubtitles v.2* corpus (Tiedemann, 2009) was used to estimate the language signatures.

During data acquisition it turned out that some subtitles in languages unconnected with the project had ended up in the dataset, the most frequent of which were Italian and Danish; to detect such files separately, corresponding signatures were added.

After manual inspection of the language identification results, we determined that the majority of languages was identified correctly. The only small problem consisted of a couple of dozen files with gibberish or unconventional content (like “as-dfasfd”, “qwertyqwerty”, “whoop whoop! shh-huff! ding dong!”) and empty files.

The results of language identification against the manually specified languages or language pairs are presented in Table 2. Comparing the number of subtitles in the correctly placed files to the conversion results, the total subtitle loss at this point is around 95 000 subtitles, or 0.8% percent of the converted subtitles. However, given the different number of files in the two languages of every language pair, further loss is going to be greater.

5 Document Alignment

The next step was to identify pairs of subtitle files (documents) that were translations of each other. The fastest way to perform document alignment is based on the file names, since this does not involve reading the contents of the files. For that we collected and documented the file naming conventions in the dataset, discovering the following patterns:

- file names of the aligned pair differing only in the language (e.g. “*Movie_Title_en.txt*” and “*Movie_Title_fr.txt*”)
- file names starting with the same 4-to-5-digit ID (e.g. “**12345**.en.txt” and “**12345**nl6.txt”)
- file names containing the same 9-symbol ID (digits and capital letters), followed by a 3-character language code (e.g. “*Deutsche Titel-AXGM0102A_DEU.PAC*” and “*English Title-AXGM0102A_ENG.PAC*”)

⁶<http://search.cpan.org/~mpiotr/Lingua-Ident-1.7/>

Manually Specified	Automatically Identified	#files	#subs ($\cdot 10^3$)	Manually Specified	Automatically Identified	#files	#subs ($\cdot 10^3$)
English–Dutch	English	1 606	863.0	English–Spanish	English	1 694	849.0
	Dutch	1 617	833.9		Spanish	1 711	851.3
	Other	8	3.5		Other	6	2.3
English–French	English	2 369	1 066.4	English–Swedish	English	1 100	636.5
	French	2 376	1 067.7		Swedish	1 157	635.5
	Other	20	7.2		Other	10	5.6
English–German	English	6 919	1 958.7	Serbian–Slovenian	Serbian	402	233.7
	German	5 124	1 884.7		Slovenian	391	175.1
	Other	14	2.9		Other	2	1.5
English–Portuguese	English	1 145	560.3	Total	Correct-1	15 235	6 167.5
	Portuguese	1 142	552.4		Correct-2	13 518	6 000.6
	Other	4	1.7		Other	64	24.8

Table 2: Results of automatic language identification, contrasted with manually specified language pairs; the “Other” languages do not include Italian and Danish, as these are not covered in the SUMAT project.

- 8-symbol file names starting with the same movie ID (4 letters) and a 2-character language code (e.g. “MISSENDC.txt” and “MISSNLDV.TXT”)

Even while comparing file names, it is inefficient to try to align a document to all other documents, so we trimmed the search space by comparing only files within the same genre and domain.

After the initial file name-based processing, 52.1% of the subtitle files specified as parallel were identified as such. We processed the remaining files with a time code similarity-based approach to document alignment: two documents are considered parallel if at least 90% of the time codes correspond to each other.⁷

As a result of joint file name- and subtitle-based processing, we discovered alignments for 68.6% of the documents. We processed the remaining third of the dataset manually, which resulted in detected file pairs for 83% of all the files specified as parallel; the remaining 17% were added to the corresponding monolingual datasets.

The resulting numbers of aligned document pairs and subtitles are summarized in Table 3; the coverage of document alignment in terms of subtitles is 87.9% of the converted parallel dataset.

Manual reviewing of the unaligned files, initially specified as parallel, revealed that a large amount of the files were missing their counterpart.

Another problem with document alignment arose from subtitle files, which were translated and saved in parts, indicating a many-to-one document correspondence; these occurred in the English–German language pair. As a result only the first (English) part of the translation was aligned with

the full (German) document, putting the other parts into the monolingual datasets. This reflects negatively on the number of subtitles in this language pair after document alignment.

Language pair	#file pairs	#subs ($\cdot 10^3$)
English–Dutch	1 530	831.9 / 801.2
English–French	2 232	989.4 / 989.5
English–German	4 009	1 337.3 / 1 520.2
English–Portuguese	1 126	544.8 / 547.0
English–Spanish	1 641	810.9 / 811.9
English–Swedish	1 055	609.1 / 594.3
Serbian–Slovenian	380	219.1 / 169.7
Total	11 973	5 342.6 / 5 433.9

Table 3: Document alignment results: the number of file pairs and subtitles per language pair.

6 Subtitle Alignment

The main state-of-the-art work on subtitle alignment (Tiedemann, 2007, 2009) aligned corpora at the sentence level, so we had to come up with an approach of our own to align subtitles.

The main assumption in the planning phase of the SUMAT project was that almost all translated subtitles would have directly matching time codes, which would make subtitle alignment trivial. It turned out, however, that several issues made this task more “interesting”: some companies translate subtitles without preserving the time code template, which results in more loose translations and many-to-one correspondences between subtitles. Also due to a different movie cut or version, portions of the translated subtitles can be missing and

⁷see the next section on subtitle alignment for more details

Language pair	#file pairs	#sub pairs ($\cdot 10^3$)	#tokens ($\cdot 10^6$)
English–Dutch	1 515	688.7	6.89 / 5.75
English–French	2 202	944.1	9.33 / 8.72
English–German	3 841	954.9	9.20 / 8.01
English–Portuguese	1 123	523.4	5.16 / 4.60
English–Spanish	1 613	779.5	7.59 / 6.83
English–Swedish	1 047	577.5	5.87 / 4.86
Serbian–Slovenian	380	111.9	1.25 / 1.50
Total	11 721	4 580.0	45.29 / 40.27

Table 4: Subtitle alignment results: the number of aligned file pairs, subtitle pairs and tokens per language pair in the final corpora.

subsequent portions shifted.

To account for these complications, we designed a dynamic programming algorithm, based on subtitle shift similarity: subsequent subtitle alignments with a certain shift are endorsed if the shift stays almost constant. The same algorithm checks for many-to-one matches; merging is achieved by using the starting time code of one subtitle and the ending time code of a subsequent subtitle.

To assess the quality of the alignments, we aligned small held-out datasets of approximately 500 parallel subtitles per language pair manually. The average precision and recall of the alignments were 0.94 and 0.91, respectively.

As a final step we tokenized the aligned subtitles and converted them to lower-case. Serbian and Slovenian data was tokenized with a tool from the PLATTOS system (Rojc and Kacic, 2007) and the remaining data with the Moses toolkit⁸ tokenizer.

The resulting sizes of the final parallel corpora are presented in Table 4. According to the numbers the final corpora constitute a total of 85.0% of the document-aligned dataset and 74.7% of the unaligned, converted dataset. However, this estimate is overly pessimistic, since many subtitles were merged as a result of 1-to-N subtitle alignment. Data loss rates per language pair range from over 50% (German, Serbian) to 5% (Portuguese), although these estimates are exaggerated as well; it is important to note that the different rates per language are caused by the characteristics of the supplied subtitles, and not the language itself.

7 Conclusions

The SUMAT project has started by turning raw subtitle files into clean parallel corpora, usable for

training statistical translation models. We have described the problems that were encountered during the preparation of the files as well as our solutions.

The total data loss from raw subtitle files to final parallel corpora is below 25% and the corpus sizes are mostly sufficient for training translation models.

The main reason for data loss is human error, manifesting as incorrectly specified subtitle language pairs and file format inconsistencies. Added to this, the subtitle alignment algorithm was unable to fully cope with loose translations and subtitle time correspondences.

The next step in the project is training the baseline MT systems for all translation directions, thus evaluating the collected datasets in practice.

References

- Armstrong, S., C. Caffrey, M. Flanagan, D. Kenny, M. O’Hagan, and A. Way. 2006. Leading by example: Automatic translation of subtitles via EBMT. *Perspectives*, 14(3):163–184.
- Rojc, M. and Z. Kacic. 2007. Time and space-efficient architecture for a corpus-based text-to-speech synthesis system. *Speech Communication*, 49(3):230–249.
- Tiedemann, J. 2007. Improved sentence alignment for movie subtitles. In *Proceedings of RANLP-07*, pages 582–288, Borovets, Bulgaria.
- Tiedemann, J. 2009. News from OPUS – a collection of multilingual parallel corpora with tools and interfaces. In *Proceedings of RANLP-09*, pages 237–248, Borovets, Bulgaria.
- Volk, M., R. Sennrich, C. Hardmeier, and F. Tidström. 2010. Machine translation of TV subtitles for large scale production. In *Proceedings of the 2nd Joint EM+/CNGL Workshop on “Bringing MT to the User”*, pages 53–62, Denver, CO.

⁸<http://www.statmt.org/moses>

Building English-Chinese and Chinese-English MT engines for the computer software domain

Maxim Khalilov

TAUS Labs
Oudeschans 85-3
Amsterdam, 1011KW
The Netherlands
maxim@tauslabs.com

Rahzeb Choudhury

TAUS
Oudeschans 85-3
Amsterdam, 1011KW
The Netherlands
rahzeb@translationautomation.com

Abstract

In this paper we present two sets of English-Chinese and Chinese-English machine translation trials conducted by TAUS Labs on computer software content. The goal of this study is twofold: (1) to share our experience on training and optimizing of Moses-based engines driven by translation memories provided by industrial users and (2) to give to the users the idea of results, cost and effort associated with training of MT engines.

1 Introduction: goals and approach

We describe a series of English-Chinese and Chinese-English machine translation trials conducted by TAUS Labs¹ on computer software content. Statistical MT engines were trained and tested on the basis of open-source software using Amazon Elastic Cloud² as a remote server. Parallel corpora were downloaded from the TAUS Data Association repository³.

In this study we focused on the following particular questions that MT users are interested in:

- How well do statistical customizable MT engines based on Moses perform in comparison with Google Translate?
- Which Chinese word segmentation and re-ordering strategies improve translation performance?
- How expensive (in terms of time and money) is the process of MT engine training?

- How well do the automatic evaluation metrics BLEU, TER and GTM correlate with each other? What are the results of human evaluation?

While in the majority of experiments published in academic conferences tend to use only free publicly available corpora to feed MT engines, we trained our systems on the data provided by 10 industrial publishers.

2 Data

Experiments were conducted using different variations of the Chinese-English training corpus, built on a basis of translation memories coming from the software industry. Test and development datasets were provided by EMC⁴.

The training dataset contains around 22 million words on the English side and around 23 millions on the Chinese side. The development set was 500 lines long (7,000 on the Chinese side), while translation systems were tested on the corpus of around 15,000 words.

3 Baseline and experiments

The SMT system used in the experiments was implemented within the open-source MOSES toolkit (Koehn et al., 2007). Training and tuning procedures are detailed on the MOSES web page⁵.

Word alignment was estimated with GIZA++ tool⁶ (Och, 2003), coupled with mkcls⁷ (Och, 1999), which allows for statistical word clustering for better generalization. A 3-gram target lan-

©2012 European Association for Machine Translation.

¹<http://www.tauslabs.com>

²<http://aws.amazon.com/ec2>

³<http://www.tausdata.org>

⁴<http://www.emc.com/>

⁵<http://www.statmt.org/moses/>

⁶code.google.com/p/giza-pp/

⁷<http://www.fjoch.com/mkcls.html>

guage model was estimated using the SRI LM toolkit (Stolcke, 2002).

In the writing system of Chinese, texts are not segmented by words, but Moses operates with words (tokens) rather than with unbroken strings. We used two alternative segmenters for Chinese in the pre-processing step: the Stanford Chinese segmenter⁸ (Tseng et al., 2005) and the Simplified Chinese segmenter (the Peterson segmenter⁹) with the goal to determine which segmentation strategy leads to better MT system performance.

Two reordering methods are widely used along with Moses-based MT systems:

Distance-based reordering (Koehn et al., 2003): a simple distance-based reordering model default for Moses system.

MSD (Tillman, 2004): a lexicalized data-driven reordering model. The MSD model is used together with a distance-based reordering.

4 Evaluation methodology

Automatic evaluation. In English-Chinese experiments, Chinese reference translation was pre-segmented using one of the two segmentation tools (the Peterson or the Stanford segmenter) in order to make the evaluation as fair as possible. The reason for that was an intention to minimize the segmentation effect for Chinese portion of the data and focus the evaluation on the correctness of lexical choice and word order.

In Chinese-English trials, all the automatically generated translation hypotheses and reference translation were detokenized using *detokenizer.pl* script distributed as a Moses package.

We used three evaluation metrics to measure translation quality in a resource-light way:

- GTM (Turian et al., 2003), a precision-recall metric measuring similarity between MT output and reference translation. It takes into account the number of adjacent words shared by translation hypothesis and reference.
- TER (Snover et al., 2006), a metric based on the counting transformations required to reconstruct the reference translation from the MT output, while preserving the content of the source. TER estimates the number of edits

⁸<http://nlp.stanford.edu/software/segmenter.shtml>

⁹<http://www.mandarintools.com/segmenter.html>

required to change a system output into one of the references.

- BLEU (Papineni et al., 2002), a simple evaluation metric that performs better on capturing fluency rather than adequacy of the translation. BLEU shows how many words are shared between MT output and human-made reference, benefiting sequential words.

BLEU is still a de-facto standard evaluation tool for academic research on MT, despite its obvious disadvantages. BLEU tends to give a very high score with a short output, so long as all its n-grams are present as a reference. Besides, BLEU is mostly a precision metric, taking recall into account in a very simple way by considering only the measure for sentence length.

While BLEU is criticized within academic and industrial MT communities because in many cases it does not show good correlation with human judgment (Callison-Burch, 2006), GTM is reported to be a more reliable way to measure translation quality, at least for certain domains (O'Brien, 2011). Due to this reason and since it has the strong correlation with post-editing effort (Tatsumi, 2009) GTM was selected as the primary indicator of translation quality.

TER is currently considered more reliable metric than BLEU for some of the most popular translation applications since it gives a better indication of the post-editing effort compared to BLEU (O'Brien, 2011).

A comparison with free online engine was completed for informative purposes. Since Google is not a member of TAUS Data Association, it does not have access to the parallel corpus that was used to train the Moses systems.

The evaluation conditions for English were case-sensitive and included punctuation marks. The Chinese translation generated by Google Translate was re-segmented to preserve the consistency of evaluation.

Human evaluation. The native speaker evaluator was provided with the original text in source language and the outputs of the four translation systems. They were asked to assess the quality of 100 lines from the test corpus using the following unique scale to measure the acceptability of the output at the segment level.

Using the methodology described in Roturier (2009) and Specia (2011) we apply a 4-level scale to measure output acceptability:

- Excellent (E): no post-editing required;
- Good (G): only minor post-editing is required;
- Medium (M): significant post-editing is required;
- Poor (P): it would be better to manually retranslate from scratch (post-editing is not worthwhile).

We also used the aggregated score following a simple approach to assign a certain weight to each category, multiply the number of occurrence by those weights, sum them up and normalize:

$$K = \frac{\sum_{i \in P, M, G, E} w_i * N_i}{N} \quad (1)$$

where $N = \sum_{i \in P, M, G, E} N_i$, $w_P = 1$, $w_M = 2$, $w_G = 3$ and $w_E = 4$.

5 Results

5.1 Automatic scores

We contrast the results shown by 4 translation systems per direction with the performance delivered by Google Translate (Table 1 and Figure 1).

5.2 Human evaluation

Some of the systems under consideration were analyzed by human judges following the strategy described in Section 4. Early results can be found in Table 2.

SID	Segment.	Reord.	GTM	TER	BLEU
Chinese-English					
1	Peters.	MSD	67.95	36.51	49.41
2	Peters.	Dist.	67.22	37.81	48.46
3	Stanf.	MSD	64.99	40.32	45.16
4	Stanf.	Dist.	64.32	40.55	44.52
G	Google	N/A	62.95	63.40	24.78
English-Chinese					
1	Peters.	MSD	76.75	39.35	36.51
2	Peters.	Dist.	76.63	39.62	34.29
3	Stanf.	Dist.	76.57	40.95	32.44
4	Stanf.	MSD	76.54	40.82	33.69
G	Google	N/A	60.81	56.99	9.40

Table 1: Automatic scores.

SID	Segment.	Reord.	P	M	G	E	K
Chinese-English							
1	Peters.	MSD	11	18	43	28	2.88
4	Stanf.	Dist.	11	17	46	26	2.87
G	Google	N/A	13	26	40	21	2.69
English-Chinese							
1	Peters.	MSD	65	10	10	11	1.59
4	Stanf.	MSD	66	12	10	11	1.64
G	Google	N/A	64	17	11	8	1.63

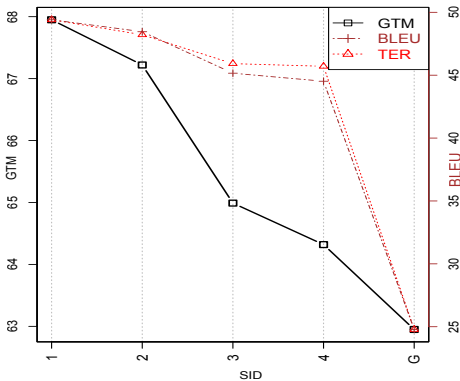
Table 2: Human evaluation results.

5.3 Correlation of automatic scores

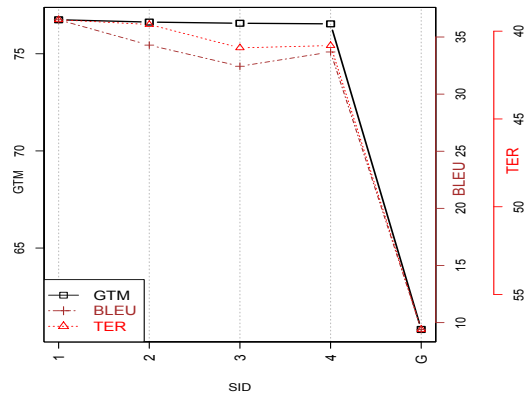
The experiments gave us an opportunity to check how well GTM, TER and BLEU correlate for a single-reference evaluation task.

Trial	BLEU-TER	BLEU-GTM	TER-GTM
Ch.-En.	-0.84	-0.60	0.66
En.-Ch.	-0.99	-0.98	0.45

Table 3: BLEU, TER and GTM correlation.



(a) Chinese-English experiments.



(b) English-Chinese experiments.

Figure 1: GTM, TER and BLEU scores.

The Pearson correlation coefficient shows strong dependence between BLEU and TER and, to a lesser extent, between BLEU and GTM metrics, which is significantly stronger for Chinese-English translation.

The results of manual judgement for Chinese-English confirm the results of automatic evaluation in grosso mode. However, there is a significant discrepancy in BLEU/TER/GTM scores and human results for English-Chinese: while according to the automatic scores Moses translations are much better than Google Translate, human evaluation shows that customized Moses and Google Translate perform virtually indistinguishable from each other. We explain it by an effect of non-ideal target-side segmentation that affects automatic scores, but is disregarded by the evaluator¹⁰.

6 Cost and effort

Data processing has been done on a local machine (regular laptop). We assume that the cost associated with its usage is virtually zero.

Cost associated with AWS usage: the total technology costs for these trials were around 28 euros per direction (4 MT engines, 2 different datasets).

Human and time resources: data preparation was done in parallel for both translation directions. While the most time-consuming part, which is training corpus processing, was shared for both trials, development and test corpora were cleaned, tokenized and segmented independently. The data preparation process took around 16 hours.

MT engine training, system optimization, and backing-up amounted to 60 hours, equally distributed between 2 master engines. Around 30% (18 hours) of that lapsed time required human resources (mostly, on the data preparation step).

7 Findings and future work

We operate with an open-source Moses toolkit that implements the entirely data-driven approach to MT. The corpus-based nature of this software implies high dependence on parallel data that is fed to the MT engine.

- Unsurprisingly, we find that in domain texts are translated much better by Moses MT

solutions trained on specific material than the high-quality, but general-purpose Google Translate tool. The best Moses-based system performs two times better than Google Translate in terms of BLEU score (+7% in terms of GMT) for Chinese-English translation. Moses-based solutions outperform the online Google solution by almost four times (BLEU) and +20 % (GMT) when translating from English into Chinese.

- Access to the right data, which is a core element of MT customization is the key aspect in getting competitive translation performance. This should be taken into account by decision makers when adopting or integrating MT.
- The choice of word segmentation strategy for Chinese can have significant impact on the delivered translation. Segmentation of Chinese portion of parallel corpora (training, development and test) with the use of a rather simple, but efficient Peterson segmenter leads to a better performance than segmentation done using Stanford segmenter based on pattern recognition algorithm.
- Notable finding of this study is that some of the evaluation metrics based on different principles are well correlated. BLEU (the metric that estimates the number of n-grams shared by translation hypothesis and human reference) and TER (the metric based on counting of number of text transformations) report high correlation for both directions ($|r|=0.84$ for English-Chinese and $|r|=-0.99$ for Chinese-English). GTM is well correlated with BLEU. Correlation for English-Chinese translation is much weaker than for Chinese-English.
- The results of human evaluation confirm the scores shown by automatic metrics for Chinese-English trials, but do not verify the huge degradation in Google Translate performance shown by BLEU, TER and GTM scores for English-Chinese.

References

Koehn, Ph., F. Och, and D. Marcu. 2003. Statistical phrase-based machine translation. In *Proceedings of the HLT-NAACL'03*, pages 48–54.

¹⁰A calculation of r correlation between automatic scores and human evaluation results is not presented in this paper due to a low number of manually evaluated systems.

- Koehn, Ph., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: open-source toolkit for statistical machine translation. In *Proceedings of ACL'07*, pages 177–180, Prague, Czech Republic.
- O'Brien, Sh. 2011. Towards predicting post-editing productivity. *Machine Translation*, 25(3):197–215.
- Och, F. 1999. An efficient method for determining bilingual word classes. In *Proceedings of ACL'99*, pages 71–76, Maryland, MD, USA.
- Och, F. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of ACL'03*, pages 160–167, Sapporo, Japan.
- Papineni, K., S. Roukos, T. Ward, and W. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL'02*, pages 311–318, Philadelphia, PA, USA.
- Roturier, J. 2009. Deploying novel mt technology to raise the bar for quality: a review of key advantages and challenges. In *Proceedings of the MT Summit XII*, pages 1–8, Ottawa, Canada, August.
- Snover, M., B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, pages 223–231, Cambridge, Massachusetts, USA, August.
- Specia, L. 2011. Exploiting objective annotations for measuring translation post-editing effort. In *Proceedings of EAMT 2011*.
- Stolcke, A. 2002. SRILM: an extensible language modeling toolkit. In *Proceedings of SLP'02*, pages 901–904.
- Tatsumi, M. 2009. Correlation between automatic evaluation scores, post-editing speed and some other factors. In *Proceedings of MT Summit XII*, pages 332–339, Ottawa, Canada, August.
- Tillman, C. 2004. A unigram orientation model for statistical machine translation. In *Proceedings of HLT-NAACL'04*, pages 101–104, Boston, MA, USA.
- Tseng, H., P. Chang, G. Andrew, D. Jurafsky, and Ch. Manning. 2005. A conditional random field word segmenter. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*.
- Turian, J., L. Shen, and I.D. Melamed. 2003. Evaluation of machine translation and its evaluation. In *Proceedings of the MT Summit IX*, pages 386–393, New Orleans, USA, September.

Statistical Machine Translation prototype using UN parallel documents

Bruno Pouliquen, Christophe Mazenc
World Intellectual Property Organization
Global Databases Service

34, chemin des Colombettes
CH-1211 Geneva 20

Bruno.Pouliquen@wipo.int,
Christophe.Mazenc@wipo.int

Cecilia Elizalde, Jose Garcia-Verdugo
United Nations

Department for General Assembly
and Conference Management
Documentation Division, Spanish Translation Service
405 42nd Street
New York NY 10017

elizalde@un.org,
garcia-verdugo@un.org

Abstract

This paper presents a machine translation prototype developed with the United Nations (UN) corpus for automatic translation of UN documents from English to Spanish. The tool is based on open source Moses technology and has been developed by the World Intellectual Property Organization (WIPO). The two organizations pooled resources to create a model trained on an extensive corpus of manually translated UN documents. The performance of the SMT system as a translation assistant was shown to be very satisfactory (using both automatic and human evaluation). The use of the system in production within UN is now under discussion

1 Introduction

This paper describes a prototype for the automatic translation of United Nations documents¹.

The tool has been the subject of experiments within the United Nations, including a structured human evaluation carried out by three professional translators.

The number of documents translated by this UN Division per year is 33,670 (90 million words) in the six official languages (Arabic, Chinese, English, French, Russian and Spanish).

The UN has an extensive parallel corpus of high-quality human translations collected from 2000 to

2011 for all language combinations, since the norm is that parliamentary documents are to be translated to all the six official languages and issued simultaneously.

Quality is a paramount consideration for the translation of parliamentary documents at the UN: translators are highly skilled professionals, 50% of the translations are revised by a senior reviser, 50% are subject to self-revision, afterwards 80% of translations are subject to additional proofreading or scoping².

Due to the growing demand for translations and budgetary considerations, the percentage of contractual translation (currently 20%) is bound to increase. However, contractual translators do not have access to the same document and terminology databases and IT tools as internal staff, and therefore the quality of their translations suffers.

The UN documents submitted for translation in New York deal with a great diversity of subjects, including 10%-15% of documents relating to budgetary and administrative issues that are good candidates for computer-assisted translation because they contain around 30% of repetitive language.

UN translators have been exposed to machine translation through Google Translate (either directly or through CAT tools) and have found that the output quality, for the purposes of the translation of UN documents, has been decreasing over the years as documents from other organizations were added³. Their expectation is to explore the possibilities of a SMT tool trained only with UN documents. There is also the expectation to im-

© 2012 European Association for Machine Translation.

¹ Documents provided by the Documentation Division (New York) of the Department for General Assembly and Conference Management, the main entity of the United Nations Secretariat charged with the production of parliamentary documentation. The Documentation Division in New York deals with the translation of parliamentary documentation.

² A lighter proofreading where only numbers, titles and number of paragraphs are checked.

³ Google used UN documents to train its MT tool, <http://www.reuters.com/article/2007/03/28/us-google-translate-idUSN1921881520070328>

prove the quality and consistency of the contractual translation by providing contractors with the same toolkit as internal staff and/or applying MT and post-editing.

In parallel, WIPO has already developed such a SMT with a similar-sized corpus (called WIPO-COPPA⁴, see Pouliquen & Mazenc, 2011)

A preliminary test was launched using the WIPO tool (described in Pouliquen et al, 2011) in which: a statistical machine translation (SMT) system was trained using the UN corpus in order to evaluate the quality of such a tool (especially in comparison with other tools).

UN Spanish Translation Service (STS) has been exposed to MT (mainly from English to Spanish), rule-based systems required too much work to adapt their own terminology while SMT-based systems like Google/Bing/Language Weaver yield good results. It was decided to launch an experiment with this language pair. UN STS gave WIPO access to their 64,619 English-Spanish documents.

2 Context/State of the art

At the UN, due to the large volume of translated pages and recent budgetary restrictions, there is a growing demand to decrease costs and increase throughput by leveraging IT tools as applied to translation and to increase quality and consistency, in particular for the jobs translated by contractual translators. Most in-house translators type or dictate their translations and look for information in monolingual and bilingual document databases, and terminology databases. At the Spanish Translation Service, around 25% of the documents are prepared using CAT tools⁵. Translators have been exposed to SMT and have expressed interest in including this technology in their regular toolkit.

Various techniques can be used in Machine Translation (Koehn 2010): rule-based systems, example-based translation, statistical machine translation and hybrid systems.

⁴ English-French Patent corpus of 170 Million words Freely available for research purpose at <http://www.wipo.int/patentscope/en/data/products.html#coppa>

⁵ UN translators use mainly SDL products with file-based translation memories. The UN is currently developing its own web-based computer-assisted translation, referencing and terminology tool in the context of a global project called *gText*, using internal developers

An international organization like the UN has 6 working languages (plus German), which means that, if such an organization wanted a translation tool in all language pair combinations, it would require 42 translation engines. A rule-based translation system would be extremely costly to build and maintain. A data-driven approach is usually more suitable when a big parallel corpus exists.

Some UN parallel corpora are already available on the Web: UN Corpora⁶ (Rafalovitch et al. 2009) provides a 3.5 million word corpus which contains only a part of the General Assembly Resolutions for eight sessions only and has not been updated. Multi UN⁷ (Eisele et al. 2010) has built a more extensive corpus of 370 Million words however this corpus is now outdated (up to September 2010) and not sentence-aligned.

In December 2011, the validity of a 1994 agreement with LDC was reconfirmed. The Linguistic Data Consortium (see Graff 1994) will make an updated UN corpus available for research purposes.

For the purpose of the current experiment, the Spanish Translation Service (STS) provided its full collection of English-Spanish bitexts from 2000 to 2011, composed of 64,619 documents (equivalent to about 220 million words).

With this high-quality parallel corpus, SMT was chosen, with a flexible and free engine: *Moses* (Koehn et al. 2007).

2.1 The English-Spanish parallel corpus

The SMT is trained with a parallel corpus extracted from previously translated UN documents from 2000 up to December 2011 (62,757 English-Spanish documents after filtering⁸). The provided corpus is extracted from HTML bitext files⁹. We chose to re-align every text as WIPO's

⁶ <http://www.uncorpora.org/>

⁷ <http://www.euromatrixplus.net/multi-un/>

⁸ The documents all originated from UN headquarters (New York), more documents can be included in the future from UN-Geneva and UN-Vienna. We filtered out documents not in the right language or having an unrecognized format.

⁹ UN document division has a simple script that matches pairs of documents with the selected language pair and a commercial alignment robot that generates the corresponding HTML table. The robot alignment algorithm relies heavily on document formatting (Microsoft Word 97/2000 format) and automatically discards document pairs that exceed a specific misalignment threshold. The resulting bitexts contain a significant amount of misaligned segment,

aligner is tailored for machine translation and produces cleaner alignments.

Starting with this material, we tried to build a reasonably clean bilingual corpus by applying the following steps (some of the cleaning techniques were successful in previous WIPO experiments):

- carrying out sentence splitting of documents (using a home-made splitter, based on sentence boundaries and a list of abbreviations)
- tokenizing each sentence (using a home-made tokenizer based on *Lucene* framework¹⁰)
- using *Champollion* (Ma 2006) to align sentences, we developed a Java version which allows to split further long sentences (having more than 80 words). The tool uses a bilingual dictionary which we extract from previously extracted model.
- computing an “aligned-segment-matching-score” for each aligned segment (taking into account a previously learned bilingual dictionary)
- filtering out whole documents having an average-segment-matching-score below a given threshold (empirically set to 0.15)
- applying a smooth filter on the segment-matching-score ([0.1,0.2,0.4,0.2,0.1]) which will “propagate” the score of a segment to the adjacent ones, filtering out the segments having a “smooth” score lower than a second threshold (empirically set to 0.3)
- filtering out sentences having more than 80 words¹¹ (or only one word)
- filtering out pairs of sentences where the ratio (number of English words/number of Spanish words is more than 9)
- applying some regular expression replacement rules (deleting xml tags, uniform accents, etc.)

As a result 10,251,816 aligned pairs of segments were obtained (210 Million words in English, 240 Million in Spanish). The quality of alignment is reasonable, however attempts should always be made to improve the quality in the future.

as well as up to 30% of segments containing no text at all (mainly figures, formatting elements and symbols).

¹⁰ We used the standard tokenizer (McCandless, 2010) and updated it so that it recognizes email addresses, internet hostnames, URLs, XML tags, references, Greek letters, apostrophes, etc.

¹¹ This filter is perhaps too aggressive but the word alignment speed (and quality) will usually be poor on big sentences.

2.2 Training the model

Moses can be trained using our parallel corpus.

2,000 segments were retained as our development set in order to carry out the optimization, (see section 3.3). As the documents are big, only two documents were part of the development set, it was decided to keep these two documents apart for the training. A first test set was selected as a random selection of 1000 segments out of remaining segments of these two documents.

Mgiza++ (Gao & Vogel, 2008) was used to align words in sentences. On a *Linux* server (48 cores of 2.5Ghz) it ran for 2.5 days on the corpus. We then stored this information in a *Lucene* index so we can use it for our concordancer (see section 3.2) or as a translation memory (TM) index.

The language model is built using SRILM toolkit (Stolke, 2002) with 5-grams. The model is generated out of the Spanish texts of the corpus (239,424,105 words).

2.3 Optimization

Attempts to optimize the performance of the system with various settings were carried out:

The generated phrase table contains 272 million entries, in such a huge table, some phrases are very unlikely to be seen in other documents. A decision was taken to try to “prune” this phrase table (in such a big corpus a phrase that occurs only once has a high probability of being useless and even erroneous). The “pruning” method as described in Johnson et al. (2007) was used with the suggested parameters i.e.: delete all phrases which occur only once in our training corpus and, for each phrase, only the first 30 translation candidates are kept. The “pruned” phrase table now contained 50 million entries (19% of the original). The speed of the translation improved and, as expected, the quality improved as well (see line “pruned” in Table 1).

The reordering model (originally containing 272 million entries) was also filtered using two criteria: the source and target ngrams were kept only if they appear in the “pruned” phrase table (resulting in 50 million entries only) and only if source and target contained less than 9 words in total (resulting in 27 million entries, this last criterion is arguable, however the differences in BLEU/METEOR scores with and without this filter are negligible while the size of the reordering model is considerably reduced, see the differ-

ences between line “pruned” and “prunedmax4” in table 1).

An optimization of the settings by maximizing the BLEU score on the development set (2000 segments) was carried out using the minimum error rate training (MERT).

2.4 Preliminary results

A very first BLEU score (see section 5.1) of 65.45 was quite encouraging but not reliable as it was computed on a non-representative test set (remaining sentences of the development set).

3 Translating / graphical user interface

3.1 Server configuration

We chose to set up an architecture that allows:

- various users to work at the same time
- various alternative translations
- word alignment

The *Moses* decoder is slightly modified in order to output the first 24 proposals for each submitted translation. Each decoder is encapsulated in a Java *RMI* interface server which allows the running of several concurrent decoders (on the same or on different language pairs). Each sentence submitted is queued and sent to the next free decoder.

Our phrase tables are so big (even after the “pruning”) that it is impossible to store them in memory, we store them on disk, even so the server gives good performance. The phrase tables keep the word alignment information so that users can highlight translated words in a sentence. The server includes a *post-processor* that deletes unnecessary spaces and recases the output taking into account the input (functionality to be improved in the future).

3.2 Graphical user interface

We set up a Java Server Faces¹² Web interface to connect to the translation server. Users can interactively get translations of new documents. Alternatively they can also verify the segment previously translated through a concordancer. The interfaces were designed to be intuitive. We used *Moses*’ “keep alignment” functionality so that word translations are highlighted (as well as parallel segments), so that the user can immediately spot the good/bad translations.

3.2.1 First Web interface: gist translation

A first Web interface allows user to submit short texts and access the corresponding automatic translation (with highlighting of parallel segments/words).

Users can access alternative translation proposals by clicking on a given segment.

However, *Moses*’ mechanism limits the proposal alternatives when the sentence is too long, in such cases, the user can select a chunk of source text and gets alternatives for this new segment.

A small icon indicates to the user that a segment has already been translated (is part of the TM) and links him to the concordancer (see 3.2.3)

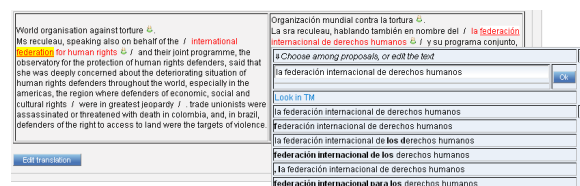


Figure 1: Gist translation, highlighting parallel segments and words (here *federación/federation*), user can access alternative translations for a given segment, the green icons are an indicator that the segment is part of the TM.

This web interface can be tested on the WIPO website, but is only suitable for Patent texts: <http://www.wipo.int/patentscope/translate>

3.2.2 Second Web interface: interactive translation

An alternative graphical user interface lets the user segment the text to be translated. With this interface the translator drives the translation by providing the segments he wants to translate. He can then immediately select alternative proposals.

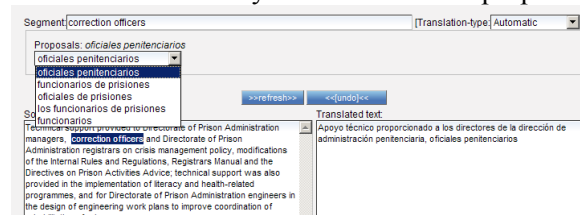


Figure 2: Interactive translation interface, user highlights the next source segment to translate and can select alternative translations

3.2.3 Concordancer

Users can access the concordancer using a Web interface. The concordancer is based on a *Lucene* index containing the information result of the word alignment (using *grow-diag-final-and file* – (Koehn 2010 p. 118)). This concordancer displays the segments containing the search term

¹² <http://javaserverfaces.java.net/>

and the corresponding aligned words. A first window displays the usage of the term by year, a second window displays the aligned words by order of frequency, so the user can immediately see which translation is the most common (see Figure 3 for an example).

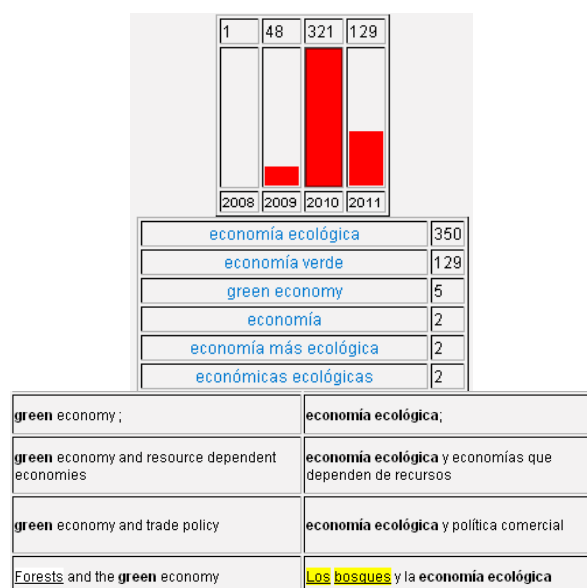


Figure 3: Concordancer for term “green economy”, the top graphic shows the term usage over years, then the most used translations, then the parallel segments with a link to the corresponding document.

4 Results/evaluation

4.1 Automatic evaluation

The BLEU and METEOR scores (Papineni et al. 2002, Denkowski & Lavie 2011) were used to compare human translation with automatic translation. It was decided to launch an automatic evaluation on a second test set: a random selection of 1,000 segments of new documents (i.e. documents published in January 2012, only one reference translation per document).

Table 1 gives some of our BLEU/METEOR scores according to specific experiments.

Table 1: Speed and scores computed using various configurations and tools (second test set, 1000 segments from “non-repetitive” documents published in January 2012)

Experiment	Speed seconds /segment	Model size ¹³	BLEU	METEOR
Baseline	7.60	69G	47.06	60.79
Pruned	6.39	12G	47.34	61.17
PrunedMax4	6.20	7.8G	47.31	61.23
Google translate ¹⁴	n/a	n/a	39.99	54.96
Bing translator ¹⁵	n/a	n/a	38.20	53.81
Mert optimized	6.30	7.8G	47.87	61.34

Reading Table 1 gave us a good idea on how well the system was performing. It performed better than the two publically available commercial tools. The speed was acceptable even if the models are stored on disk (pruning our models gave better or equal scores, while improving the size and speed of the models).

However human evaluators said that this second test set was belonging to a category of “non-repetitive” documents. The scores are much lower than our first BLEU score (65.45 on the first test set, see section 2.4). The reason was that the first BLEU was calculated with a set containing many segments from a Security Council resolution, which is a repetitive document, while the second test set contained narrative reports, non-cyclic reports, and documents non-related to the parliamentary processes of the Secretariat.

Then we decided to ask UN translators for examples of such “repetitive” documents, they gave us 13 new documents, containing 786 parallel segments, and the scores were much higher as shown on Table 2. These documents contained administrative and internal reports generated at the Secretariat, usually related to an administrative cycle, including budgetary and audit cycles, as well as Security Council and General Assembly resolutions. In general, these repetitive documents, as well as resolutions, are translated closer to the English version to keep parallelism, which in turn helps parliamentary negotiations,

¹³ The model size is the size of the “binarized” phrase table and reordering model with option *alignment-info* (<http://www.statmt.org/moses/?n=Moses.AdvancedFeatures#ntoc2>). The language model, not included, represents 1.7G.

¹⁴ <http://translate.google.com/> (February 2012)

¹⁵ <http://www.microsofttranslator.com> (February 2012)

while what we called “non-repetitive” documents are translated in a much freer writing style.

According to some estimations done by the Documents Control Unit, around 30% of the documents translated in New York have some degree of reprise, which might make them suitable for MT.

Table 2: BLEU/METEOR scores on "repetitive documents" (third test set: 786 segments)

System	BLEU	METEOR
Baseline	77.25	84.35
Google translate	59.70	73.77
Bing translator	58.49	73.69
Mert optimized	79.40	85.47

This BLEU score of 79.40 is quite impressive. A first general conclusion is that two systems have to be compared on exactly the same test set as shown by high differences of the three computed scores (47.87, 65.45 and 79.40).

4.2 Human evaluation of automatic translation

4.2.1 Preliminary evaluation

The first subjective evaluation was done by UN Spanish Translation Service in January 2012, translating a real job, a performance report of a peacekeeping mission (A/66/602) with the MT system. In order to verify the accuracy with the terminology produced by the MT system as compared to the mandatory terminology databases for this category of document, the job was translated using automatic terminology recognition (Mutiterm) The overall evaluation was that the automatic translation output was very good, in particular because the terms were accurate and consistent with the official terminology, and typing was significantly reduced. Even if most sentences needed reworking during post-editing, some were totally satisfactory.

One of the translators who were assigned to translate this document was a new recruit and her output was subject to revision. The reviser found that the quality of the translation was above average for a new recruit translating this challenging category of documents, as the terminology was consistently used, the meaning was accurate and the style was adequate. In the subjective opinion of the reviser, this might be an indication that for some categories of documents, the use of MT could help new recruits to produce translations better aligned with internal stylistic preferences and terminology; this would also apply for

contractors, who do not have access to the same document and terminology resources as the internal staff. This hypothesis must be further explored and validated with relevant tests.

Some other evaluations were done with different categories of documents, as notes for the President of body/organism sessions (very good quality), intranet news (poor quality) and administrative reports (good quality). As expected, a statistical machine translation tool trained with UN documents is not useful for translating all categories of documents, but a significant amount of them, in particular those that are included in the training and have some specific styles and terminology.

4.2.2 Set-up of the test

A second structured evaluation was done with three human evaluators, using the second test set. We knew that it was a “difficult” test set, however the output of human evaluation on such difficult test set is maybe more objective than on an easy one (as the third test set with close to 80 BLEU score). The three evaluators were chosen by the Chief of the Spanish Translation Service for their professionalism and were translators with more than 20 years of professional experience each. The evaluation was conducted over three full days. We have chosen to evaluate the translations using the known metrics: fluency and adequacy (see for example Denkowski & Lavie 2010).

Fluency rates how good the output Spanish is (using the following scale 5: Flawless 4: Good 3: Non-native 2: Disfluent 1: Incomprehensible) and adequacy rates the amount of information that has been transferred between original English and the Spanish translation (using the scale 5: All 4: Most 3: Much 2: Little 1: None).

At the time this evaluation was done the recasing was not working properly (partially fixed in later version), therefore we asked the evaluators to ignore case (*‘naciones unidas’* – in lower-case – is considered as a good translation for *‘United Nations’*).

4.2.3 Results

The three experts blindly (i.e. ignoring others’ judgments) evaluated the translation of the 1,000 segments (same segments as on Table 1). We decided not to display the reference translation, in order not to influence the judgment of the experts. A specific Web interface was built.

The experts had a minimal training on the evaluation tool and discussed about how to interpret and apply the metrics beforehand. In fact the three experts often agreed on the scores (when we compute the maximum disagreement score between the average on one evaluation, the overall average –on the 1000 evaluations– is 0.65 only).

On average the fluency is 3.94 the adequacy is 4.28.

Evaluators agreed on the final score, most of the content is maintained in the translation (adequacy more than 4), the fluency of the translation is almost “good” (fluency 3.94).

4.2.4 Feedback

The Spanish translators who participated in both evaluations as well as in other individual and informal tests found that the overall quality of the MT prototype output was good in general and very good for some specific categories of documents (for instance, peacekeeping budgets, as the ones used in Table 2), where a large volume of similar documents were included in the training. However, these particularly good MT documents were not included in the structured test. According to the feedback provided by some evaluators, the sentences included in the human evaluation were not the most repetitive and formulaic. For this reason, the use of domains might be advisable in the future. Although it is practically impossible to automatically sort the New York documents by categories using the UN symbol (an alphanumeric ID contained in all documents issued to the Official Document System¹⁶).

The Concordancer interface was used by the Senior Terminologist of the Spanish Translation Service, who also served as human evaluator, and she found that it was very useful to validate terminology records, despite some bugs in the current version.

Translators in other duty stations, including Vienna, Geneva and Santiago, were aware of the interfaces and were encouraged to try them. An additional training using Vienna and Geneva documents is expected at a later stage (these duty stations deal with a more limited and consistent set of subjects, so the duty station could be used as a proxy for domains).

According to the feedback received from the Chief of STS and other staff members, some translators and revisers are already using the tool for real jobs, in particular for some categories of documents, including Security Council and

peacekeeping. In their opinion, the quality of the output of the system is very high and lends itself for post-editing. These translators and revisers appreciate that the terminology is consistent with UN terminology and style norms. In this respect, the feedback is particularly positive from senior revisers. In effect, as they are used to revision and are familiar with UN standards, they find useful to work with MT and post-editing. Some other translators are using the system in combination with Trados and also report very high satisfaction.

As per the feedback of some UN users, in some categories of documents, the output of MT allows translators to speed-up the translation process. However, they report that this requires a different intellectual effort that is similar to revision but still more intensive, as in some cases, the system might produce sentences with high fluency but low accuracy (for instance, grammar is acceptable but the meaning is transferred partially or not at all). Translators and managers agree that further evaluations need to be done in order to validate the benefits of MT in productivity and quality, as well as to determine the threshold of usability of MT for post-editing. There is a strong interest from translators in STS in developing a bridge between the system and CAT tools (Trados and Mercury), as well as to develop a service to translate full documents. Finally, it is important to note that as a result of this experiment, the scope of *gText*, a current global project to develop terminology, reference and CAT tools for all UN duty stations, was expanded to include also the development of machine translation systems for all the UN official languages.

5 Conclusion and future work

We had to face a scalability problem with such a big corpus. However WIPO had already successfully trained a similar scale model. This experience shows that open source solutions can sometimes provide better results than generic commercial products. The data-driven approach requires limited human resource and still provides good results. It is planned to launch similar experiments with other language pairs: English-Russian, English-Chinese and English-Arabic. We expect worse results as it is more challenging than translating from English to Spanish or French due to the highly different morphological structure of the languages.

¹⁶ ods.un.org

In such an experiment the final word should always be left to the final users from UN.

They judged the Web interface as intuitive and requiring very little training. An integration with existing CAT tools is already on the way.

Future work includes: (a) testing of effort/productivity gains of MT and post-editing in some categories of documents and its use in conjunction with CAT tools (as in the experiment done by Plitt & Masselot, 2010), (b) testing the system with other language pairs (c) improving the user interface and (d) integrating with third party products.

Acknowledgements

The authors would like to thank the following managers and staff members of the Department for General Assembly and Conference Management of the United Nations Headquarters in New York for supporting the project: Mr. Shaaban M. Shaaban, Under-Secretary-General, Mr. Franz Baumann, Assistant Secretary-General, the members of the Departmental Management Group, the Information and Communications Technology Committee and the Technology Advisory Group of the Documentation Division, Ms. Maria Nobrega, Chief of the Spanish Translation Service, Ms. Ana Larrea, Training Officer of the Spanish Translation Service, and very specially, Igor Shpinov, who supported this project enthusiastically since its very beginning and facilitated its authorization. The project would not be possible without the dedicated collaboration and open-mindedness of Ms. Maria Barros, Ms. Rosario Fernandez and Ms. Carla Raffo, who served as human evaluators of the system and the staff of the Spanish Translation Service who provided constant feedback. .

Thank you to Laurent Gottardo for his idea in presenting bars per year in the concordancer. Special thanks to Paul Halfpenny for his valuable proof-reading.

References

- Denkowski, Michael & Alon Lavie. 2011. "Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems", Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation
- Denkowski, Michael & Alon Lavie. 2010. "Choosing the Right Evaluation for Machine Translation: an Examination of Annotator and Automatic Metric Performance on Human Judgment Tasks", *Proceedings of AMTA*
- Eisele, Andreas & Yu Chen. 2010. MultiUN: a multilingual corpus from United Nation documents. LREC 2010: proceedings of the seventh international conference on Language Resources and Evaluation, 17-23 May 2010, Valletta, Malta; pp.2868-2872.
- Gao, Qin & Stephan Vogel. 2008. Parallel implementations of word alignment tool. In Proceedings of the ACL'08 Software Engineering, Testing, and Quality Assurance Workshop
- Graff, David. 1994. UN Parallel Text (Complete). Linguistic Data Consortium, Philadelphia.
- Johnson, Howard, Joel Martin, George Foster, Roland Kuhn. 2007. Improving Translation Quality by Discarding Most of the Phrasetable. EMNLP-CoNLL 2007: 967-975
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris C. Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In Proceedings of ACL 07. Morristown, NJ, USA, 177-180.
- Koehn, Phillip. 2010. Statistical Machine Translation. textbook, Cambridge University Press, January 2010.
- Ma, Xiaoyi. 2006. Champollion: A Robust Parallel Text Sentence Aligner. Proceedings of LREC-2006
- McCandless, Michael, Erik Hatcher, Otis Gospodnetić. 2010. Lucene in Action. 2nd Edition. Manning Press.
- Papineni, K., S. Roukos, T. Ward, and WJ Zhu. 2002. BLEU: a method for automatic evaluation of machine translation, proc. of ACL 2002, pp. 311-318
- Pouliquen, Bruno & Christophe Mazenc, 2011, COPPA, CLIR and TAPTA: three tools to assist in overcoming the patent barrier at WIPO. *MT Summit XIII: the Thirteenth Machine Translation Summit*, 19-23 September 2011, Xiamen, China; pp.24-30
- Pouliquen, Bruno, Christophe Mazenc & Aldo Iorio: Tapta: a user-driven translation system for patent documents based on domain-aware statistical machine translation. *Proceedings of the 15th conference of the European Association for Machine Translation*, 30-31 May 2011, Leuven, Belgium; pp.5-12
- Plitt, M. & F. Masselot. 2010. A Productivity Test of Statistical Machine Translation Post-Editing in a Typical Localisation Context. *The Prague Bulletin of Mathematical Linguistics*, 93(01/2010): p 7-16
- Rafalovitch Alexandre & Robert Dale. 2009. United Nations general assembly resolutions: a six-language parallel corpus. MT Summit XII: proceedings of the twelfth Machine Translation Summit, 26-30/08/2009, Ottawa, Ontario, Canada; pp.292-299.
- Stolke, Andreas. 2002. SRILM an extensible language modeling toolkit. In Proceedings of the International Conference on Spoken Language Processing.

User Evaluation of Interactive Machine Translation Systems

Vicent Alabau, Luis A. Leiva, Daniel Ortiz-Martínez, Francisco Casacuberta

ITI/DSIC – Universitat Politècnica de València

{valabau, luileito, dortiz, fcn}@{iti, dsic}.upv.es

Abstract

Recent developments in search algorithms and software architecture have enabled multi-user web-based prototypes for Interactive Machine Translation (IMT), a technology that aims to assist, rather than replace, the human translator. Surprisingly, formal human evaluations of IMT systems are highly scarce in the literature. To this regard, we discuss experiences gained while testing IMT systems. We report the lessons learned from two user evaluations. Our results can provide researchers and practitioners with several guidelines towards the design of on-line IMT tools.

1 Introduction

Research in machine translation (MT) aims to develop computer systems which are able to translate documents without human intervention. However, current translation technology has not been able to deliver full automated error-free translations. Typical solutions to improve the quality of an MT system require manual post-editing. This serial process does not allow integrating the knowledge of the human translator into the system decisions.

One alternative to take advantage of the existing MT technologies is to apply the so-called interactive machine translation (IMT) paradigm (Langlais et al., 2002). The IMT paradigm adapts data driven MT techniques for its use in collaboration with human translators. Following these ideas, Barrachina et al. (2009) proposed a new approach to IMT, in which fully-fledged statistical MT systems are used to produce full target sentences hypotheses, or portions thereof, which can be accepted or amended by a human translator. Each corrected text segment is then used by the MT system as additional information to achieve improved suggestions. Figure 1 shows a minimal IMT session example.

source: Para ver la lista de recursos

reference: To view a listing of resources

suggestion	s	To view the resources list
interaction	p	To view
	k	<input type="text" value="a"/> listing of resources
	s	To view a listing of resources
accept	p	To view a listing of resources

Figure 1: An IMT session example, using only 1 key stroke (k) to achieve the reference sentence. Notice that the user submits partial sentences (p) to the system, which tries to complete them (s).

Following the IMT paradigm, recent developments in search algorithms and software architecture have allowed multi-user web-based translation prototypes. These systems have grown in features, e.g., allowing advanced multimodal interaction, which have also added extra complexity to the prototypes. Then, their effectiveness should be tested with respect to technology dissemination. While pure data-driven evaluations have already shown that IMT is a promising technology (Barrachina et al., 2009), surprisingly, formal human evaluations are highly scarce in the literature.

In this paper, we describe our experiences evaluating two IMT prototypes with real users: an initial, advanced version and a simplified but improved version. Our results identify important design issues, which open a discussion regarding how IMT systems should be deployed.

2 Related Work

Langlais et al. (2002) performed a human evaluation on their IMT prototype. They emulated a realistic working environment in which the users could obtain automatic completions for what they were typing. Users reported an improvement in performance; however, raw productivity decreased by 17%, although the users appreciated the tool and were confident to improve their productivity after proper training. That work was extended in the TT2 project (Casacuberta et al., 2009), where the

performance tended to increase as the participants grew accustomed to the system, over a 18-month period. A slightly different approach was studied in (Koehn, 2010). There, monolingual users evaluated a translation interface supporting IMT predictions and the so-called ‘translation options’. When translating from undecipherable languages (as Chinese or Arabic for an English speaker), richer assistance improved user performance.

3 User Interfaces and Evaluation

Previous research on multimodal interfaces in natural language processing have shown a comprehensible tendency to choose an interactive collaborative environment over a manual system for non-expert computer users (Leiva et al., 2011). We followed this approach to build a prototype with an IMT backend. We will refer to this system as the advanced demonstrator (IMT-AD, Figure 2) since it implemented a number of complementary features, which conditioned the design of the interface; e.g., the use of one boxed text field per sentence word aimed to ease e-pen interaction.

3.1 Evaluation of the Advanced Prototype

The goal of this evaluation was aimed to assess both qualitatively and quantitatively IMT-AD, and compare it to a state-of-the-art post-editing (PE) MT output. Translating from scratch was not considered since this practice is being increasingly displaced by assistive technologies. Indeed, PE of MT systems is found frequently in a professional translation workflow (TT2, 2001). Thus, in addition to IMT-AD, a post-editing version of the demonstrator (PE-AD) was developed to make a fair comparison with state-of-the-art PE systems. PE-AD used the same interface as IMT-AD, but the IMT engine was replaced by autocompletion-only capabilities as found in popular text editors.

Design Both systems were evaluated on the basis of the ISO 9241-11 standard (ergonomics of human-computer interaction). Three aspects were considered: efficiency, effectiveness, and user satisfaction. For the former, we computed the average time in seconds that took to complete each translation. For the second, we evaluated the BLEU against the reference and a crossed multi-BLEU among users’ translations. For the latter, we adapted the system usability scale (SUS) questionnaire to score the user satisfaction, by asking 10 questions that users would assess in a 1–5 Likert

scale (1:strongly disagree, 5:strongly agree), plus a text area to submit free-form comments.

Participants A group of 10 users (3 females) aged 26–43 from our research group volunteered to perform the evaluation as non-professional translators. All of them were proficient in Spanish and had an advanced knowledge of English. Although none had worked with IMT systems, all knew the basis of the IMT paradigm.

Apparatus Since participants were Spanish natives, we decided to perform translations from English to Spanish. We chose a medium-sized corpus, the EU corpus, typically used in IMT (Barachina et al., 2009), which consists of legal documents. We built a glossary for each source word by using the 5-best target words from a word-based translation model. We expected this would cover the lack of knowledge for our non-expert translators towards this particular task. In addition, a set of 9 keyboard shortcuts was designed, aiming to simulate a real translation scenario, where the mouse is typically used sparingly. Furthermore, autocompletion was added to PE-AD, i.e., words with more than 3 characters were autocompleted using a task-dependent word list. In addition, IMT-AD was set up to predict at character level interactions. We disabled the complementary features to focus the evaluation on basic IMT.

Procedure Three disjoint sentence sets (C1, C2, C3) were randomly selected from the test dataset. Each set consisted of 20 sentence pairs and kept the sequentiality of the original text. Sentences longer than 40 words were discarded. C3 was used in a warm up session, where users gained experience with the IMT system (5–10 min per user on average) before carrying out the actual evaluation. Then, C1 and C2 were evaluated by two user groups (G1, G2) in a counterbalanced fashion: G1 evaluated C1 on PE-AD and C2 on IMT-AD, while G2 did C1 on IMT-AD and C2 in PE-AD.

Results Although the results were not conclusive (there were no statistical differences between groups), we observed some trends. First, the time spent (efficiency) per sentence on average in the IMT system was higher than in PE (67 vs. 62 s). However, the effectiveness was slightly higher for IMT in BLEU with respect to the reference (41.5 vs. 40.7) and with respect to a cross-validation with other user translations (78.9 vs. 77.4). This

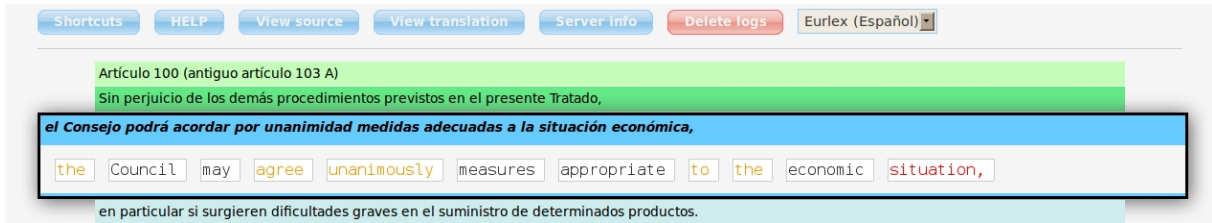


Figure 2: Detail of the advanced web-based interface with a boxed text field for each word.

	PE-AD	IMT-AD
Avg. time (s)	62 ($SD = 51$)	67 ($SD = 65$)
BLEU	40.7 (13.4)	41.5 (13.5)
Crossed BLEU	77.4 (4.5)	78.9 (4.8)
Global Satisfaction	2.5(1.2)	2.1(1.2)

Table 1: Summary of the results for the first test.

suggested that the IMT system helped to achieve more consistent and standardized translations.

Finally, users perceived the PE system more satisfactorily than the IMT system, although the global scores were 2.5 for PE and 2.1 for IMT, which suggested that users were not comfortable with none of the systems. IMT failed to succeed in questions regarding the system being easy to use, consistent, and reliable. This was corroborated by the submitted comments. Users complained about having too many shortcuts and available edit operations, some operations not working as expected, the word-box based interface, and some annoying common mistakes in the predictions of the IMT engine (e.g., inserting a whitespace instead of completing a word, which would be interpreted as two different words). One user stated that the PE system “was much better than the [IMT] predictive tool”. Regarding PE, users mainly questioned the usefulness of the autocompletion feature.

3.2 Simplified Web Based Prototype

The results from the first evaluation were quite disappointing. Not only participants took more time to complete the evaluation with IMT-AD, but they also perceived that IMT-AD was more cumbersome and unreliable than PE-AD. However, we still observed that IMT-AD had been occasionally beneficial, and probably the bloated UI was the cause for IMT to fail. Thus, we developed a simplified version of the original prototype (Figure 3).

Design In this case, the word-box based interface was changed to a simple text area. In addition,

the edit operations were simplified to allow only word substitutions and single-click rejections. Besides, we expected that the simplification of the interface logic would reduce some of the programming bugs that bothered users in the first evaluation. The PE interface was simplified in the same way. Furthermore, the autocompletion feature was improved to support n -grams of arbitrary length.

Participants Fifteen participants aged 23–34 from university English courses (levels B2 and C1 from the Common European Framework of Reference for Languages) were paid to perform the evaluation (5 € each). A special price of 20 € was given to the participant who would contribute with the most useful comments about both prototypes. It was found that, following this method, participants were more verbose when providing feedback.

Apparatus In this case, a different set of sentences ($C1'$, $C2'$, $C3'$) was randomly extracted from the EU corpus.

Procedure To avoid the bias regarding which system was being used, sentences were presented in random order, and the type of system was hidden to the participants. As a consequence, users could not evaluate each system independently. Therefore, a reduced questionnaire with just two questions was shown on a per-sentence basis. **Q1** asked if the system suggestions were useful. **Q2** asked if the system was cumbersome to use. A text area for free-form comments was also included.

Results Still with no statistical significance, we found that the IMT prototype was perceived now better than PE. First, interacting with IMT was more efficient than with PE on average (55 s vs. 69 s). The number of interactions was also lower (79 vs. 94). Concerning user satisfaction, the IMT system was perceived as more helpful (3.5 vs. 3.1) but also more cumbersome (3.1 vs. 2.9). However, in this case the differences were narrower. On the

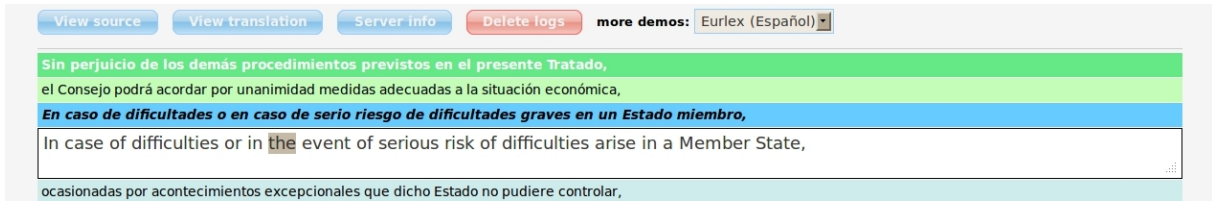


Figure 3: Detail of the simplified web-based interface.

	PE-BD	IMT-BD
Avg. time (s)	69 ($SD = 42$)	55 ($SD = 37$)
No. interactions	94 (60)	79 (55)
Q1 (Likert scale)	3.1 (1.2)	3.5 (1.1)
Q2 (Likert scale)	2.9 (1.2)	3.1 (1.3)

Table 2: Summary of results for the second test.

other hand, IMT received 16 positive comments whereas PE received only 5. Regarding negative comments, the counts were 35 (IMT) and 31 (PE). While the number of negative comments is similar, there was an important difference regarding the positive ones. Finally, the users' complaints of the IMT system can be summarized in the following items: *a)* system suggestions changed too often, offering very different solutions; *b)* while correcting one mistake, subsequent words that were correct were changed by a worse suggestion; *c)* system suggestions did not keep gender, number, and time concordance; *d)* if the user goes back in the sentence and performs a correction, parts of the sentence already corrected were not preserved on subsequent system suggestions.

4 Discussion and Conclusions

Our initial UI performed poorly when tested with real users. However, when the UI design was adapted to the users' expectations, the results were encouraging. Note that in both cases the same IMT engine was evaluated under the hood. This fact remarks the importance of the UI design when evaluating a highly interactive system as IMT is.

The literature had reported good experimental results in simulated-user scenarios, where IMT is focused on optimizing some automatic metric. However, user productivity is strongly related to how the user interacts with the system and other UI concerns. For instance, a suggestion that changes on every key stroke might obtain better automatic results, whereas the user productivity decreases because of the cognitive effort needed to process

those changes. Therefore, a new methodology is required for optimizing interactive systems (like IMT) towards the user.

In sum, the following issues should be addressed in an IMT system: *1)* user corrections should not be modified, since that causes frustration; *2)* system suggestions should not change dramatically between interactions, in order to avoid confusing the user; *3)* the system should propose a new suggestion only when it is sure that it improves the previous one.

We hope these considerations will reduce the gap between translators and researchers needs, so that future developments can have an impact on the translation industry.

Acknowledgments

This research has received funding from the EC's 7th Framework Programme (FP7/2007-2013) under grant agreement No. 287576 - CasMaCat, and from the Spanish MEC/MICINN under the MIPRCV project (CSD2007-00018). We would also like to thank the participants and the Centro de Lenguas at the UPV.

References

- Barrachina, S., O. Bender, F. Casacuberta, J. Civera, E. Cubel, S. Khadivi, A. L. Lagarda, H. Ney, J. Tomás, and E. Vidal. 2009. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35(1):3–28.
- Casacuberta, F., J. Civera, E. Cubel, A. L. Lagarda, G. Lapalme, E. Macklovitch, and E. Vidal. 2009. Human interaction for high quality machine translation. *Communications of the ACM*, 52(10):135–138.
- Koehn, P. 2010. Enabling Monolingual Translators: Post-Editing vs. Options. In *Proc. ACL-HLT*.
- Langlais, P., G. Lapalme, and M. Loranger. 2002. TRANSType: Development-Evaluation Cycles to Boost Translator's Productivity. *Machine Translation*, 15(4):77–98.
- Leiva, L. A., V. Romero, A. H. Toselli, and E. Vidal. 2011. Evaluating an Interactive-Predictive Paradigm on Handwriting Transcription: A Case Study and Lessons Learned. In *Proc. COMPSAC*.
- TT2. 2001. TransType2 - Computer Assisted Translation. Project Technical Annex. Information Society Technologies (IST) Programme, IST-2001-32091.

Oral Session 2 – Research Papers

Translate, Predict or Generate: Modeling Rich Morphology in Statistical Machine Translation

Ahmed El Kholly and Nizar Habash

Center for Computational Learning Systems, Columbia University

475 Riverside Drive New York, NY 10115

{akholy, habash}@ccls.columbia.edu

Abstract

We compare three methods of modeling morphological features in statistical machine translation (SMT) from English to Arabic, a morphologically rich language. Features can be modeled as part of the core translation process mapping source tokens to target tokens. Alternatively these features can be generated using target monolingual context as part of a separate generation (or post-translation inflection) step. Finally, the features can be predicted using both source and target information in a separate step from translation and generation. We focus on three morphological features that we demonstrate through a manual error analysis to be most problematic for English-Arabic SMT: gender, number and the determiner clitic. Our results show significant improvements over a state-of-the-art baseline (phrase-based SMT) of almost 1% absolute BLEU on a medium size training set. Our best configuration models the determiner as part of core translation and predicts gender and number separately, and handles the rest of the features through generation.

1 Introduction

Translation into English has been the focus of many research efforts in Statistical Machine Translation (SMT). However, recently, translation into other languages has been receiving increasing attention, especially translation into morphologically rich languages (Sarıkaya and Deng, 2007; Elming and Habash, 2009; Yeniterzi and Oflazer, 2010).

One of the main issues in SMT is the sparsity of parallel data for many language pairs espe-

cially when the source or target language is morphologically rich. Morphological richness comes with many challenges and the severity of these challenges increases when translating from a morphologically poor language to a morphologically richer language.

In this paper, we address these challenges through different modeling methods.¹ In our approach, morphological features can be modeled as part of the core translation process mapping source tokens to target tokens. Alternatively these features can be generated using target monolingual context as part of a separate generation (or post-translation inflection) step. Finally, the features can be predicted using both source and target information in a separate step before generation. We focus in our experiments on English-Arabic SMT and we work on three morphological features that we found, through a manual error analysis, to be most problematic for English-Arabic SMT: gender, number and the determiner clitic. Our results show improvements over a state-of-the-art baseline (phrase-based SMT) of almost 1% absolute BLEU on a medium size training set of 4M words. Our best configuration models the determiner as part of core translation, predicts gender and number features separately, and handles the rest of the features through generation. We test our approach on a blind test set and we got the same relative improvements across the different systems. However, when scaling up the data set, the advantage of using morphological modeling disappears, which is not surprising.

2 Related Work

There have been numerous efforts studying the effect of applying morphological processing or using morphological information on SMT quality. In one approach, Factored SMT, morphological features can be modeled jointly as factors in the trans-

lation process (Koehn et al., 2007). These factors can be used in different translation and generation expansion steps. One of the main drawbacks of this approach is the combinatorial expansion of the number of translation options.

Another approach is to model translation and morphology independently in a sequential manner. A common method within this approach is to morphologically preprocess the training data before training the translation models, e.g., morphological tokenization of clitics (Habash and Sadat, 2006; Oflazer and Durgar El-Kahlout, 2007; Badr et al., 2008). Tokenization reduces sparsity of the data and increases the symmetry between source and target, which in return improves the quality of the translation. There is a large space of different tokenization schemes for Arabic. In our experiments, we use the Penn Arabic Treebank (PATB) tokenization scheme which was shown in previous effort by El Kholy and Habash (2010a) to perform well when translating into Arabic. As a result of tokenization, a post-processing step is needed to recombine (detokenize) the clitics back to the word. This is a somewhat complex task involving several orthographic and morphological adjustments (El Kholy and Habash, 2010b).

Another method related to our approach is using an independent morphological prediction component such as used by Minkov et al. (2007) and Toutanova et al. (2008). They use maximum entropy models for inflection prediction. Unlike our approach, they predict inflected word forms directly without going into a fine grained morphological feature prediction as we do. One of the main drawbacks of their approach is that they use stems as their base for translation instead of lemmas (see Section 3.1). On average, a lemma in Arabic could have two stems so using lemmas can make the data less sparse and make the translation model tighter. There is also work by Clifton and Sarkar (2011) where they do segmentation and morpheme prediction. They also use stems as their basic word form.

3 English-Arabic SMT Challenges

In this section, we discuss the challenges of the English-Arabic language pair in the context of MT. We also provide two error analyses that helped define the scope of our work and motivated our experimental setup.

3.1 Linguistic Facts

Unlike English, a morphologically poor language, Arabic is morphologically complex and has a large set of morphological features producing numerous word forms. While the number of (morphologically untokenized) Arabic words in a parallel corpus is 20% less than the number of corresponding English words, the number of unique Arabic word types is over twice the number of unique English word types over the same corpus size.

One aspect of Arabic’s complexity has to do with its orthography which often omits short-vowel diacritics. As a result, ambiguity is rampant. Another aspect of Arabic that contributes to this complexity is its various attachable clitics which include conjunction proclitics, e.g., $+و$ $w+$ ‘and’, particle proclitics, e.g., $+ل$ $l+$ ‘to/for’, the definite article $+ال$ $Al+$ ‘the’, and the class of pronominal enclitics, e.g., $+هم$ $+hm$ ‘their/them’. Beyond these clitics, Arabic words inflect for person (PER), gender (GEN), number (NUM), aspect (ASP), mood (MOD), voice (VOX), state (STT) and case (CAS). This morphological richness leads to thousands of inflected forms per lemma and a high degree of ambiguity: about 12 analyses per word, typically corresponding to two lemmas on average (Habash, 2010). The PATB tokenization scheme (Maamouri et al., 2004) which we use in our baseline and all experiments separates all clitics except for the determiner clitic $Al+$ (DET).

Arabic also has complex morpho-syntactic agreement rules in terms of GEN and NUM within specific constructions such as nouns with their adjectives and verbs with their subjects (Alkuhlani and Habash, 2011). The DET in Arabic is used to distinguish different syntactic constructions such as the possessive or adjectival modification.

English on the other hand barely inflects for NUM and tense and for PER in a limited context. The NUM feature in Arabic has more values (dual) than English. GEN in English is not expressed morphologically. When translating from English into Arabic, we expect to be able to model shared morphological features more than absent features or features expressed only syntactically in English or Arabic, e.g., the possessive construction.

3.2 Automatic Error Analysis

We conducted an error analysis of our baseline system on our development set (MT05) using an open-source tool for error analysis of natural lan-

guage processing tasks targeting morphologically rich languages (El Kholy and Habash, 2011). The tool aligns words in the output and the reference if they share the same lemma. Each output word receives a matching category based on the reference word it is paired with. If the output and reference words have same form, the category is Exact Match, otherwise, it is Lemma Match. Unpaired output words are tagged Unmatchable. The tool also produces detailed statistics on morphological errors. Exact Match cases are 59.0% and Lemma Match cases are 13.3%. Among Lemma Match cases, DET is the biggest single feature error. The PATB clitics errors (53.6%) together with DET (29.7%), GEN (12.8%) and NUM (10.8%) are the biggest culprits overall. This analysis suggests targeting them may be most beneficial.

3.3 Manual Error Analysis

We also performed a manual error analysis on a hundred sentences from the output of the MT05 set translated with the baseline system. Exact Match cases are 57% and Lemma Match cases are 15%. Among Unmatchable cases, 21.4% of the words have good paraphrases. We looked at the morphological errors that affect adequacy and fluency. We define a morphological adequacy error as the mistranslation of a certain morphological feature conveying a different meaning from the English. Morphological fluency errors are morpho-syntactic disagreements in the Arabic output. Table 1 summarizes our findings. In terms of adequacy, DET along with NUM are the biggest culprits overall. In terms of fluency, GEN is far worse than any other feature which highlights its importance. Another important observation is that the union of the words which affect both fluency and/or adequacy are almost 6.5% which defines the upper limit of words that can improve through morphological modeling.

4 Approach

In our approach, the process of translating English words to Arabic words is broken into a pipeline consisting of four steps:

- **Lexical Translation** from English words to tokenized Arabic lemmas and any subset of Arabic linguistic features.
- **Morphology Prediction** of linguistic features to inflect Arabic lemmas.
- **Morphology Generation** of inflected Arabic tokens from Arabic lemmas and any subset of

Arabic linguistic features.

- **Detokenization** of inflected Arabic tokens into surface Arabic words.

Arabic tokenization and lemmatization are done before training the translation models. Both lexical translation and generation are implemented as phrase-based SMT systems (Koehn et al., 2007). Morphology prediction is an optional step implemented using a supervised discriminative learning model. Generation can be done from lemmas and any subset of Arabic inflectional features. Detokenization simply stitches the words and clitics together as a post-processing step (Badr et al., 2008; El Kholy and Habash, 2010a).

We follow numerous previously published efforts on the value of tokenization for English-Arabic SMT (Badr et al., 2008; El Kholy and Habash, 2010a; Al-Haj and Lavie, 2010) and focus on the question of how to improve the translation of tokenized words using deeper representations, namely lemmas and features. Within our framework, we can model the translation of different Arabic linguistic features as part of the lexical translation step, as part of the generation step, or model them using an independent morphology prediction step. Some features, such as clitics, can be modeled well through simple tokenization and detokenization (which can be thought of as part of lexical translation).

We build on a previous effort in improving the quality of the English-to-Arabic translation through Arabic tokenization (El Kholy and Habash, 2010a). We use the best performing tokenization scheme (PATB) and the best detokenization technique on the output as our baseline. Consequently, in this paper we focus on the first three components of the pipeline and we keep the tokenization a constant across all experiments. We study different options of including three morphological features (GEN, NUM and DET) in the first three steps of the pipeline and their implications on the quality of English-to-Arabic SMT. We discuss the three steps in the following subsections.

4.1 Lexical Translation

Lexical translation is the first step in our decoding pipeline. It is trained on pre-processed text: tokenized, lemmatized and disambiguated Arabic words and English words (with limited processing) and their POS tags. We use an SMT system to translate from English words (ENGWORD) and POS tags (POS) to tokenized Arabic lemmas (AR-

Words with Morphological Errors Affecting		Percentage of Morphology Error Type								
		Tokenized PATB Clitics			Non-tokenized Morphological Features					
		CONJ	PART	PRON	DET	PER	ASP	GEN	NUM	CAS
Adequacy	2.6	3.6	7.3	7.3	38.2	1.8	7.3	12.7	30.9	0.0
Fluency	5.1	10.8	13.5	7.2	18.9	0.9	0.9	41.4	19.8	2.7
Adequacy \cup Fluency	6.5	9.6	13.0	7.6	26.8	1.4	3.4	34.2	17.8	2.0

Table 1: Column two presents the percentage of words with morphological errors that affect the adequacy and fluency of the translation quality. Starting from column three till the end are percentages of the error contributed by each morphological features. Since multiple errors can occur, these values overlap.

ALEM) plus zero or more morphological features. We use an abstract representation for the morphological features so that each word is represented as a lemma and a set of feature-value pairs. Table 2 shows a sample sentence in the above-mentioned representations. This way we simplify the translation task by targeting a less complex output. The key point here is to keep the morphological features that help the translation task and then try to generate the rest of the morphological features and inflected forms in later steps. The output of lexical translation is input to the morphological generation step directly or is first enriched by additional morphological features predicted in the morphology prediction step.

4.2 Morphology Prediction

Morphology prediction takes the output of lexical translation and tries to enrich it by predicting one or more morphological features. Unlike Toutanova et al. (2008), who predict full inflected forms and Clifton and Sarkar (2011) who predict morphemes, we predict morphological feature. This task is, in sense, a form of POS tagging. However, unlike typical tagging, which is done on fully inflected word forms, this task is applied to uninflected or semi-inflected forms – lemmas with zero or more morphology features. As such, we do not expect it to do as well as normal POS tagging/morphology disambiguation for Arabic (Habash and Rambow, 2005).

We use a Conditional Random Field (CRF) toolkit (Lafferty et al., 2001) to train a prediction module with a variety of learning features (not to be confused with the tagged linguistic features). We also make use of the alignment information produced by the MT system in the lexical translation step to get the equivalent aligned English word of each translated word. We then use this information in addition to some syntactic information on the English side as CRF learning features.

We group the CRF learning features into two sets: *Basic* and *Syntax*. The *Basic* features con-

sist of the Arabic output from the lexical translation step (lemma plus certain features), the equivalent aligned English word, English POS and English context (+/- two words). The *Syntax* features consist of the English parent word in a dependency tree, the dependency relation and the equivalent Arabic output word of the English parent. English is parsed using the Stanford Parser (Klein and Manning, 2003).

In training the CRF model, we use the same data used in training the lexical translation step (Section 5). We create three datasets from this data. The first is the original gold data where we train the CRF module on clean Arabic text and gold feature values that are determined using a state-of-the-art POS tagger for Arabic (Habash and Rambow, 2005). Although the automatic tagging does produce errors, we still call this data set *gold* since the Arabic is correctly inflected naturally occurring text. The second dataset is created by translating the whole data using the translation model created by the lexical translation step. The intuition here is to model lexical translation errors by training the CRF models on data similar in quality to its expected input. The last dataset is the combination of gold and translated dataset.

Table 3 shows the accuracy of the CRF module on a test set of 1000 sentences. CRF in general achieves a high accuracy across the different training datasets and the different training parameters. Using translated data does not outperform using gold data; however, the accuracy of predicting NUM and GEN seems to benefit from adding the translated data to the gold data. That could be explained by the fact that NUM and GEN are more affected by translation adequacy unlike DET which is more coupled with translation fluency. Overall the results are about 10-14% absolute lower than MADA (Habash and Rambow, 2005) tagging of the same features on fully inflected text; and are 20-30% absolute better than a degenerate baseline using the most common feature value.

Representation	Example
ENGWORD	saddam hussein 's half-brother refuses to return to iraq
ENGWORD)+POS	saddam#NN hussein#NN 's#POS half-brother#NN refuses#VBZ to#TO return#VB to#TO iraq#NN
ARALEM	Āax γayor šaqiyq li+ Sad~Am Husayon rafaD çawodaĥ lilaý çirAq
ARALEM+DET	Āax#det γayor#0 šaqiyq#det li+#na Sad~Am#0 Husayon#0 rafaD#0 çawodaĥ#det Āilaý#na çirAq#det
Arabic Tokenized	AlĀx γyr Alšqyq l+ SdAm Hsyn yrfD Alçwdĥ ĀlĀy AlçrAq
Arabic Script	الأخ غير الشقيق لصدام حسين يرفض العودة الى العراق

Table 2: A sample sentence showing the different representations used in our experiments.

The morphology prediction step produces a lattice with all possible feature values each having an associated confidence score. The morphology generation module discussed next will decide on the best option.

Prediction Training		Predicted Feature Accuracy		
Data Set	Model	GEN	NUM	DET
Gold	Basic	84.65	88.76	88.00
	Basic+Syntax	84.22	89.11	87.85
Translated	Basic	84.46	86.11	85.98
	Basic+Syntax	84.08	86.79	85.41
Gold +Translated	Basic	85.96	89.43	87.40
	Basic+Syntax	85.49	89.52	86.91

Table 3: Accuracy (%) of feature prediction starting from Arabic lemmas. A most-common-tag degenerate baseline would yield 67.4%, 70.6% and 59.7% accuracy for GEN, NUM, and DET, respectively. Reported MADA classification accuracy starting from fully inflected Arabic is as follows: GEN 98.2% , NUM 98.8%, DET 98.3% (Habash and Rambow, 2005).

4.3 Morphology Generation

Morphology generation maps Arabic lemmas (ARALEM) plus morphological features to Arabic inflected forms. This step is implemented as an SMT system that translate from a deeper linguistic representation to a surface representation of each token. This step is conceptually similar to the generation expansion component in Factored SMT, but it is implemented as a complete SMT system. The main advantage of this approach is that the training data is not restricted to parallel corpora. We can use all the monolingual data we have in building the system. For more details, see (El Kholly and Habash, 2012).

To evaluate the performance of this approach in generating Arabic inflected forms, we built several SMT systems translating from ARALEMs plus zero or more morphological features to Arabic inflected form. We use the same tools and setup as discussed in Section 5. Table 4 shows the BLEU scores of generating the MT05 set starting from Arabic lemmas plus different morphological fea-

Gold Generation Input	BLEU%
ARALEM	82.19
ARALEM+DET	86.62
ARALEM+NUM	86.89
ARALEM+GEN	87.32
ARALEM+GENNUM	90.18
ARALEM+GENNUMDET	94.77

Table 4: Results of generation from gold ARALEM plus different sets of morphological features. Results are in (% BLEU) on the MT05 set.

tures (GEN, NUM, DET), and their combinations. As expected, the more features are included the better the results. Here comes the trade off between the lexical translation quality and morphological generation. The BLEU scores are very high because the input is golden in terms of word order and lemma choice. These scores should be seen as the upper limit on correctness that can be expected from this step, rather than its actual performance in an end-to-end pipeline.

The morphology generation step can take the output of lexical translation directly or after predicting certain morphological features using the morphology prediction step.

5 Experiments

In this section, we present our results comparing the modeling of GEN, NUM and DET features, first as part of lexical translation versus morphological generation, and then as part of morphological prediction versus morphological generation. We also present results on a blind test set MT06, a much larger training corpus, and discuss our findings.

5.1 Experimental Setup

All of the training data we use is available from the Linguistic Data Consortium (LDC).² We use an English-Arabic parallel corpus of about 142K sentences and 4.4 million words for translation model training data. The parallel text includes Arabic News (LDC2004T17),

²<http://www ldc upenn edu>

eTIRR (LDC2004E72), English translation of Arabic Treebank (LDC2005E46), and Ummah (LDC2004T18). Word alignment is done using GIZA++ (Och and Ney, 2003). For language modeling, we use 200M words from the Arabic Gigaword Corpus (LDC2007T40) together with the Arabic side of our training data. We used 5-grams for all LMs implemented using the SRILM toolkit (Stolcke, 2002).

MADA is used to tokenize the Arabic text and produce lemmas and their accompanied morphological features. English preprocessing simply includes down-casing, separating punctuation and splitting off “’s”.

All experiments are conducted using the Moses phrase-based SMT system (Koehn et al., 2007). The decoding weight optimization was done using a set of 300 sentences from the 2004 NIST MT evaluation test set (MT04). The tuning is based on tokenized Arabic without detokenization. We use a maximum phrase length of size 8. We report results on the 2005 NIST MT evaluation set (MT05). These test sets were created for Arabic-English MT and have four English references. We arbitrarily picked the first English reference to be source and used the Arabic source as the only reference. We evaluate using BLEU-4 (Papineni et al., 2002).

Our baseline replicates the work of El Kholy and Habash (2010a), who determined that tokenizing Arabic into the PATB tokenization scheme is optimal for phrase-based SMT models. The baseline BLEU score is 29.48% using exactly the same data sets used in the rest of the experiments.

5.2 Translation vs. Generation

We compare the performance of translating English and English plus POS into Arabic lemmas plus different morphological feature combinations followed by generation of the final Arabic inflected form using the morphology generation step directly under the same conditions. The results are presented in Table 5. The best performer across all conditions is translating English words to Arabic lemmas plus DET. This is the only setup that beats the baseline system. The difference in BLEU scores between this setup and the baseline is statistically significant above the 95% level. Statistical significance is computed using paired bootstrap resampling (Koehn, 2004). This shows the importance of DET in lexical translation. English POS oddly does not help. This is perhaps a result of the

Input	A'	BLEU%
ENGWORD	ARALEM	29.47
ENGWORD+POS	ARALEM	29.26
ENGWORD	ARALEM+NUM	28.96
ENGWORD+POS	ARALEM+NUM	28.52
ENGWORD	ARALEM+GEN	28.81
ENGWORD+POS	ARALEM+GEN	28.65
ENGWORD	ARALEM+DET	30.13
ENGWORD+POS	ARALEM+DET	29.33
ENGWORD	ARALEM+GENNUM	28.82
ENGWORD+POS	ARALEM+GENNUM	28.65
ENGWORD	ARALEM+GENNUMDET	29.19
ENGWORD+POS	ARALEM+GENNUMDET	29.00

Table 5: End-to-end MT results for different settings of English input and Intermediate Arabic. Results are in (% BLEU) on our MT05 set.

added sparsity in how we modeled them (as ENGWORD+POS). It is possible a factored MT model can give different results. We plan to explore this question in the future.

5.3 Prediction vs. Generation

We compare results of two translation settings and a variety of added predicted features. The results are presented in Table 6. We can see from the results that using predicted GEN by itself does not help across the board yet it could be helpful when combined with other features. It also seems that predicting NUM when lexical translation is done with lemmas only helps the performance but that is not the case when the lexical translation is done using Lemma plus DET. Another observation is that combining GEN and NUM degrades the overall performance more than the GEN by itself; however, we get the best scores when DET is combined with them. This shows that some synergies come out when different features are combined together even if they perform badly on their own. The only fact that seems very robust is that translating English to Lemma plus DET and then predicting both GEN and NUM gives the highest scores. Predicting features using models trained on translated texts seem to also consistently do better than using models that are trained on original Arabic. The best result obtained is statistically significant compared with the best reported score in the previous section (ARALEM+DET translation).

5.4 Blind Test

We performed a blind test using the 2006 NIST MT evaluation set (MT06) and compared the results to (MT05). MT06 is a harder set to translate than MT05. However, the relative performance is

Translation	ENGWORD→ARALEM					ENGWORD→ARALEM+DET		
No Prediction	29.47					30.13		
Prediction Training	Predicted Morphological Features							
	GEN	NUM	DET	GEN+NUM	GEN+NUM+DET	GEN	NUM	GEN+NUM
Gold Basic	28.62	29.54	29.67	28.41	29.81	29.85	29.91	30.36
+Syntax	28.64	29.51	29.67	28.40	29.86	29.85	29.90	30.38
Trans Basic	28.90	29.55	29.80	28.32	29.90	29.91	29.89	30.37
+Syntax	28.87	29.58	29.80	28.77	29.90	30.02	29.92	30.41
Gold+Trans Basic	28.96	29.59	29.77	28.77	30.02	29.98	30.01	30.42
+Syntax	28.93	29.60	29.77	28.75	30.03	29.99	30.01	30.43*

Table 6: End-to-end MT results for two translation settings and a variety of added predicted features. Results are in (% BLEU) on our MT05 set. The best result in each column is bolded. The best overall result is marked with *.

maintained (around 3% relative BLEU) as shown in Table 7. Translating through Lemma plus DET and then predicting GEN and NUM is still the best option.

Model	MT05	MT06
Baseline	29.48	19.10
ENGWORD→ARALEM	29.47	18.90
ENGWORD→ARALEM+DET	30.13	19.36
ENGWORD→ARALEM+DET with GEN+NUM Prediction	30.43	19.65

Table 7: Results comparing our baselines and best performing setup on MT05 and MT06 (blind). Results are in (% BLEU).

5.5 Scaling Up

We performed experiments using a larger amount of data (15 times the size of the original dataset; also available from the LDC). Not surprisingly, the effect of our approach diminished. Although the general trends remained the same, none of the alternative settings was able to beat the baseline. We compared the percentage of the Exact Match, Lemma Match and Unmatchable words with the reference of the basic and scaled up systems. We found out that the percentage of exact matches increases while the percentage of unmatched words decreases. This is not a surprising result of using more data. The lemma match percentage decreases across the different systems. This suggests that our approach is more effective for conditions with low and medium resource size.

5.6 Discussion

The generation of fully inflected forms from uninflected lemmas (Table 5) in a purely monolingual setting such as our morphological generation step is very hard – we get only 82.2% BLEU starting with gold lemmas. Adding different combinations of gold values of the three most problematic morphological features improves the score by over

12% absolute BLEU to a higher performance ceiling (94.8% BLEU).

Automatically modeling these features at a high accuracy for SMT, however, turns out to be rather hard. If we consider using them as part of the translation step together with lemmas, we find that they almost always hurt the end-to-end (translation-generation) MT system except for the DET feature which improves over an inflected tokenized baseline by about 0.6% BLEU.

Predicting the feature values using an independent supervised learning step that has access to the English word, POS and syntax features produces accuracy scores ranging in mid to high 80s%. Comparing the prediction accuracy of GEN, NUM and DET (Table 3), we find NUM is the easiest to predict, followed by DET and then GEN. This makes sense given the information provided from English, which is inflected for NUM, but not GEN.

The results in Table 6 show that DET, as a single feature, helps more when it is part of the translation step (30.13 BLEU) compared to being predicted (29.67~29.80). In both cases, it fares better than simply leaving determining DET to the generation step (29.47).

Neither GEN nor NUM, as single features, help much (or at all) over the baselines when part of the translation step or when predicted. However, when both are combined with DET they consistently help only when GEN and NUM are predicted, not translated. It is possible that the lower performance we see as part of the translation is a product of how we translate: we do not factor these features in the translation – a direction we plan to consider in the future. We postulate that the prediction step helps because it has access to more information than used in our translation step, e.g., source language syntax.

6 Conclusions and Future Work

We compared three methods of modeling morphological features in SMT from English to Arabic: as part of core lexical translation, as part of morphological generation and using an independent morphological prediction component. The best configuration for the three most problematic morphological features for English-Arabic SMT models the determiner as part of core translation and favors predicting gender and number features separately from generation. Our approach shows improvements on a medium-size training data set but when using a very large data set the advantage of using morphological modeling disappears.

In the future, we plan to identify the best configuration for other morphological features in Arabic. We also plan to apply our approach to other target languages such as Persian and Hebrew. We will also investigate how the features we studied here can be used in a more elegant joint model such as Factored MT.

References

- Al-Haj, Hassan and Alon Lavie. 2010. The Impact of Arabic Morphological Segmentation on Broad-coverage English-to-Arabic Statistical Machine Translation. In *Proc. of AMTA'10*, Denver, CO.
- Alkuhlani, Sarah and Nizar Habash. 2011. A Corpus for Modeling Morpho-Syntactic Agreement in Arabic: Gender, Number and Rationality. In *Proc. of ACL'11*, Portland, OR.
- Badr, Ibrahim, Rabih Zbib, and James Glass. 2008. Segmentation for English-to-Arabic Statistical Machine Translation. In *Proc. of ACL'08*, Columbus, OH.
- Callison-Burch, Chris, Miles Osborne, and Philipp Koehn. 2006. Re-evaluating the Role of BLEU in Machine Translation Research. In *Proc. of EACL'06*, Trento, Italy.
- Clifton, Ann and Anoop Sarkar. 2011. Combining morpheme-based machine translation with post-processing morpheme prediction. In *Proc. of ACL'11*, Portland, OR.
- El Kholy, Ahmed and Nizar Habash. 2010a. Orthographic and Morphological Processing for English-Arabic Statistical Machine Translation. In *Proc. of TALN'10*, Montréal, Canada.
- El Kholy, Ahmed and Nizar Habash. 2010b. Techniques for Arabic Morphological Detokenization and Orthographic Denormalization. In *Proc. of LREC'10*, Valletta, Malta.
- El Kholy, A. and N. Habash. 2011. Automatic Error Analysis for Morphologically Rich Languages. In *Proc. of MT Summit XIII*, Xiamen, China
- El Kholy, Ahmed and Nizar Habash. 2012. Rich Morphology Generation Using Statistical Machine Translation. In *Proc. of INLG'12*, Utica, IL.
- Elming, Jakob and Nizar Habash. 2009. Syntactic Reordering for English-Arabic Phrase-Based Machine Translation. In *Proc. of EACL'09*, Athens, Greece.
- Habash, Nizar and Owen Rambow. 2005. Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In *Proc. of ACL'05*, Ann Arbor, MI.
- Habash, Nizar and Fatiha Sadat. 2006. Arabic Preprocessing Schemes for Statistical Machine Translation. In *Proc. of NAACL06*, New York, NY.
- Habash, Nizar. 2010. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.
- Klein, Dan and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proc. of ACL'03*.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Christopher Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Christopher Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proc. of ACL'07*, Prague, Czech Republic.
- Koehn, Philipp. 2004. Statistical significance tests for machine translation evaluation. In *Proc. of EMNLP'04*, Barcelona, Spain.
- Lafferty, J., A. McCallum, and F.C.N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of the International Conference on Machine Learning*.
- Maamouri, Mohamed, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *NEMLAR Conference on Arabic Language Resources and Tools*, Cairo, Egypt.
- Minkov, Einat, Kristina Toutanova, and Hisami Suzuki. 2007. Generating complex morphology for machine translation. In *Proc. of ACL'07*, Prague, Czech Republic.
- Och, Franz Josef and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–52.
- Oflazer, Kemal and Ilknur Durgar El-Kahlout. 2007. Exploring different representational units in English-to-Turkish statistical machine translation. In *Proc. of ACL'07*.
- Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proc. of ACL'02*, Philadelphia, PA.
- Sarikaya, Ruhi and Yonggang Deng. 2007. Joint morphological-lexical language modeling for machine translation. In *Proc. of NAACL07*, Rochester, NY.
- Stolcke, Andreas. 2002. SRILM - an Extensible Language Modeling Toolkit. In *Proc. of ICSLP'02*, Denver, CO.
- Toutanova, Kristina, Hisami Suzuki, and Achim Ruopp. 2008. Applying morphology generation models to machine translation. In *Proc. of ACL'08*, Columbus, OH.
- Yeniterzi, Reyhan and Kemal Oflazer. 2010. Syntax-to-morphology mapping in factored phrase-based statistical machine translation from english to turkish. In *Proc. of ACL'10*, Uppsala, Sweden.

Exploiting Shared Chinese Characters in Chinese Word Segmentation Optimization for Chinese-Japanese Machine Translation

Chenhui Chu, Toshiaki Nakazawa, Daisuke Kawahara, Sadao Kurohashi

Graduate School of Informatics, Kyoto University

Yoshida-honmachi, Sakyo-ku

Kyoto, 606-8501, Japan

{chu, nakazawa}@nlp.ist.i.kyoto-u.ac.jp {dk, kuro}@i.kyoto-u.ac.jp

Abstract

Unknown words and word segmentation granularity are two main problems in Chinese word segmentation for Chinese-Japanese Machine Translation (MT). In this paper, we propose an approach of exploiting common Chinese characters shared between Chinese and Japanese in Chinese word segmentation optimization for MT aiming to solve these problems. We augment the system dictionary of a Chinese segmenter by extracting Chinese lexicons from a parallel training corpus. In addition, we adjust the granularity of the training data for the Chinese segmenter to that of Japanese. Experimental results of Chinese-Japanese MT on a phrase-based SMT system show that our approach improves MT performance significantly.

1 Introduction

As there are no explicit word boundary markers in Chinese, word segmentation is considered as an important first step in MT. Studies showed that a MT system with Chinese word segmentation outperforms the one treating each Chinese character as a single word, and the quality of Chinese word segmentation affects the MT performance (Xu et al., 2004; Chang et al., 2008). It has been found that besides segmentation accuracy, segmentation consistency and granularity of Chinese words are also important for MT (Chang et al., 2008). Moreover, optimal Chinese word segmentation for MT is dependent on the other language, therefore, a bilingual approach is necessary (Ma and Way, 2009).

Zh: 小坂先生是日本临床麻醉学会的创始人。
Ja: 小坂先生は日本臨床麻醉学会の創始者である。
Ref: Mr. Kosaka is the founder of The Japan Society for Clinical Anesthesiologists.

Figure 1: Example of Chinese word segmentation problems in Chinese-Japanese MT.

Most studies focus on language pairs between Chinese and other languages that have white spaces between words (e.g. English). We focus on Chinese-Japanese MT, where segmentation is needed for both sides. Segmentation for Japanese successfully achieves F-score nearly 99% (Kudo et al., 2004), while that for Chinese is still about 95% (Wang et al., 2011). Therefore, we only do word segmentation optimization for Chinese, and keep the Japanese segmentation results.

Similar to the previous works, we think the following two problems of Chinese word segmentation are important for Chinese-Japanese MT. The first problem is unknown words, which is the major difficulty faced by a Chinese segmenter affecting segmentation accuracy and consistency. Taking “Kosaka” in Figure 1 as an example, which is a proper noun in Japanese. Because “Kosaka” is a unknown word for the Chinese segmenter, it is mistakenly segmented into two tokens, while the Japanese word segmentation result is correct.

The second problem is word segmentation granularity. Most Chinese segmenters adapt the famous Penn Chinese Treebank (CTB) standard (Xia et al., 2000), while most Japanese segmenters adapt a shorter unit standard. Therefore, the segmentation unit in Chinese may be longer than Japanese even for the same concept. This can increase the number of 1-to-n alignments which makes the word alignment task more difficult. Taking “founder”

Meaning	snow	love	begin
TC	雪(U+96EA)	愛(U+611B)	發(U+767C)
SC	雪(U+96EA)	爱(U+7231)	发(U+53D1)
Kanji	雪(U+96EA)	愛(U+611B)	発(U+767A)

Table 1: Examples of common Chinese characters (TC denotes Traditional Chinese and SC denotes Simplified Chinese).

in Figure 1 as an example, the Chinese segmenter recognizes it as one token, while the Japanese segmenter splits it into two tokens because of the different word segmentation standards.

To solve the above problems, we propose an approach based on a bilingual perspective, and exploit common Chinese characters shared between Chinese and Japanese in Chinese word segmentation optimization for MT. We extract Chinese lexicons from a parallel training corpus based on common Chinese characters to augment the system dictionary of a Chinese segmenter. In addition, we adjust the granularity of the training data for the Chinese segmenter to that of Japanese by means of extracted Chinese lexicons. We conducted experiments on Chinese-Japanese MT tasks using a phrase-based SMT system, and experimental results indicate that our approach can improve MT performance significantly.

2 Common Chinese Characters

Different from other language pairs, Chinese and Japanese share Chinese characters. In Chinese the Chinese characters are called Hanzi, while in Japanese they are called Kanji. Hanzi can be divided into two groups, Simplified Chinese (used in mainland China and Singapore) and Traditional Chinese (used in Taiwan, Hong Kong and Macao). The number of strokes needed to write characters has been largely reduced in Simplified Chinese, and the shapes may be different from the ones in Traditional Chinese. Because Kanji characters originated from ancient China, many common Chinese characters exist between Hanzi and Kanji. Table 1 gives some examples of common Chinese characters in Traditional Chinese, Simplified Chinese and Japanese with their Unicode.

Chinese characters contain significant semantic information, and common Chinese characters share the same meaning, so they can be valuable linguistic clues for many Chinese-Japanese NLP tasks. Many studies have been done to exploit common Chinese characters. Tan et al. (1995)

used the occurrence of identical common Chinese characters (e.g. “snow” in Table 1) in automatic sentence alignment task. Goh et al. (2005) detected common Chinese characters where Kanji are identical to Traditional Chinese but different from Simplified Chinese (e.g. “love” in Table 1). They used Chinese encoding converter¹ which can convert Traditional Chinese into Simplified Chinese, and built a Japanese-Simplified Chinese dictionary. Chu et al. (2011) made use of the Unihan database² to detect common Chinese characters which are visual variants of each other (e.g. “begin” in Table 1), and proved the effectiveness of common Chinese characters in Chinese-Japanese phrase alignment. In this paper, we focus on Simplified Chinese-Japanese MT and exploit common Chinese characters in Chinese word segmentation optimization.

3 Chinese Word Segmentation Optimization

3.1 Chinese Lexicons Extraction

We extract Chinese lexicons from a parallel training corpus through the following steps:

- Step 1: Segment Chinese and Japanese sentences in the parallel training corpus.
- Step 2: Convert Japanese tokens which are made up of Kanji only³ into Simplified Chinese using the Kanji to Hanzi conversion method described in (Chu et al., 2011).
- Step 3: Extract the converted tokens as Chinese lexicons if they exist in the corresponding Chinese sentence. Here, we propose two extraction strategies:
 - Strategy 1: Only extract tokens which have a different word boundary in the segmented Chinese sentence.
 - Strategy 2: Extract all tokens.

For example, using Strategy 1, “小坂(Kosaka)”, “创始(found)” and “者(person)” in Figure 1 are extracted, but using Strategy 2, “先生(Mr.)”, “日本(Japan)”, “临床(clinical)”, “麻醉(anesthesia)” and “学会(society)” are also extracted. Note that although “创始↔創始(found)”, “临床↔臨

¹<http://www.mandarintools.com/zhcode.html>

²<http://unicode.org/charts/unihan.html>

³Japanese has several kinds of character types other than Kanji.

CTB	JUMAN
AD	副詞(adverb)
CC	接統詞(conjunction)
CD	名詞(noun)[数詞(numeral noun)]
FW	未定義語(undefined word)[アルファベット(alphabet)]
IJ	感動詞(interjection)
M	接尾辞(suffix)[名詞性名詞助数辞(measure word suffix)]
NN	名詞(noun)[普通名詞(common noun)/サ変名詞(sahen noun)/形式名詞(formal noun)/副詞的名詞(adverbial noun), 接尾辞(suffix)[名詞性名詞接尾辞(noun suffix)/名詞性特殊接尾辞(special noun suffix)]
NR	名詞(noun)[固有名詞(proper noun)/地名(place name)/人名(person name)/組織名(organization name)]
NT	名詞(noun)[時相名詞(temporal noun)]
PU	特殊(special word)
VA	形容詞(adjective)
VV	動詞(verb)/名詞(noun)[サ変名詞(sahen noun)]

Table 2: Chinese-Japanese POS tags mapping table.

床(clinical)” and “麻醉↔麻醉(anesthesia)” are not identical, because “創↔創(create)”, “臨↔臨(arrive)” and “醉↔醉(drunk)” are common Chinese characters, “創始(found)” is converted into “创始(found)”, “臨床(clinical)” is converted into “临床(clinical)” and “麻醉(anesthesia)” is converted into “麻醉(anesthesia)” in Step 2.

In preliminary experiments, we extracted 14,359 lexicons using Strategy 1, and 18,584 lexicons using Strategy 2 from a paper abstract parallel corpus containing 680K sentence pairs.

3.2 Chinese Lexicons Incorporation

Several studies showed that using a system dictionary is helpful for Chinese word segmentation (Low et al., 2005; Wang et al., 2011). Therefore, we use a corpus-based Chinese word segmentation and POS tagging tool with a system dictionary. We incorporate the extracted lexicons into the system dictionary. The extracted lexicons are not only effective for the unknown word problem, but also helpful to solve the word segmentation granularity problem.

However, setting POS tags for the extracted lexicons is problematic. To solve this problem, we made a POS tags mapping table between Chinese and Japanese by hand. For Chinese, we use the POS tagset used in CTB which is also used in our Chinese segmenter. For Japanese, we use the POS tagset defined in the morphological analyzer JUMAN (Kurohashi et al., 1994). JUMAN adapts a POS tagset containing sub POS tags. For example, the POS tag “名詞(noun)” contains sub POS

tags such as “普通名詞(common noun)”, “固有名詞(proper noun)”, “時相名詞(temporal noun)” etc. Table 2 shows a part of the Chinese-Japanese POS tags mapping table we made, the sub POS tags of JUMAN are written inside of the brackets.

We assign POS tags for the extracted Chinese lexicons by converting the POS tags of Japanese tokens assigned by JUMAN into POS tags of CTB. Note that not all POS tags of JUMAN can be converted into POS tags of CTB, and vice versa. For the ones that cannot be converted, we do not incorporate them into the system dictionary. In preliminary experiments, 294 lexicons in Strategy 1 and 1,581 lexicons in Strategy 2 were discarded.

3.3 Short Unit Transformation

Bai et al. (2008) showed that adjusting Chinese word segmentation to make tokens 1-to-1 mapping as many as possible between a parallel sentences can improve alignment accuracy which is crucial for corpus-based MT. Wang et al. (2010) proposed a short unit standard for Chinese word segmentation that is more similar to the Japanese word segmentation standard, which can reduce the number of 1-to-n alignments and improve MT performance.

Here, we propose a method to transform the annotated training data of Chinese segmenter into Japanese word segmentation standard using the extracted Chinese lexicons, and use the transformed data for training the Chinese segmenter. Because the extracted lexicons are derived from Japanese word segmentation results, they follow Japanese

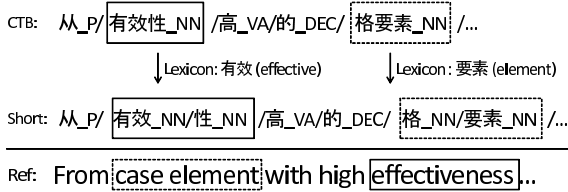


Figure 2: Example of short unit transformation.

word segmentation standard. Therefore, we utilize these lexicons for short unit transformation. We use Chinese lexicons extracted by Strategy 2 described in Section 3.1 and modify every token in the training data for the Chinese segmenter. If the token is longer than a extracted lexicon, we simply split it. Figure 2 gives an example of this process, where “有效(effective)” and “要素(element)” are both extracted lexicons. Because “有效性(effectiveness)” is longer than “有效(effective)”, it is split into “有效(effective)” and “性” (a noun suffix), and “格要素(case element)” is longer than “要素(element)”, it is split into “格(case)” and “要素(element)”. For POS tags, we keep the originally annotated one for the split tokens.

We do not use extracted lexicons that are composed of only one Chinese character, because these lexicons may lead to undesirable transformation results. Taking the Chinese character “歌(song)” as an example, “歌(song)” can be used as a single word, but we also can use “歌(song)” to construct other words by combining it with other Chinese characters, such as “歌颂(praise)”, “诗歌(poem)” etc. Obviously, splitting “歌颂(praise)” into “歌(song)” and “颂(eulogy)”, or splitting “诗歌(poem)” into “诗(poem)” and “歌(song)” is undesirable. Also, there are few consecutive tokens in the training data that can be combined to one extracted lexicon, we do not consider this pattern.

4 Experiments

We conducted Chinese-Japanese translation experiments to show the effectiveness of exploiting common Chinese characters in Chinese word segmentation optimization.

4.1 Settings

4.1.1 Parallel Training Corpus

The parallel training corpus we used is a paper abstract corpus provided by JST⁴ and NICT⁵. This

⁴<http://www.jst.go.jp>

⁵<http://www.nict.go.jp/>

	Ja	Zh
# sentences	680k	
# words	21.8M	18.2M
# Chinese characters	14.0M	24.2M
average sentence length	32.9	22.7

Table 3: Statistics of Chinese-Japanese training corpus.

corpus was created by the Japanese project “Development and Research of Chinese-Japanese Natural Language Processing Technology”. The statistics of this corpora are shown in Table 3.

4.1.2 Chinese Annotated Corpus

We used two types of manually annotated Chinese corpus for training the Chinese segmenter. One is NICT Chinese Treebank, which is from the same domain as the parallel training corpus and contains 9,792 sentences. Note that the annotated sentences in this corpus are not included in the parallel training corpus. The other corpus is CTB 7 (LDC2010T07)⁶. We made the training data from CTB 7 using the same method described in (Wang et al., 2011), and it contains 31,131 sentences.

4.1.3 Chinese and Japanese Segmenters

For Chinese, we used a corpus-based word segmentation and POS tagging tool with a system dictionary, weights for the lexicons in the system dictionary are automatically learned from the training data using averaged structured perceptron (Collins, 2002). For Japanese, we used JUMAN (Kurohashi et al., 1994).

4.1.4 SMT Model

We used the state-of-the-art phrase-based SMT toolkit Moses (Koehn et al., 2007) with default options, except for the distortion limit (6→20). It was tuned by MERT using another 500 development sentence pairs.

4.1.5 Test Sets

We translated 5 test sets of Chinese sentences from the same domain as the parallel training corpus. The statistics of the test sets are shown in Table 4. Note that all sentences in the test sets are not included in the parallel training corpus.

4.2 Results and Evaluation

We conducted Chinese-Japanese translation experiments on NICT Chinese Treebank and CTB 7,

⁶<http://www ldc.upenn.edu/>

	T1	T2	T3	T4	T5
# sentences	255	336	391	395	393
# words	6.5K	8.7K	10.0K	11.7K	16.5K
# CC	8.6K	10.6K	12.9K	15.8K	22.0K
avg. sen. len.	44.9	47.0	45.4	52.2	74.1

Table 4: Statistics of test sets (T denotes test set and CC denotes Chinese characters).

comparing the following four experimental settings:

- Baseline: Only using the lexicons extracted from Chinese annotated corpus as the system dictionary for the Chinese segmenter.
- Strategy 1: Incorporate the Chinese lexicons extracted by Strategy 1 described in Section 3.1 into the system dictionary.
- Strategy 2: Incorporate the Chinese lexicons extracted by Strategy 2 described in Section 3.1 into the system dictionary.
- Short unit: Incorporate the Chinese lexicons extracted by Strategy 2 into the system dictionary and train the Chinese segmenter on the short unit training data transformed in Section 3.3.

Table 5 shows the BLEU scores for Chinese-to-Japanese translation using NICT Chinese Treebank. Short unit achieved best MT performance. The extracted Chinese lexicons also improved BLEU scores significantly. Besides test set 2 and test set 5, Strategy 2 achieved better improvement than Strategy 1. We think the reason is that Strategy 2 extracted more lexicons which is helpful to solve the unknown word problem.

Table 6 shows the BLEU scores for Chinese-to-Japanese translation using CTB 7. Although Strategy 2 obtained higher BLEU scores than the baseline, compared to Strategy 1, the improvement is not significant. We investigated the reason and found that there are many overlaps between lexicons extracted from the parallel training corpus and lexicons extracted from the annotated training data. For example, “蛋白质(protein)” was extracted from the annotated training data and overlaps “蛋白(protein)” and “质(quality)” extracted from the parallel training corpus. When the Chinese segmenter tries to segment “蛋白质(protein)”, the overlap can lead to inconsistent segmentation results. Although more extracted Chinese lexicons

BLEU	T1	T2	T3	T4	T5
baseline	48.86	47.09	37.18	27.21	24.29
strategy 1	50.41	48.22	39.25	28.33	26.44
strategy 2	50.77	47.96	39.83	28.54	26.29
short unit	52.04	49.55	39.96	28.73	26.63

Table 5: Results of Chinese-to-Japanese translation experiments using NICT Chinese Treebank.

BLEU	T1	T2	T3	T4	T5
baseline	51.03	48.98	40.52	29.20	26.08
strategy 1	52.42	51.78	41.20	30.61	28.20
strategy 2	51.53	50.47	41.30	29.57	26.77
short unit	52.83	51.13	41.57	31.01	28.82

Table 6: Results of Chinese-to-Japanese translation experiments using CTB 7.

is more helpful to solve the unknown word problem, it also leads to more overlaps. Because Strategy 2 extracted more lexicons than Strategy 1, more overlaps are also produced. We investigated the number of overlaps. For CTB 7, the overlap number between Strategy 2 is 2,399, it greatly exceeds the number between Strategy 1 which is 1,388. While for NICT, the overlap number between Strategy 2 is 1,759, and between Strategy 1 is 1,694, the difference is not significant. In brief, there is a tradeoff between the unknown word problem and the overlap problem using our proposed method. However, by short unit transformation, the overlap problem can be solved. Taking the same example “蛋白质(protein)”, because it is split into “蛋白(protein)” and “质(quality)” in short unit transformation, overlaps will not exist any more. Therefore, short unit using CTB 7 also showed the best MT performance.

Comparing Table 5 with Table 6, we notice that the BLEU scores using NICT Chinese Treebank are lower than using CTB 7. We think the reason is the size of the training data. The number of annotated sentences in NICT Chinese Treebank is less than 1/3 of CTB 7. Therefore, less lexicons are extracted from NICT Chinese Treebank than CTB 7. The number of extracted lexicons from NICT Chinese Treebank is only 13,471, while from CTB 7 it is 26,202. Also, the weights for many lexicons extracted from the parallel training corpus can not be learned correctly using NICT Chinese Treebank as training data. However, short unit using NIC-

Input: 本/論文/中/, /提/議/考/慮/現/存/實/現/方/式/的/功/能/ /適/應/性/ /決/定/對/策/目/標/
的/保/密/基/本/設/計/法/。

Output: 本/論/文/で/は/, /提/案/す/る/適/應/的/ /對/策/を/決/定/す/る/セ/キュ/リ/テ/ィ/基
本/設/計/法/を/考/え/る/現/存/の/實/現/方/式/の/機/能/ /を/目/標/と/し/て/い/る/。

Short unit (BLEU=56.33)

Input: 本/論文/中/, /提/議/考/慮/現/存/實/現/方/式/的/功/能/ /適/應/性/ /決/定/對/策/目/標/
的/保/密/基/本/設/計/法/。

Output: 本/論/文/で/は/, /提/案/す/る/考/え/現/存/の/實/現/方/式/の/機/能/的/ /適/應/性/
 /を/決/定/す/る/對/策/目/標/の/セ/キュ/リ/テ/ィ/基/本/設/計/法/を/提/案/す/る/。

Reference

本/論/文/で/は/, /對/策/目/標/を/現/存/の/實/現/方/式/の/機/能/的/ /適/合/性/ /も/考/慮/
し/て/決/定/す/る/セ/キュ/リ/テ/ィ/基/本/設/計/法/を/提/案/す/る/。

(In this paper, we propose a basic security design method also consider
functional suitability of the existing implementation method for determining
countermeasures target.)

Figure 3: Example of translation improvement.

T Chinese Treebank still achieved even better MT performance than the baseline using CTB 7.

We also conducted Japanese-to-Chinese translation experiments. Results show that our proposed approach also can improve the MT performance. However, compared to Chinese-to-Japanese translation, the improvement is not significant. We think the reason is the input sentence. For Chinese-to-Japanese translation, the segmentation of input Chinese sentences has been optimized. While for Japanese-to-Chinese translation, our proposed approach does not change the segmentation results of input Japanese sentences.

4.3 Discussion

4.3.1 Changes in Vocabulary and Phrase Table Size

We compared the Chinese vocabulary and phrase table size changes before and after exploiting common Chinese characters in Chinese word segmentation optimization. Table 7 shows the comparison results using NICT Chinese Treebank and CTB 7. The decrease of Chinese vocabulary size after optimization indicates the improvement of Chinese segmentation consistency, while the increase of phrase table size after optimization means the increase of translation knowledge.

4.3.2 Short Unit Effectiveness

Experimental results indicate that our proposed approach can improve MT performance significantly, especially for short unit. We present one example to show the effectiveness of short unit.

	vocabulary		phrase table	
	NICT	CTB 7	NICT	CTB 7
baseline	653K	509K	848M	861M
strategy 1	523K	439K	859M	867M
strategy 2	527K	438K	858M	868M
short unit	461K	396K	881M	896M

Table 7: Comparison of vocabulary and phrase table size changes before and after optimization.

Figure 3 shows an example of translation improvement by short unit compared to the baseline. The difference between short unit and the baseline is whether “適應性(suitability)” is split in Chinese or not, while the Japanese segmenter splits it. By splitting it, short unit improves word alignment and phrase extraction which eventually effects the decoding process. In decoding, short unit treats “功能適應性(functional suitability)” as one phrase, while the baseline separates it leading to a undesirable translation result.

4.3.3 Short Unit Transformation Percentage

One encouraging result is that, although the Chinese lexicons used for short unit transformation were extracted from a paper abstract domain corpus which is not the same domain that CTB 7 belongs to, short unit still achieved significant MT performance improvement using CTB 7. To identify the reason, we investigated the percentage of transformed tokens. In NICT Chinese Treebank, there are 6,623 tokens out of 257,825 been transformed to 13,469 short unit tokens, the percentage is about 2.57%. In CTB 7, there are 19,983 token-

s out of 718,716 been transformed to 41,336 short unit tokens, the percentage is about 2.78%. This result shows the strength of our proposed short unit transformation method. Although the lexicons used for short unit transformation are extracted from a paper abstract domain, these lexicons also work well for short unit transformation on Chinese annotated corpus of other domains (i.e. CTB 7).

4.3.4 Short Unit Transformation Problems

Furthermore, we investigated the details of the transformed tokens. Based on our manual investigation, over 90% of the transformed results are correct. However, some transformation problems still exist. One problem is transformation ambiguity. We present one example to show this kind of problem. There is a long token “充电器(charger)” in the annotated training data, and a lexicon “电器(electric equipment)” extracted from the parallel training corpus, so the long token is split into “充(charge)” and “器(electric equipment)”, which is undesirable. However, we found that a extracted lexicon “充电(charge)” also exists and using this lexicon the long token can be split into “充电(charge)” and “器(device)” successfully. We think this kind of ambiguity can be solved using a statistical method.

Another problem is POS tag assignment for the transformed short unit tokens. Our proposed method simply keep the originally annotated POS tag of the long token for the transformed short unit tokens, it works well in most cases. However, there are also some exceptions. For example, there is a long token “被实验者(test subject)” in the annotated training data, and a lexicon “实验(test)” extracted from the parallel training corpus, so the long token is split into “被(be)”, “实验(test)” and “者(person)”. As the POS tag for the original long token is NN, the POS tags for the transformed short unit tokens are all assigned to NN, which is undesirable for “被(be)”. The correct POS tag for “被(be)” should be LB. We think a external dictionary would be helpful to solve this problem. Furthermore, the transformed short unit tokens may have more than one possible POS tags. All these problems are future work of this study.

5 Related Work

Exploiting lexicons from external resources (Peng et al., 2004; Chang et al., 2008) is a way to deal with the unknown word problem. However, the external lexicons may not be very efficient for a

specific domain. Some studies (Xu et al., 2004; Ma and Way, 2009) used a method of learning a domain specific dictionary from the character-based alignment results of a parallel training corpus, which separate every Chinese character, and consider consecutive Chinese characters as a lexicon in n-to-1 alignment results. Our proposed method differs from previous studies, we obtain a domain specific dictionary by extracting Chinese lexicons directly from a segmented parallel training corpus, making word alignment is unnecessary.

The goal of our proposed short unit transformation method is to make the segmentation results of Chinese and Japanese a 1-to-1 mapping, which can improve alignment accuracy and MT performance. Bai et al. (2008) proposed a method of learning affix rules from a aligned Chinese-English bilingual terminology bank to adjust Chinese word segmentation in the parallel corpus directly aiming to achieve the same goal. Our proposed method does not adjust Chinese word segmentation directly. Instead, we utilize the extracted Chinese lexicons to transform the annotated training data of a Chinese segmenter into short unit standard, and do segmentation using the retrained Chinese segmenter.

Wang et al. (2010) also proposed a short unit transformation method. The proposed method is based on transfer rules and a transfer database. The transfer rules are extracted from alignment results of annotated Chinese and segmented Japanese training data. The transfer database is constructed using external lexicons, and is manually modified. Our proposed method learns transfer knowledge based on common Chinese characters. Moreover, we do not use external lexicons, and manual work is not needed.

6 Conclusions

In this paper, we pointed out two main problems in Chinese word segmentation for Chinese-Japanese MT, namely unknown words and word segmentation granularity. To solve the problems, we proposed an approach of exploiting common Chinese characters shared in Chinese and Japanese. Common Chinese characters have been successfully exploited in many Chinese-Japanese NLP tasks, we exploited them in Chinese word segmentation optimization for MT in this study. Experimental results of Chinese-Japanese MT on a phrase-based SMT system indicated that our approach can improve MT performance significantly.

However, there are still some problems in our proposed short unit transformation method. We plan to solve these problems to further improve MT performance. Furthermore, we only evaluated our proposed approach on a parallel corpus from abstract paper domain, where Chinese characters are more frequently used than general domains in Japanese. In the future, we plan to evaluate the proposed approach on parallel corpus of other domains.

References

- Bai, Ming-Hong, Keh-Jiann Chen, and Jason S.Chang. 2008. Improving word alignment by adjusting chinese word segmentation. In *Proceedings of the Third International Joint Conference on Natural Language Processing*, pages 249–256, Hyderabad, India, January. Association for Computational Linguistics.
- Chang, Pi-Chuan, Michel Galley, and Christopher D. Manning. 2008. Optimizing Chinese word segmentation for machine translation performance. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 224–232, Columbus, Ohio, June. Association for Computational Linguistics.
- Chu, Chenhui, Toshiaki Nakazawa, and Sadao Kurohashi. 2011. Japanese-chinese phrase alignment using common chinese characters information. In *Proceedings of MT Summit XIII*, pages 475–482, Xiamen, China, September.
- Collins, Michael. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 1–8. Association for Computational Linguistics, July.
- Goh, Chooi-Ling, Masayuki Asahara, and Yuji Matsumoto. 2005. Building a Japanese-Chinese dictionary using kanji/hanzi conversion. In *Proceedings of the International Joint Conference on Natural Language Processing*, pages 670–681.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June. Association for Computational Linguistics.
- Kudo, Taku, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying conditional random fields to japanese morphological analysis. In Lin, Dekang and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 230–237, Barcelona, Spain, July. Association for Computational Linguistics.
- Kurohashi, Sadao, Toshihisa Nakamura, Yuji Matsumoto, and Makoto Nagao. 1994. Improvements of Japanese morphological analyzer JUMAN. In *Proceedings of the International Workshop on Sharable Natural Language*, pages 22–28.
- Low, Jin Kiat, Hwee Tou Ng, and Wenyuan Guo. 2005. A maximum entropy approach to chinese word segmentation. In *Proceedings of the 4th SIGHAN Workshop on Chinese Language Processing (SIGHAN05)*, pages 161–164.
- Ma, Yanjun and Andy Way. 2009. Bilingually motivated domain-adapted word segmentation for statistical machine translation. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 549–557, Athens, Greece, March. Association for Computational Linguistics.
- Peng, Fuchun, Fangfang Feng, and Andrew McCallum. 2004. Chinese segmentation and new word detection using conditional random fields. In *Proceedings of Coling 2004*, pages 562–568, Geneva, Switzerland, Aug 23–Aug 27. COLING.
- Tan, Chew Lim and Makoto Nagao. 1995. Automatic alignment of Japanese-Chinese bilingual texts. *IE-ICE Transactions on Information and Systems*, E78-D(1):68–76.
- Wang, Yiou, Kiyotaka Uchimoto, Junichi Kazama, Canasai Kruengkrai, and Kentaro Torisawa. 2010. Adapting chinese word segmentation for machine translation based on short units. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may.
- Wang, Yiou, Jun'ichi Kazama, Yoshimasa Tsuruoka, Wenliang Chen, Yujie Zhang, and Kentaro Torisawa. 2011. Improving chinese word segmentation and pos tagging with semi-supervised methods using large auto-analyzed data. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 309–317, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.
- Xia, Fei, Martha Palmer and Nianwen Xue, Mary Ellen Okurowski, John Kovarik, Fu dong Chiou, and Shizhe Huang. 2000. Developing guidelines and ensuring consistency for chinese text annotation. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation*, Athens, Greece.
- Xu, Jia, Richard Zens, and Hermann Ney. 2004. Do we need chinese word segmentation for statistical machine translation? In Streiter, Oliver and Qin Lu, editors, *ACL SIGHAN Workshop 2004*, pages 122–128, Barcelona, Spain, July. Association for Computational Linguistics.

Hebrew Morphological Preprocessing for Statistical Machine Translation

Nimesh Singh and **Nizar Habash**
Center for Computational Learning Systems
Columbia University
nks2118@columbia.edu
habash@ccls.columbia.edu

Abstract

This paper presents a range of preprocessing solutions for Hebrew-English statistical machine translation. Our best system, using a morphological analyzer, increases 3.5 BLEU points over a no-tokenization baseline on a blind test set. The next best system uses Morfessor, an unsupervised morphological segmenter, and obtains almost 3.0 BLEU points over the baseline.

1 Introduction

Much research in statistical machine translation (SMT) has shown the importance of morphological preprocessing (aka, tokenization, segmentation) on translation quality. The common wisdom in the field is that such preprocessing helps, especially for morphologically rich languages, such as Arabic, Spanish or Finnish, because it reduces model sparsity and increases source-target symmetry (particularly when the target is morphologically poor, as in English). However, the value of preprocessing generally decreases with added training data, and is highly dependent on the language pair and particular preprocessing approach (Popović and Ney, 2004; Lee, 2004; Goldwater and McClosky, 2005; Habash and Sadat, 2006; Fishel and Kirik, 2010; Al-Haj and Lavie, 2012).

In this paper, we present results from a set of experiments to determine an optimal preprocessing method for Hebrew-English SMT, a language pair with limited previously published work (Lavie et al., 2004; Lembersky et al., 2011). We report on three types of preprocessing techniques using deterministic regular-expressions, unsupervised morphology learning, and morphological analysis and

disambiguation. Our results show that using a morphological analyzer helps translation quality the most, followed by using an unsupervised morphological segmenter.

The paper is structured as follows: Section 2 presents relevant related work. Section 3 discusses the linguistic challenges of translating Hebrew to English. Section 4 describes the different preprocessing techniques we study. And Section 5 presents our evaluation results.

2 Related Work

A wide range of preprocessing techniques have been studied for a variety of language pairs requiring different treatments. Nießen and Ney (2004) studied the impact of various types of morpho-syntactic restructuring on German-English SMT and Popović and Ney (2004) studied the effect of splitting words into stems and suffixes on SMT into English from Spanish, Catalan and Serbian. Their results show significant error reduction when stemming is used. Koehn and Knight (2003) compared different methods for compound splitting when translating from German to English. All of their methods improve SMT quality over a no-splitting baseline; however, the methods with the highest accuracy are not the best SMT performers. Lee (2004) investigated the use of automatic alignment of POS tagged English and affix-stem segmented Arabic to determine whether affixes should be kept separate, deleted or reattached to stems. Her results show that morphological preprocessing helps, but only for the smaller corpora sizes she investigated. As size increases, the benefits diminish. Goldwater and McClosky (2005) showed that incorporating various methods for specifying morphological information in Czech-English SMT (e.g., lemmatization and different styles of seg-

mentation) improves translation quality especially when the different methods are combined. Habash and Sadat (2006) compared a variety of what they called tokenization schemes and techniques for Arabic-English SMT. Their work and that of Lee (2004) are especially relevant since Arabic is a Semitic language like Hebrew. This paper is closest in its approach to Habash and Sadat (2006). We refer to their work further below. We do not discuss efforts on translation into morphologically rich languages although similar approaches have been investigated (El Kholly and Habash, 2012a; Al-Haj and Lavie, 2012)

As for the use of unsupervised morphology in SMT, Virpioja et al. (2007) and Fishel and Kirik (2010) presented some experiments with mixed results. They suggested that language pairs different from those they studied (Danish-Finnish-Swedish and Estonian-English, respectively), may benefit from unsupervised morphology. Snyder and Barzilay (2008) presented results on learning Hebrew morphology using parallel and monolingual resources.

Until recently, there has not been much parallel Hebrew-English data (Tsvetkov and Wintner, 2010), and consequently little work on Hebrew-English SMT. Lavie et al. (2004) built a transfer-based translation system for Hebrew-English and so did Shilon et al. (2012) for translation between Hebrew and Arabic. Lembersky et al. (2011), using the above-mentioned parallel corpus, compared the behavior of different SMT systems using training data sets that vary in reference translation directionality.

To our knowledge this is the first study comparing different tokenization techniques for Hebrew-English SMT. We successfully show that unsupervised morphology segmentation helps for Hebrew-English SMT, but a more linguistically sophisticated system with a morphological analyzer does best.

3 Hebrew in the Context of SMT

We present in this section some relevant Hebrew linguistic facts. This is followed by an analysis of out-of-vocabulary errors in the baseline system described in Section 5.

3.1 Hebrew Linguistic Facts

Hebrew poses computational processing challenges typical of Semitic languages such as Ara-

bic (Itai and Wintner, 2008; Shilon et al., 2012; Habash, 2010). Similar to Arabic, Hebrew orthography uses optional diacritics and its morphology uses both root-pattern and affixational mechanisms. Hebrew inflects for gender, number, person, state, tense and definiteness. Furthermore, Hebrew has a set of attachable clitics that are typically separate words in English, e.g., conjunctions (such as $+ו w+$ ‘and’),¹ prepositions (such as $+ב b+$ ‘in’), the definite article ($+ה h+$ ‘the’), or pronouns (such as $+הם hm$ ‘their’). These issues contribute to a high degree of ambiguity that is a challenge to translation from Hebrew to English or to any other language. Some of these clitics undergo morphotactic transformations that only add to the words’ ambiguity. For example, the sequence of the *preposition + article* $+ה+ב b+h+$ ‘in the’ results in the deletion of the letter for the article: $+ב b+$ ‘in the’. This makes the string $+ב b+$ ambiguous as ‘in a’ or ‘in the’.²

The different clitics appear in a generally strict order around the base word:

conjunction
relativizer
preposition
definite article
base word
pronominal clitic.

The definite article and the pronominal clitics do not co-occur. The conjunction $+ו w+$ ‘and’ and relativizer $+ש š+$ ‘that/who’ can appear with all parts-of-speech (nouns, verbs, prepositions, pronouns, etc.). Prepositions are mostly nominal and the definite article is strictly nominal.³ Pronominal clitics can attach to nouns and prepositions and infrequently to verbs (archaic).⁴ For example, the word *בשורה* $bšwrh$ has the following possible nominal analyses among others: *בשורה* $bšwrh$ ‘gospel’, *ב+שורה* $b+šwrh$ ‘in+(a/the) line’, and *ב+שורה+ה* $b+šwr+h$ ‘in her bull [lit. in+bull+her]’.

¹The following Hebrew 1-to-1 transliteration is used (in Hebrew lexicographic order): *abgdhwzxtiklmns’pcqršt*. All examples are undiacritized and final forms are not distinguished from non-final forms.

²The deleted article survives as a vowel which is written as an optional diacritic.

³Infinitive verbs in Hebrew have a prefix $+ל l+$ that can be considered a verbal particle ‘to’.

⁴Hebrew has an interrogative particle proclitic $+ה h+$ ‘is it true that ...?’ that is now archaic. The subordinating conjunction proclitic $+כש kš+$ ‘as, when’ can also attach to most words. Some prepositions can violate the order described above when they appear before the relativizer $+ש š+$, e.g., $+מ+ש m+š+$ ‘from that’. We do not handle these cases in our regular expression methods.

In this paper, we focus on the question of morphological segmentation of clitics in Hebrew words to make them easier to translate into English. We do not investigate deeper models of morphology that target lemmatization or inflectional features such as gender, number, and tense (El Kholly and Habash, 2012b).

3.2 Hebrew Out-of-Vocabulary Errors

The Out-of-Vocabulary (OOV) rate in our baseline development set is rather high: 7.0% of all tokens and almost 18% of all types. This is primarily due to the limited size of the parallel text we have access to (Tsvetkov and Wintner, 2010). The resource limitation is a good reason to consider morphological preprocessing given insights from previous published work (Lee, 2004; Habash and Sadat, 2006). We analyzed 10% of all the OOVs, a total of 80 cases from 40 sentences. Verbs are the most frequent part-of-speech (43%) followed by nouns (31%), adjectives (21%) and proper nouns (5%). The definite article $\text{+ה } h\text{+}$ appears in one-quarter of all cases, and the conjunction $\text{+ו } w\text{+}$ ‘and’ in one-fifth. Various prepositional clitics appear a total of 20% and the relativizer $\text{+ש } \text{š+}$ occurs in one-tenth of all cases. Only one case of a pronominal enclitic was in the sample studied (1.25%). About two-fifths of all cases do not involve any attached clitics (39%), almost one-half have one clitic (47%) and less than one-seventh have two (14%). About 60% of these cases can be potentially addressed by clitic tokenization.

4 Hebrew Preprocessing Techniques

We consider three preprocessing techniques: regular-expressions, unsupervised morphology learning (Creutz and Lagus, 2007), and morphological analysis and disambiguation (Adler, 2009).

4.1 Regular Expression Segmentation

In the first technique, we use simple regular expressions that deterministically segment the Hebrew word. We define four levels of segmentation schemes which we call S1, S2, S3, and S4. S1 splits off the conjunction $\text{+ו } w\text{+}$ ‘and’ and the relativizer $\text{+ש } \text{š+}$ ‘that/who’. S2 includes S1, and additionally splits off the preposition clitics $\text{+ב } b\text{+}$ ‘in/on’, $\text{+כ } k\text{+}$ ‘like/as’, $\text{+ל } l\text{+}$ ‘to/for’, and $\text{+מ } m\text{+}$ ‘from’. S3 includes S2, and additionally splits off $\text{+ה } h\text{+}$ ‘the’. Finally, S4 includes S3, and additionally splits off pronominal enclitics (unless the

definite article is present). The relative order of these components, which is discussed in Section 3, is strictly preserved. The clitics’ order and form are the only linguistic information utilized in this technique. These segmentation schemes are comparable to the tokenization schemes used by Habash and Sadat (2006) for Arabic: $S1 \approx D1$, $S2 \approx D2$, and $S4 \approx D3$. S3 is in between D2 and D3. To distinguish between schemes and techniques, we use REGEX-*scheme* to designate the regular expression techniques, e.g., REGEX-S1 is the regular expression technique targeting the S1 scheme.

The regular expressions directly apply these rules using no word-context information. As a result, this technique is very fast and is likely to make a lot of errors. Since the phrase-based SMT approach is robust to such segmentation errors (to a limit), we still expect this technique to help over the baseline.

4.2 Morfessor: Unsupervised Morphology

In the second technique, we use Morfessor (MORF), a state-of-the-art tool for unsupervised segmentation of words into morphemes (Creutz and Lagus, 2007). It is language independent, i.e., uses no linguistic knowledge. Instead, it creates a lexicon of morphs, such that the lexicon is both concise and can be used to build any word in the input. The conciseness is measured by combining a cost of the text based on its probability when represented by the morphemes in the lexicon with a cost based on the size of the lexicon. MORF then searches the space of segmentations to minimize that cost. It can be used in one of two modes, either learning a model directly from the input it is segmenting, or learning a model from one training set, and applying that segmentation model to an independent input set. In our experiment, we trained MORF on the word list of the combined training and tuning data sets, then applied that model to each data set, training, tuning, development, and test, in the second mode. We did not use additional monolingual data for training MORF in this paper although this is an interesting idea to study in the future. MORF is fairly quick, but slower than regular expressions. Similar to regular expressions, MORF does not use word-context, i.e., the segmentation is deterministic once a model is built. Furthermore, the produced segmentation is not guaranteed to be a well-defined tokenization scheme or

Base	Gloss	REGEX-S1	REGEX-S2	REGEX-S3	REGEX-S4	MORF	HTAG
להבדיל ✓ <i>lhbdyl</i>	to distinguish	להבדיל ✓ <i>lhbdyl</i>	ל+הבדיל <i>l+hbdl</i>	ל+ה+בדיל <i>l+h+bdyl</i>	ל+ה+בדיל <i>l+h+bdyl</i>	להבדיל ✓ <i>lhbdyl</i>	להבדיל ✓ <i>lhbdyl</i>
שליט ✓ <i>šlyT</i>	ruler	ש+ליט <i>š+lyT</i>	ש+ל+יט <i>š+l+yT</i>	ש+ל+יט <i>š+l+yT</i>	ש+ל+יט <i>š+l+yT</i>	שליט ✓ <i>šlyT</i>	שליט ✓ <i>šlyT</i>
השלום <i>hšlwm</i>	the peace	השלום <i>hšlwm</i>	השלום <i>hšlwm</i>	ה+שלום ✓ <i>h+šlwm</i>	ה+שלום ✓ <i>h+šlwm</i>	ה+שלום ✓ <i>h+šlwm</i>	ה+שלום ✓ <i>h+šlwm</i>
להלאים ✓ <i>lhlaym</i>	to nationalize	להלאים ✓ <i>lhlaym</i>	ל+הלאים <i>l+hlaym</i>	ל+ה+לאים <i>l+h+laym</i>	ל+ה+לאים <i>l+h+laym</i>	ל+ה+לאים <i>l+h+la+ym</i>	להלאים ✓ <i>lhlaym</i>
לאור <i>lawr</i>	in light of	לאור <i>lawr</i>	ל+אור ✓ <i>l+awr</i>	ל+אור ✓ <i>l+awr</i>	ל+אור ✓ <i>l+awr</i>	לאור <i>lawr</i>	ל+אור ✓ <i>l+awr</i>

Table 1: Word Segmentation Examples. Linguistically valid segmentations that are consistent with the gloss are marked with ✓.

	Token Increase	Similarity to Baseline	Accuracy	
			Gold-S4	Gold (Scheme)
REGEX-S1	113%	87.4%	70.1%	99.7% (S1)
REGEX-S2	141%	62.2%	65.3%	79.1% (S2)
REGEX-S3	163%	46.3%	68.2%	70.6% (S3)
REGEX-S4	190%	33.8%	54.5%	
MORF	124%	81.6%	72.9%	
HTAG	130%	71.8%	94.0%	
Gold-S4	136%	68.4%		

Table 2: Tokenization system statistics.

to be linguistically correct. These are clearly important limitations given what we know about Hebrew morphology.

4.3 Hebrew Morphological Analysis and Disambiguation

In the third technique, we use a Hebrew morphological tagger (HTAG) (Adler, 2009). The tagger uses a morphological analysis component (or dictionary) together with a disambiguation component trained in an unsupervised manner. The tokenization produced by this tool resembles the S4 scheme discussed above but is context sensitive. This technique is the most linguistically rich of the three techniques used. This results in the most accurate segmentation of words into true morphemes; however, it is the slowest of all the methods. We do not experiment with variations of the schemes based on the tagger’s choices as Habash and Sadat (2006) did for Arabic.

4.4 Comparing the Techniques

Table 1 presents some examples of the output of different techniques from our development set.

Linguistically correct (at least with regards to the chosen glosses) are indicated.

Table 2 presents three comparison angles contrasting the different techniques presented above. All statistics are computed over a 50-sentence sample consisting of 600 hand-annotated (gold reference) words from the development set. The gold annotations are in a linguistically correct S4 scheme (the maximally verbose scheme). The first column, labeled *Token Increase*, shows the ratio of the number of tokens in a particular scheme to the corresponding number in the baseline system (no tokenization). As expected, the ratio increases as the number of segmentation decisions increases, with REGEX-S4 having the highest ratio. MORF and HTAG have similar numbers and are in between REGEX-S1 and REGEX-S2. The general trends in the full development set are consistent with the studied sample except that the ratios are around 4% lower on average.

The second column presents similarity to the no-tokenization baseline, or in other words, the percentage of unchanged words in the input. As expected REGEX-S1 and REGEX-S4 are the

least and most aggressive techniques, respectively. MORF is not as aggressive as HTAG.

The last two columns list the accuracy of the tokenization techniques against the gold annotation in S4 scheme as well as against a matching scheme converted from the human annotation to match the appropriate less verbose schemes (S1, S2 and S3). REGEX-S1 is highly accurate (99.7%) in its limited decisions. But HTAG has the best accuracy on the most verbose scheme (S4). The worst accuracy is for REGEX-S4. It is hard to judge MORF since it is not necessarily intended to match an S4 scheme, but we provide the number for comparison reasons. In close inspection, MORF seems to make odd decisions: in $\approx 82\%$ of the time, no tokenization is made, but in the other 18% very wild and excessive decisions take place.

5 Evaluation

5.1 Experimental Settings

We test a total of six systems (REGEX-S1, REGEX-S2, REGEX-S3, REGEX-S4, MORF, HTAG), as well as a no-tokenization baseline. For all of the systems, our data is a Hebrew-English sentence-aligned corpus produced by Tsvetkov and Wintner (2010). We split the data into training, tuning, development, and test sets. The training and tuning data sets are used for training and tuning the translation models. Experiments were initially run on the development data set, and finally run on the test data set when all settings and schemes were finalized. Table 3 presents the data subset details.

In the baseline, the Hebrew data is tokenized just to split punctuation. English data is white-space/punctuation tokenized and lowercased. The English MT output is true-cased using the recaser tool that is part of the Moses toolkit (Koehn et al., 2007). The recaser is trained on the English side of the training and tuning sets. For the baseline and all of the experiments, the preprocessing is applied to all data sets - training, tuning, development, and test. After preprocessing, but before training, we filter down to sentences of 100 tokens or less in length. As a result, with more tokenization, there are fewer eligible sentences. The difference is minor, however. We train the translation models and decode with the Moses toolkit (Koehn et al., 2007). We used two English language models, held constant across all experiments: a trigram language model from the English side of the training data

and a large 5-gram language model that preexisted this effort from English Gigaword (Graff and Cieri, 2003). Feature weights are tuned to maximize BLEU (Papineni et al., 2002) using Minimum Error Rate Training (Och, 2003) for each system separately.

5.2 Results and Discussion

The results are summarized in Table 4. Results are presented in terms of BLEU (Papineni et al., 2002), NIST (Doddington, 2002) and METEOR (Banerjee and Lavie, 2005).⁵

There is a general trend of improvement in BLEU score going down the table. Each subsequent experiment does better than the last in both the development and test data sets, with the exceptions of REGEX-S3 and REGEX-S4 as compared to REGEX-S2. This is a similar trend to Arabic (Habash and Sadat, 2006). Morphological analysis has a clear impact on translation quality, with both MORF and HTAG scoring higher than the regular expression systems. HTAG also is consistently the best performer in terms of all studied metrics. All differences in BLEU and NIST scores between all systems and the baseline and between MORF and HTAG are statistically significant above the 95% level. The differences between MORF and HTAG and each of REGEX-S1 through REGEX-S4 are also significant. Statistical significance is computed using paired bootstrap resampling with 1000 samples (Koehn, 2004).

The METEOR results support HTAG being the best system; however, the METEOR difference between HTAG and MORF is much bigger than in BLEU; and MORF is not consistently ranked second best.

It's notable that although MORF has no Hebrew-specific linguistic knowledge behind it, it is competitive with the REGEX techniques. This seems to show that linguistic information may not be sufficient to make a non-sophisticated technique perform well, and that unsupervised segmentation can go quite far.

Loosely, OOV levels drop as scores improve, but there are a few exceptions. REGEX-S4 has a lower OOV level than the other regular expression experiments, but its performance varies. A particularly notable exception is HTAG compared to MORF, where MORF has a significantly lower

⁵We used METEOR v1.2 with HTER task mode (Denkowski and Lavie, 2010).

Data Set	Sentences	Tokens	Types	Token OOV	Type OOV
Training	64,155	853,827	83,606		
Tuning	500	7,299	3,762	683 (9.4%)	677 (18.0%)
Development	1,000	11,405	4,386	798 (7.0%)	786 (17.9%)
Test	1,000	14,354	6,249	1311 (9.1%)	1288 (20.6%)

Table 3: Data set statistics.

	Development				Test			
	BLEU %	NIST	METEOR	OOV	BLEU %	NIST	METEOR	OOV
Base	20.96	5.3015	42.99	798	19.31	5.4951	44.36	1311
REGEX-S1	21.54	5.3805	44.61	587	20.39	5.6468	45.46	985
REGEX-S2	22.21	5.4491	43.26	401	21.69	5.8082	46.50	671
REGEX-S3	22.38	5.5365	44.33	318	21.61	5.8761	46.60	567
REGEX-S4	21.24	5.4021	42.22	273	21.07	5.8067	46.03	461
MORF	23.06	5.5590	43.16	28	22.25	5.9751	46.53	48
HTAG	23.09	5.6317	44.87	349	22.79	6.1033	48.20	556
COMBO1	22.69	5.5612	43.47	44	22.72	6.0381	47.20	74
COMBO2	22.68	5.5458	43.78	159	22.69	6.0275	47.17	250

Table 4: Results on development and test sets in multiple MT evaluation metrics. OOVs are presented in absolute (not percentage) counts.

Hebrew	החמאס ייהנה מהשפע הזה ויחזק את מעמדו.
Reference	Hamas will benefit from this bonanza.
Base	Hamas ייהנה this מהשפע and his status.
S1	Hamas ייהנה מהשפע and will the status.
S2	Hamas will benefit from abundance this will his status.
S3	Hamas will benefit from abundance and adds the status.
S4	Hamas will this affect this abundance standing and adds.
MORF	Hamas will be here what plenty and he adds the status.
HTAG	Hamas will benefit from abundance and will his status.
Hebrew	יש לנו קומקום ופלאטה בחדר.
Reference	We have an electric kettle and a hotplate in our room.
Base	We have brought ופלאטה in the room.
S1	We have קומקום and פלאטה in the room.
S2	We have קומקום and פלאטה in the room.
S3	We've got קומקום and פלאטה in the room.
S4	We have kettle and ופלאט room.
MORF	We've got a complete wonder anywhere.
HTAG	We've got kettle and פלאטה in the room.

Table 5: Translation examples.

OOV level, but also lower scores. By looking at the data, it is very clear that MORF’s aggressive segmentation is behind the low OOV level, while it seems that HTAG always does the correct level of segmentation. Because of this, MORF’s lower OOV level does not necessarily seem to contribute to better MT quality.

The example translations in Table 5 demonstrate some of these points. In the first example, OOV words are a major problem for the baseline system. By REGEX-S2, OOV is no longer a problem. Systems that segment more begin to produce more extraneous words. Finally, HTAG, instead of over-segmenting, produces the same output as REGEX-S2. In the second example, much more segmentation is required to deal with the OOV words. Once again, HTAG closely matches the REGEX-based system with the best output, and manages to successfully translate one of the OOV words. On the other hand, MORF shows its overaggressive segmentation, as it eliminates OOV words, but comes up with completely unrelated words instead.

Preliminary Combination Experiments In a preliminary combination experiment, we considered two simple ideas to combine the power of HTAG with other systems. First, for every sentence in the output of HTAG, if the sentence has an OOV, and MORF does not, we replace the HTAG output with the MORF output (COMBO1 in Table 4). Note that if a MORF sentence has even one OOV word, the corresponding HTAG sentence would not be replaced, even if it had several OOV words. Second, we retranslate the HTAG sentence after we replace each HTAG OOV with a tokenization from one of the other systems that makes the OOV invocable in HTAG (COMBO2 in Table 4). This is done with a preference for the most conservative REGEX-S1 down to the least conservative REGEX-S4 and then backing off to MORF. A replacement would not happen if either no method had a tokenization, or the tokenization didn’t produce tokens in the phrase table for HTAG. This second scenario was especially likely for MORF tokenizations. The results are not promising, scoring lower than HTAG. These experiments suggest that the OOVs that are unhandled are very hard to address without additional data or more intensive language-specific OOV handling approaches (Habash, 2008). More sophisticated approaches to MT combination can be explored in the future (Rosti et al., 2007).

6 Conclusions and Future Work

We explored a range of preprocessing solutions for Hebrew-English SMT. Our best system, using a morphological analyzer and tagger, increases 3.5 BLEU points over a no-tokenization baseline on a blind test set. The next best result we got (as measured by BLEU) uses Morfessor, an unsupervised morphological segmenter. In the future, we plan to explore combinations of the different tokenization schemes, both pre- and post-translation, perhaps using lattices (Dyer et al., 2008). We also plan to consider Hebrew-specific OOV solutions similar to work by Habash (2008) on Arabic.

Acknowledgments

The work presented here was supported in part by a Google research award. We would like to thank Or Biran, Alon Lavie, Yuval Marton, and Shuly Wintner for helpful feedback and discussions.

References

- Adler, Meni. 2009. *Hebrew Morphological Disambiguation: An Unsupervised Stochastic Word-based Approach*. Ph.D. thesis, Ben Gurion University.
- Al-Haj, Hassan and Alon Lavie. 2012. The impact of Arabic morphological segmentation on broad-coverage English-to-Arabic statistical machine translation. *Machine Translation*, 26:3–24.
- Banerjee, Satanjeev and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan.
- Creutz, Mathias and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Trans. Speech Lang. Process.*, 4:3:1–3:34.
- Denkowski, Michael and Alon Lavie. 2010. Extending the meteor machine translation evaluation metric to the phrase level. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 250–253, Los Angeles, California.
- Doddington, George. 2002. Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics. In *Human Language Technology*, pages 128–132, San Diego.
- Dyer, Christopher, Smaranda Muresan, and Philip Resnik. 2008. Generalizing Word Lattice Translation. In *Proceedings of ACL-08: HLT*, Columbus, Ohio.

- El Kholy, Ahmed and Nizar Habash. 2012a. Orthographic and morphological processing for English–Arabic statistical machine translation. *Machine Translation*, 26:25–45.
- El Kholy, Ahmed and Nizar Habash. 2012b. Translate, Predict or Generate: Modeling Rich Morphology in Statistical Machine Translation. In *Proceedings of EAMT 2012*, Trento, Italy.
- Fishel, Mark and Harri Kirik. 2010. Linguistically motivated unsupervised segmentation for machine translation. In *Proceedings of the seventh International Conference on Language Resources and Evaluation (LREC)*, Valletta, Malta.
- Goldwater, Sharon and David McClosky. 2005. Improving Statistical MT Through Morphological Analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 676–683, Vancouver, Canada.
- Graff, David and Christopher Cieri. 2003. English Gigaword, LDC Catalog No.: LDC2003T05. Linguistic Data Consortium, University of Pennsylvania.
- Habash, Nizar and Fatiha Sadat. 2006. Arabic pre-processing schemes for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers, NAACL-Short '06*, pages 49–52, Stroudsburg, PA.
- Habash, Nizar. 2008. Four Techniques for Online Handling of Out-of-Vocabulary Words in Arabic-English Statistical Machine Translation. In *Proceedings of ACL-08: HLT, Short Papers*, pages 57–60, Columbus, Ohio.
- Habash, Nizar. 2010. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.
- Itai, Alon and Shuly Wintner. 2008. Language resources for Hebrew. *Language Resources and Evaluation*, 42:75–98.
- Koehn, P. and K. Knight. 2003. Empirical methods for compound splitting. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1*, pages 187–193.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Christopher Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Christopher Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.
- Koehn, Philipp. 2004. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of the Empirical Methods in Natural Language Processing Conference (EMNLP'04)*, Barcelona, Spain.
- Lavie, A., S. Wintner, Y. Eytani, E. Peterson, and K. Probst. 2004. Rapid prototyping of a transfer-based Hebrew-to-English machine translation system. In *10th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI)*.
- Lee, Y.S. 2004. Morphological analysis for statistical machine translation. In *Proceedings of HLT-NAACL 2004: Short Papers*, pages 57–60.
- Lembersky, Gennadi, Noam Ordan, and Shuly Wintner. 2011. Language Models for Machine Translation: Original vs. Translated Texts. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 363–374, Edinburgh, Scotland, UK.
- Nießen, Sonja and Hermann Ney. 2004. Statistical Machine Translation with Scarce Resources using Morpho-syntactic Information. *Computational Linguistics*, 30(2).
- Och, Franz Josef. 2003. Minimum Error Rate Training for Statistical Machine Translation. In *Proceedings of the 41st Annual Conference of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan.
- Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, PA.
- Popović, Maja and Hermann Ney. 2004. Towards the Use of Word Stems and Suffixes for Statistical Machine Translation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, pages 1585–1588, Lisbon, Portugal.
- Rosti, Antti-Veikko, Necip Fazil Ayan, Bing Xiang, Spyros Matsoukas, Richard Schwartz, and Bonnie Dorr. 2007. Combining outputs from multiple machine translation systems. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 228–235, Rochester, New York.
- Shilon, Reshef, Nizar Habash, Alon Lavie, and Shuly Wintner. 2012. Machine translation between Hebrew and Arabic. *Machine Translation*, 26:177–195.
- Snyder, Benjamin and Regina Barzilay. 2008. Unsupervised Multilingual Learning for Morphological Segmentation. In *Proceedings of ACL-08: HLT*, pages 737–745, Columbus, Ohio.
- Tsvetkov, Y. and S. Wintner. 2010. Automatic acquisition of parallel corpora from websites with dynamic content. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, pages 3389–3392.
- Virpioja, Sami, Jaakko J. Vrynen, Mathias Creutz, and Markus Sadeniemi. 2007. Morphology-aware statistical machine translation based on morphs induced in an unsupervised manner. In *Proceedings of Machine Translation Summit*, Copenhagen, Denmark.

Poster Session 1 – User and Project Papers

Building Translation Awareness in Occasional Authors: A User Case from Japan

Midori Tatsumi

Toyohashi University of Technology
midori.tatsumi2@mail.dcu.ie

Anthony Hartley

Toyohashi University of Technology
a.hartley@imc.tut.ac.jp

Hitoshi Isahara

Toyohashi University of Technology
isahara@tut.jp

Kyo Kageura

University of Tokyo
kyo@p.u-tokyo.ac.jp

Toshio Okamoto

Toyota Boshoku Corporation
toshio_okamoto@toyota-
boshoku.co.jp

Katsumasa Shimizu

Toyota Boshoku Corporation
katsumasa_shimizu@toyota-
boshoku.co.jp

Abstract

We report the early stages of an industrial-academic collaboration to build translation awareness within a global Japanese company where non-professional authors are called upon to write ‘global job manuals’ for internal dissemination. Following an analysis of current practice, we devised a document template and simple writing rules which we tested experimentally with two MT systems. Overall, native-speaker judges found that the quality of the Japanese was maintained or improved, while the impact on the raw English translations varied according to MT system. The case study has wider implications for the acceptance of structured authoring by non-professional and occasional writers.

Toyota Boshoku Corporation also has a distinctive ethos, which places high value on Corporate Social Responsibility, on building ties with local communities and on employee welfare. In day-to-day management it adheres at every level to the twin principles of *kaizen* (continuous improvement) and *genchi-genbutsu*, code of action based on establishing the root cause of problems.

To promote on the job training and education, the company pairs new recruits with ‘workplace seniors’. As an extension of this concept, key staff in Japan are now being asked to capture their know-how in ‘global job manuals’ so that their expertise can be shared widely across the company network, within and outside Japan. This will entail translation, initially into English.

The current project was set up to explore the use of MT as a cost-effective means of meeting this need.

1 Starting Points

1.1 User profile and need

Toyota Boshoku Corporation is a Japanese company with a global presence in the design and manufacturing of automobile components. Operating in around 90 companies worldwide, the group is aware that the use of Japanese will become problematic as it further globalizes its operations. The designation of English as the official company language is under consideration.

1.2 Writers and readers

From an MT perspective, it is widely acknowledged that the typological ‘distance’ between Japanese and English hampers the achievement of high-quality translation. A well-known way of mitigating this problem is the Controlled Language (CL) approach. However, as Nyberg et al. (2003) stress, the main condition for the successful implementation of CL is to employ trained, professional authors. The challenge of the current project is to promote consistency and clarity of writing by people who do not see themselves primarily as authors and who are called upon only occasionally to write for a readership beyond their immediate working environment. Thus, they bring no prior training to the task and

cannot be expected to invest great effort in extending their language awareness.

At the same time, their readers are ‘insiders’ with experience of the corporate culture and can be expected to tolerate some infelicity of expression provided the content is understandable.

These constraints suggest that some form of ‘CL-lite’ may be appropriate. But is it feasible?

1.3 Document format and style

We based our diagnosis of the current situation on three work manuals comprising 33, 20, and 53 pages, or 177,742, 10,433, and 32,366 characters respectively. Using Systran 7 Premium we machine translated all three documents and had one post-edited by a professional translator.

The documents were rendered in Excel, which is a format widely used in Japan in both the technical and administrative domains. As a result, many sentences were broken across two or more cells, which had a predictably negative impact on MT quality. Repairing the breaks improved raw translation quality but destroyed the layout of the exported document. A considerable proportion of the text was embedded in figures and other graphic objects, and much of this was not extracted by the Systran filters. Together, these two factors increased translation costs by an estimated 20%.

The documents were very heterogeneous in wording and style. MT output quality was correspondingly patchy, even when we applied a user dictionary created from the glossary accompanying the manual, augmented with terms identified by the translator.

2 Proposed Remedies

2.1 Word template

With an ultimate goal of implementing some sort of XML format, such as DITA, we designed a MS Word template that organises information into concepts, tasks, reference, etc. It also requires an explicit characterization of readers and their purpose in consulting the document, in order to encourage a user-oriented mindset in the writer.

As an incentive for the next set of authors to abandon Excel, we implemented a full stylesheet, and provided a simple tool for extracting candidate terms from the document and inserting them into a formatted glossary.

2.2 Authoring guidelines

Authors of the existing manuals received no guidance on writing style as such. We reviewed the relatively scarce work on CL in Japanese, which dates back to (Nagao and Tanaka, 1984), who describe a ‘machine-readable’ Japanese. Yoshida (1987) outlines a framework for designing a ‘standardised’ Japanese for MT. Kaji (1999) offers a few Japanese examples. Sato et al. (2003) focus on interaction, while the efficacy of the rules proposed by Ogura et al. (2010) is not validated with empirical evidence. General technical and business writing guidebooks¹ provided suggestions for some of the guidelines we formulated. Others were chosen to remedy known problems of Japanese to English MT.

We ended up with the following 10 guidelines which we believed to be accessible and easy to implement.

- a. Do not use single-byte Katakana characters

Katakana is the only one of the three writing systems of Japanese that can also be written in single byte, which can perturb tokenisation by MT systems.

- b. Do not use mathematical symbols in sentences

Symbols are often used in sentences to represent relations concisely.

- c. Do not use nakaguro (bullet) as a delimiter

Nakaguro is often used to separate parallel list items in a sentence. MT systems can fail to distinguish parallel items (underlined) from the surrounding text.

会社のステージ・業績に応じた賃金、
賞与の水準

- d. Avoid using inappropriate Kanji characters

This equates to spelling mistakes in English.

- e. Avoid creating long noun strings

- f. Do not use ‘perform’ to create a *sa*-verb

Sa-verbs are widely used and are formed by adding a ‘do’ verb after a noun. Instead of using the simple する, writers commonly add ‘perform’ or ‘execute’ (行う／実行する).

- g. Avoid topicalisation

¹ 日本語スタイルガイド 第2版 (一般財団法人テクニカルコミュニケーション協会編著), 読得できる文章・表現 200 の鉄則 (日経 BP 社出版局)

Japanese is a ‘topic-prominent’ language. Some MT systems fail to translate non-subject topics correctly when they are signalled by the default topic particle は.

- h. Do not connect sentences to make a long sentence
- i. Do not interrupt a sentence with a bulleted list
- j. Avoid listing numerous parallel items in a sentence; use a bulleted list instead

3 Trial evaluation

From the existing manuals we selected six sentences violating each of the 10 rules and edited them according to the guidelines. We translated both versions with Excite² and Google Translate³, ‘off the shelf’. Native-speaker judges were recruited within the company to evaluate both the Japanese and the English MT outputs.

3.1 Questionnaire design and completion

We wanted to establish whether the quality of the Japanese source text written according to the guidelines is as good as or better than that of the text written without guidelines. We also wanted to know whether one or both are acceptable or not. The 20 judges were shown a pair of ‘before’ and ‘after’ sentences at a time and asked to evaluate each of them on the four-point scale in Figure 1. (For convenience we provide the questions in English gloss.)

The following two sentences convey the same content but are written using different words. Please evaluate the readability of each sentence.

A 欠勤・早退・遅刻・離業など、業務に従事していないときの賃金は、原則として支払いません。

B 欠勤・早退・遅刻・離業など、業務に従事していないときは、原則として賃金を支払いません。

How readable is A? Tick the closest option:

○ Easy ○ Fairly easy ○ Fairly difficult ○ Difficult

How readable is B? Tick the closest option:

○ Easy ○ Fairly easy ○ Fairly difficult ○ Difficult

Figure 1. Question to judges of Japanese

We surmised that showing two sentences at a time would lead the judges to focus on readabil-

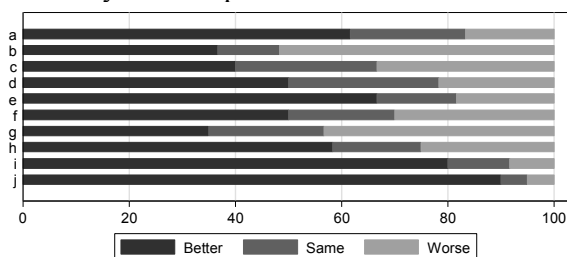
ity in terms of expression rather than content. Moreover, although the judges were not explicitly asked to compare the two and decide which was better, we thought that, if they perceived a difference in readability between the two texts, they might differentiate between them in their judgment.

In evaluating the English translations we asked the judges to say whether they thought sentence A more readable than B, B more readable than A, or A and B equally readable. This decision was dictated by the small number of judges available (eight).

The ordering of the ‘before’ and ‘after’ pairs was randomised for both languages. For the English translations, each judge saw (in random order) an equal number of outputs from each MT system, and no judge saw translations by both systems of the same Japanese source pair. We obtained four judgments for each source and target pair. The questionnaires were answered online.

3.2 Readability of the Japanese

The questionnaire design enabled us to draw conclusions on both the relative and absolute readability of the Japanese text.



In relative terms, Figure 2 shows that most of the guidelines achieved the objective of improving or at least maintaining the quality of the text, in so far as they were valued as Better or Same by at least two thirds of the judges.

The exceptions were *b* (Avoid symbols) and *g* (Avoid topicalisation). Guideline *c* (Avoid nakaguro) also received a rather low evaluation, which suggests that the use of non-linguistic devices to relate meaningful parts of a sentence promotes concision. The result for *g* was somewhat expected, since topicalisation does not usually compromise readability for humans and editing sentences to eliminate topicalisation can result in wordiness.

The greatest positive impact on readability was registered by guidelines *i* (Do not interrupt

² <http://www.excite.co.jp/world/>

³ <http://translate.google.com/>

the sentence before bulleted lists) and *j* (Avoid listing parallel items in a sentence).

To ground the absolute readability of the text, we converted the rating options to numbers as follows: ‘Easy to read’ = 4, ‘Fairly easy’ = 3, ‘Fairly difficult’ = 2, ‘Difficult’ = 1.

Table 1 compares the median values of the evaluation results for JAO (‘original’) and JAR (‘rewritten’). We see that overall readability for both JAO and JAR is rather good; there is no category whose median value is lower than 2. This is not surprising, however, since all sentences have been written by a human.

More important, there are no categories for which JAR received a lower score. This suggests that the guidelines we used for this experiment was generally successful in maintaining and even raising the quality of Japanese sentences.

	JAO	JAR	EXC	GOO
a	3	4	0	1
b	3	3	1	-4
c	3	3	0	-1
d	2.5	3	1	1
e	3	4	-3	3
f	3	3	3	-1
g	3	3	-1	3
h	3	3	-5	2
i	2	4	5	2
j	2	4	2	-2

Table 1. Readability and translation quality

3.3 Translation quality of the English

The last two columns of Table 1 give the net sum of the judgments comparing ENO/ENR-Excite and ENO/ENR-Google, respectively, in the range +12 to -12. It appears that only with rules *d* and *i* do all three indicators improve. Hartley et al. (2012) discuss the conflicting impacts of the rules on Excite (RBMT) and Google (SMT).

Note that these are relative changes in the performance of the same system given modified inputs. Limitations on the availability of competent judges prevented us from grounding the judgments in terms of the acceptability of the sentences, as we did with the Japanese input.

4 Conclusions

The fact that we are dealing with non-professional and possibly reluctant writers is a big factor. We have emphasized readability of the Japanese since, if it is perceived to suffer, authors will be likely to simply reject the guidelines. But the fact that simple rules did consis-

tently maintain or improve readability may motivate the writers to use them, even if only two rules also consistently raise MT quality.

Some 90 authors are creating global job manuals by a June 2012 deadline. Although use of the template is not mandatory, a majority are expected to use it.

The next step is to translate the manuals and establish, with Toyota Boshoku staff, the necessary quality benchmark for post-editing. This may be attainable using either Japanese translators without English native review or by Toyota Boshoku staff outside Japan who are not professional translators. We will also investigate how closely the authors adhered to the guidelines.

Acknowledgments

This work was funded by the Strategic Information and Communication R&D Promotion Programme of the Ministry of Internal Affairs and Communications, Japan.

References

- Hartley, Anthony, Midori Tatsumi, Hitoshi Isahara, Kyo Kageura and Rei Miyata. 2012. Readability and Translatability Judgments for ‘Controlled Japanese’. *European Association for Machine Translation*. Trento, IT.
- Kaji, Hiroyuki. 1999. Controlled Languages for Machine Translation: State of the art. *Machine Translation Summit VII*. Singapore. 37-39.
- Nagao, Makoto and Nobuyoshi Tanaka. 1984. Support System for Writing Texts Based on Controlled Grammar. *Information Processing Society of Japan*, NL-44:33-40.
- Nyberg, Eric, Teruko Mitamura and Willem-Olaf Huijsen. 2003. Controlled Language for Authoring and Translation. H. Somers (ed.). *Computers and Translation*. Benjamins, Amsterdam, NL. 245-281.
- Ogura, Hidesato, Mayo Kudo and Hideo Yanagi. 2010. Simplified Technical Japanese: Writing Translation-Ready Japanese Documents. *Information Processing Society of Japan*, DD-5:1-8.
- Sato, Satoshi, Masatoshi Tsuchiya, Masahiro Asaoka, Masahiro Asaoka and Qingqing Wang. 2003. Standardizing Japanese Sentences. *Information Processing Society of Japan*, NL-4:133-140.
- Yoshida, Sho. 1987. Standardizing Japanese and Design of Controlled Japanese. Organizing Committee of 1st University Science Public Symposium (ed.). *Characteristics of Japanese and Machine Translation*. Tokyo, Japan. 132-142.

Efficiency-based evaluation of aligners for industrial applications*

Antonio Toral
School of Computing
Dublin City University
Dublin, Ireland

atoral@computing.dcu.ie

Marc Poch
IULA, Universitat
Pompeu Fabra
Barcelona, Spain

marc.pochriera@upf.edu

Pavel Pecina
Faculty of Mathematics and
Physics, Charles University
Prague, Czech Republic

pecina@ufal.mff.cuni.cz

Gregor Thurmair
Linguattec GmbH
Munich, Germany

g.thurmair@linguatec.de

Abstract

This paper presents a novel efficiency-based evaluation of sentence and word aligners. This assessment is critical in order to make a reliable use in industrial scenarios. The evaluation shows that the resources required by aligners differ rather broadly. Subsequently, we establish limitation mechanisms on a set of aligners deployed as web services. These results, paired with the quality expected from the aligners, allow providers to choose the most appropriate aligner according to the task at hand.

1 Introduction

Aligners refer in this paper to tools that, given a bilingual corpus, identify corresponding pairs of linguistic items, be they sentences (sentence aligners) or words (word aligners). Alignment is a key component in corpus-based multilingual applications. First, alignment is one of the most time-consuming tasks in building Machine Translation (MT) systems. In terms of quality, good alignment is decisive for the final quality of the MT system; bad alignment decreases MT quality and inflates the phrase table with spurious translations with very low probabilities, which reduces system performance. Finally, for terminology acquisition, the choice of a good aligner determines whether the results of a term extraction tool are usable or not; alignment quality on phrase level differs from

less than 5% (usable) to more than 40% (unusable) error rate (Aleksic and Thurmair, 2012).

The performance of aligners is commonly evaluated extrinsically, i.e. by measuring their impact in the result obtained by a MT system that uses the aligned corpus (Abdul-Rauf et al., 2010; Lardilleux and Lepage, 2009; Haghighi et al., 2009). Intrinsic evaluations have also been carried out, mainly by measuring the Alignment Error Rate (AER), precision and recall (von Waldenfels, 2006; Varga et al., 2005; Moore, 2002; Haghighi et al., 2009). Intrinsic evaluation is less popular due to two reasons (Fraser and Marcu, 2007): (i) it requires a gold standard and (ii) the correlation between AER and MT quality is very low. Both types of evaluation have, however, a common aspect; they focus on measuring the quality of the output produced by aligners. Conversely, seldom if at all has it been considered to assess the efficiency of aligners, i.e. to measure the computational resources consumed (e.g. execution time, use of memory). However, this assessment is critical if the aligners are to be exploited in an industrial scenario.

This work is part of a wider project, whose objective is to automate the stages involved in the acquisition, production, updating and maintenance of language resources required by MT systems. This is done by creating a platform, designed as a dedicated workflow manager, for the composition of a number of processes for the production of language resources, based on combinations of different web services.

The present work builds upon (Toral et al., 2011), where we presented a web service architecture for sentence and word alignment. Here we extend this proposal by evaluating the efficiency of the aligners integrated, and subsequently im-

We would like to thank Daniel Varga and Adrien Lardilleux for their feedback on Hunalign and Anymalign, respectively. We would like to thank Joachim Wagner for his help on using the cluster. This research has been partially funded by the EU project PANACEA (7FP-ITC-248064).

© 2012 European Association for Machine Translation.

proving the architecture by implementing limitation mechanisms that take into account the results.

2 Evaluation

We have integrated a range of state-of-the-art sentence and word aligners into the web service architecture. The sentence aligners included are Hunalign (Varga et al., 2005), GMA¹ and BSA (Moore, 2002). As for word aligners, they are GIZA++ (Och and Ney, 2003), BerkeleyAligner (Haghighi et al., 2009) and Anymalign (Lardilleux and Lepage, 2009). For a detailed description of the integration please refer to (Toral et al., 2011).

In order to evaluate the efficiency of the aligners, we have run them over different amounts of sentences of a bilingual corpus (from 5k to 100k adding 5k at a time for sentence alignment and from 100k to 1.7M adding 100k at a time for word alignment). For all the experiments we use sentences from the Europarl English–Spanish corpus,² which contains over 1.7M sentence pairs. The aligners are executed using the default values for their parameters. All the experiments have been run in a cluster node with 2 Intel Xeon X5670 6-core CPUs and 96 GB of RAM. The OS is GNU/Linux. The resources consumed have been measured using the following parameters of the GNU command `time`:

- %S (CPU-seconds used by the system on behalf of the process) plus %T (CPU-seconds that the process used directly), to measure the execution time. We limit our experiments to 100k seconds.
- %M (maximum resident set size of the process during its lifetime, in Kilobytes), to measure the memory used.

Figure 1 shows the execution times (logarithmic scale) of the sentence aligners. It emerges that the time required by GMA is considerable higher compared to the other two aligners (e.g., for 45k sentences GMA takes approximately 16 and 20 times longer than BSA and Hunalign, respectively). The gap grows exponentially with the input size.

Figure 2 shows the memory consumed by the sentence aligners. Hunalign has a steeper curve (for 45k sentences, Hunalign uses 6 and 4 times more memory than BSA and GMA, respectively).

¹<http://nlp.cs.nyu.edu/GMA/>

²<http://www.statmt.org/europarl/>

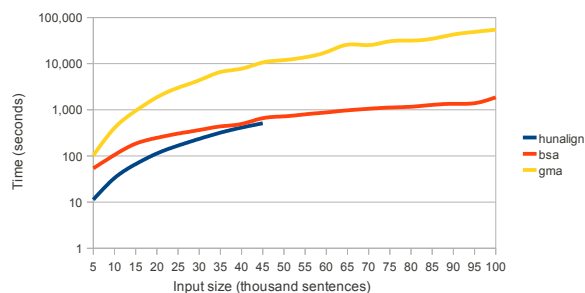


Figure 1: Execution time for sentence aligners

In fact Hunalign was not able to align inputs of more than 45k sentences due to memory issues.³ Table 1 contains all the measurements for sentence alignment.

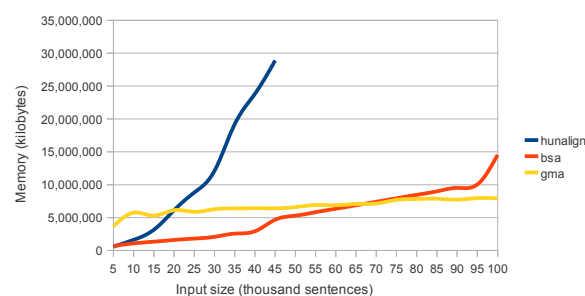


Figure 2: Memory used by sentence aligners

i	Time (seconds)			Memory (M bytes)		
	hun	bsa	gma	hun	bsa	gma
5	11	54	103	584	684	3,677
10	33	105	405	1,616	1,079	5,749
15	66	185	950	3,146	1,337	5,305
20	113	247	1,866	6,115	1,597	6,126
25	168	305	3,004	8,803	1,807	5,878
30	234	364	4,370	12,104	2,070	6,276
35	319	436	6,578	19,211	2,559	6,390
40	412	494	7,775	23,827	2,919	6,433
45	510	659	10,609	28,892	4,679	6,415
50	-	721	11,947	-	5,297	6,594
55	-	797	13,768	-	5,824	6,915
60	-	878	17,780	-	6,347	6,888
65	-	973	25,787	-	6,872	7,061
70	-	1,053	25,251	-	7,415	7,143
75	-	1,120	30,513	-	7,940	7,692
80	-	1,165	31,591	-	8,469	7,832
85	-	1,277	34,664	-	8,991	7,872
90	-	1,348	42,720	-	9,518	7,730
95	-	1,391	48,823	-	10,043	7,969
100	-	1,863	54,350	-	14,537	7,911

Table 1: Detailed results for sentence aligners. **i** input sentences (thousand), **hun** hunalign

Figure 3 shows the execution times for word aligners. GIZA++ is the most efficient word aligner, consistently across the different inputs.

³A constant in the source code of Hunalign establishes the maximum amount of memory it will use, by default 4GB; we increased it to 64GB. Moreover, it can split the input into smaller chunks with `partialAlign` (it cuts the data into chunks of approximately 5,000 sentences each, based on hapax clues found on each side), however we did not use this preprocessing tool but only the aligner itself.

The performance of Berkeley is similar to that of GIZA++ for the first runs but the difference of execution time grows with the size of the input. There are no results for Berkeley for over 1,1M sentences as the time limit is exceeded. Finally, the behaviour of Anymalign does not correlate at all with the size of the input. This has to do with the very nature of this aligner.⁴

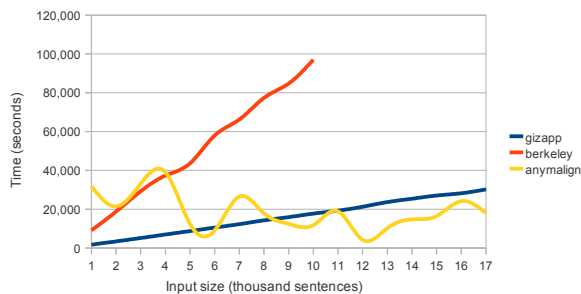


Figure 3: Execution time for word aligners

Figure 4 shows the memory required by word aligners. Berkeley consistently requires more memory than both GIZA++ and Anymalign. The requirements of GIZA++ and Anymalign are similar, although slightly lower for the latter. Table 2 contains all the measurements for word alignment.

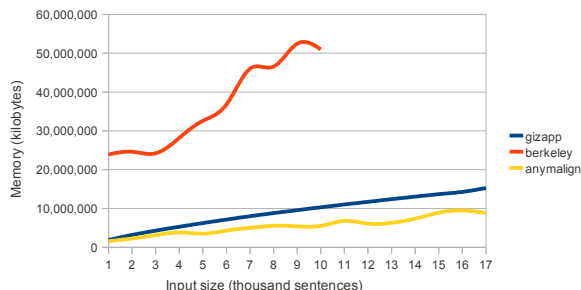


Figure 4: Memory used by word aligners

3 Limiting web services

The previous section has shown that the computational resources required by state-of-the-art aligners are very different. These resources are limited and must be taken into account when they are being shared by users using web services.

We have studied ways on establishing limitations for the aligners deployed as web services. Two kinds of limitations are explored and implemented: (i) the number of concurrent executions and (ii) the input size allowed for each aligner.

The web services are developed using Soaplab2.⁵ This tool allows to deploy web

⁴Anymalign runs are random, its stop criterion can be based on the number of alignments it finds per second, we set this parameter to the most conservative value supported, i.e. 1 alignment per second.

⁵<http://soaplab.sourceforge.net/soaplab2/>

i	Time (k seconds)			Memory (M bytes)		
	giz	brk	any	giz	brk	any
1	1,7	9,0	31,9	1,894	23,906	1,582
2	3,4	18,8	21,4	3,181	24,619	2,277
3	5,1	29,2	33,2	4,293	24,222	3,142
4	6,9	37,3	39,0	5,292	28,190	3,818
5	8,7	43,6	12,4	6,245	32,586	3,525
6	10,5	58,0	9,0	7,144	36,773	4,304
7	12,3	66,2	26,5	8,008	45,999	5,017
8	14,2	77,3	17,8	8,807	46,545	5,531
9	15,9	84,7	12,4	9,565	52,437	5,407
10	17,7	97,0	11,8	10,313	50,977	5,522
11	19,3	-	18,9	11,030	-	6,800
12	21,2	-	4,1	11,713	-	6,107
13	23,6	-	10,1	12,403	-	6,301
14	25,4	-	14,8	13,057	-	7,382
15	27,0	-	16,5	13,688	-	8,931
16	28,2	-	24,2	14,272	-	9,469
17	30,2	-	17,9	15,270	-	8,860

Table 2: Detailed results for word aligners. **i** input sentences (hundred thousand), **giz** GIZA++, **brk** Berkeley, **any** Anymalign

services on top of command-line applications by writing files that describe the parameters of these services in ACD format.⁶ Soaplab2 then converts the ACD files to XML metadata files which contain all the necessary information to provide the services. The Soaplab server is a web application run by a server container (Apache Tomcat⁷ in our setup) which is in charge of providing the services using the generated metadata.

Figure 5 shows the diagram of the program flow for web services that incorporates limitation mechanisms.⁸ The modules are the following:

- *tool.acd* (e.g. *bsa.acd*), contains the metadata of the web service in ACD format.
- *ws.sh*, controls other modules that implement the waiting and execution mechanisms.
- *init_ws.sh*, contains the code that implements the limitation on the number of concurrent executions and waiting queue. The web service is in waiting state while it is executing this script.
- *tool.sh* (e.g. *bsa.sh*), executes the tool. The web service is in executing state while it is executing this script.
- *ws_vars.sh*, contains all the variables used by the different web services.
- *ws_common.sh*, contains code routines shared by different web services.

⁶<http://soaplab.sourceforge.net/soaplab2/MetadataGuide.html>

⁷<http://tomcat.apache.org/>

⁸The code is available under the GPL-v3 license at BLIND

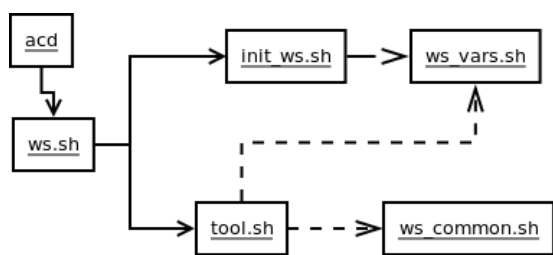


Figure 5: Diagram of the program flow

3.1 Limitation of concurrent executions

The limitation of concurrent executions is controlled by two variables, `MAX_WS_WAIT` and `MAX_WS_EXE`, set in `ws_vars.sh`. They hold the maximum number of web services that can be concurrently waiting and executing, respectively.

The following actions are carried out when a web service is executed. First, `tool.acd` calls `ws.sh`. This one calls sequentially two scripts: `init_ws.sh` and `tool.sh`. `init_ws.sh` checks if the waiting queue is full and aborts the execution if so. Otherwise it puts the execution in waiting state and checks periodically whether the execution queue is full. When there is a free execution slot, `init_ws.sh` exits returning the control to `ws.sh`, which changes the state to executing and calls `tool.sh`.

3.2 Limitation of input size

The limitation of input/output data size can be performed at three levels: Tomcat, Soaplab and web service. Tomcat provides a parameter, `MaxPostSize`, which indicates the maximum size of the `POST` in bytes that will be processed. Soaplab allows us to put a size limit (in bytes) to the output of web services using a property. The user can establish a general limit that applies to every web service, and/or specific limits that apply to any web service in particular.

Both these methods allow us to limit the input/output of web services in bytes. However, limiting the size according to different metrics might be useful. For example, the inputs of aligners are usually measured in number of sentences (rather than number of bytes). Limits of number of input sentences have been established at the web service level for each aligner following the results obtained in the evaluation (Section 2). Variables with the desired maximum input size in number of sentences have been added for each aligner in `ws_vars.sh`. A function included in `ws_common.sh` checks the size of the input whenever an aligner is executed.

4 Conclusions

This paper has presented, to the best of our knowledge, the first efficiency-based evaluation of sentence and word aligners. This assessment is critical in order to make a reliable use in industrial scenarios, especially when they are offered as services. The evaluation has showed that the resources required by aligners differ rather broadly. These results, paired with the quality expected from the aligners, allow providers to choose the most appropriate aligner according to the task at hand.

References

- Abdul-Rauf, S., M. Fishel, P. Lambert, S. Noubours, and R. Sennrich. 2010. Evaluation of Sentence Alignment Systems (Project at the Fifth Machine Translation Marathon).
- Aleksic, V. and G. Thurmair. 2012. Rule-based MT system adjusted for narrow domain (ACCURAT Deliverable D4.4.). Technical report.
- Fraser, A. and D. Marcu. 2007. Measuring Word Alignment Quality for Statistical Machine Translation. *Computational Linguistics*, 33:293–303.
- Haghighi, A., J. Blitzer, J. DeNero, and D. Klein. 2009. Better word alignments with supervised ITG models. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 923–931.
- Lardilleux, A. and Y. Lepage. 2009. Sampling-based multilingual alignment. In *Proceedings of RANLP*, pages 214–218, Borovets, Bulgaria.
- Moore, R. C. 2002. Fast and accurate sentence alignment of bilingual corpora. In *Proceedings of AMTA*, pages 135–144.
- Och, F. J. and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29:19–51.
- Toral, A., P. Pecina, A. Way, and M. Poch. 2011. Towards a User-Friendly Webservice Architecture for Statistical Machine Translation in the PANACEA project. In *Proceedings of EAMT*, pages 63–72, Leuven, Belgium.
- Varga, D., L. Németh, P. Halácsy, A. Kornai, V. Trón, and V. Nagy. 2005. Parallel corpora for medium density languages. In *Proceedings of RANLP*, pages 590–596, Borovets, Bulgaria.
- von Waldenfels, R. 2006. Compiling a parallel corpus of slavic languages. Text strategies, tools and the question of lemmatization in alignment. In *Beiträge der Europäischen Slavistischen Linguistik*, pages 123–138.

Evaluation of Machine-Translated User Generated Content: A pilot study based on User Ratings

Linda Mitchell

SALIS
Dublin City University
Ballymun, Dublin 9, Ireland

linda.mitchell17@mail.dcu.ie

Johann Roturier

Symantec SES EMEA Research
Ballycoolin Business Park
Blanchardstown, Dublin 15, Ireland

johann_roturier@symantec.com

Abstract

This paper presents the results of an experimental pilot user study, focusing on the evaluation of machine-translated user-generated content by users of an online community forum and how those users interact with the MT content that is presented to them. Preliminary results show that ratings are very difficult to obtain, that a low percentage of posts (21%) was rated, that users need to be well informed about their task and that there is a weak correlation between the length of the post (number of words) and its comprehensibility.

1 Introduction

This study follows up on the work described in Roturier and Bensadoun (2011), in which four machine translation systems were compared in order to evaluate their suitability in translating user-generated content. In the present study, the objective is different since feedback on machine-translated content is solicited from actual users of an existing community forum (rather than using linguists or bilingual technical support agents). Thus, an additional objective is to analyse how users interact with the MT content presented to them. This paper is divided into four parts: in Section 2, related work is briefly discussed. In Section 3, the experimental design of this study is presented, while in Section 4 preliminary results are reported. In Section 5, we make some conclusions and outline possible future work.

2 Related Work

The machine-translation of user-generated content has been identified as being potentially useful to allow communication between various user groups that do not share a common language (Flournoy and Rueppel, 2010). Indeed, it was announced in 2010 that TripAdvisor would be using Language Weaver-powered translations to make hotel reviews available in multiple languages¹. More recently, Facebook announced they would be using Microsoft's Bing Translate for Page content². Recent research work has also been performed in this area, including Roturier and Bensadoun (2011) and Banerjee et al. (2011). However, no study has focused on how machine-translated content would be received in-context by existing users of a forum community.

3 Experimental Design

The current German Norton Community forum³ is composed of multiple sections, known as "boards". We decided to create a specific board, where machine-translated content would be published⁴. In the introduction to this board, it was explained to users that this board is used to show machine translated posts (from English into German). A user (Max_MÜ) was created; a fictitious "MT robot" whose name is used to post machine-translated content. Additionally, a de-

¹ http://blogs.forrester.com/tim_walters/10-07-15-sdl_casts_vote_machine_translation_language_weaver_acquisition

² <https://www.facebook.com/photo.php?fbid=10150491112449572&set=a.121044129571.125587.10381469571&type=1>

³ <http://de.community.norton.com>

⁴ <http://de.community.norton.com/t5/L%C3%B6sung-nicht-gefunden/bd-p/Max>

scription was added to Max_MÜ's profile⁵, introducing himself and explaining the study. Max_MÜ's signature says explicitly that each of its post has been machine translated. One communication thread was opened and floated to the top to explain the study and present the users' feedback option including the voting mechanism. Feedback options consist of the newly developed voting mechanism, and the already existing options of commenting on the machine-translated posts and giving kudos, "a way for you to give approval to content that you think is helpful, well-formed, insightful, or otherwise generally valuable in the community"⁶.

3.1 Voting Mechanism

To collect genuine user feedback, a voting mechanism was developed. This mechanism consists of the question whether the machine translated post was comprehensible and the option of selecting either "yes" or "no", which is then send to a database via the "vote" button. This was written in Javascript and included on every page of the MT board. It was inserted to the left of each post in the MT board, as shown below:

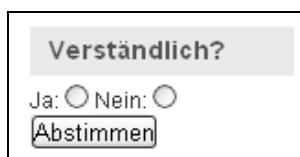


Figure 1. Feedback mechanism

3.2 Evaluation Criteria

In this experiment, comprehensibility, which refers to "the extent to which the text as a whole is easy to understand" (Hovy et al. 2002), is measured for machine translated user-generated content. It is evaluated in this study using a binary evaluation system: The user answers the question of whether a post was comprehensible or not, with either "yes" or "no" (see Figure 1).

3.3 Evaluation Data

The evaluation data was obtained from the English Norton forum⁷. In a first step, ninety threads were identified from different boards (Norton Internet Security, Norton 360, Online Family and

Norton Mac). The threads had to fulfill the condition that in addition to a question, they had to have one post marked as a solution. The process of retrieving two messages per thread (question and answer) from the English forum was automated using API requests and a script in Python. For the translation of the posts the API of the Microsoft Translator system was used⁸ since it is the system that had obtained the highest comprehensibility and fidelity scores in Roturier and Bensadoun (2011).

3.4 Experiment Procedure

For three weeks, the MT board was solely opened to the gurus (eight users). During this period, six valid votes were received. This test period showed that the voting mechanism worked and that users would have to be motivated by posts constantly to vote. The board was opened to the public (users and non-users) on 11 January 2012. Every week, ten new threads were posted to the MT board.

4 Results

4.1 Sections

During the evaluation time frame, votes were recorded for non-machine-translated content; after repeatedly specifying that users should only vote for content posted by Max_MÜ. This suggests that users do not necessarily read the introduction to the board or any other related post. Figure 2 shows the number of ratings collected per week. While there was an increase in votes initially, the number of ratings decreased noticeably after week 12. This might be related to user motivation and is a topic that will need to be addressed in the future. The number of different users who voted per week never exceeded five. While the users mostly voted for one or two posts at a time, there were instances of users voting for more posts (e.g. 18 posts Wk6).

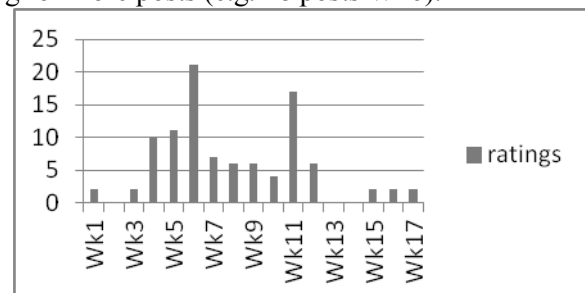


Figure 2. Number of valid ratings per week

⁵<http://de.community.norton.com/t5/user/viewprofilepage/user-id/6115>

⁶<http://community.norton.com/t5/Announcements/New-Feature-KUDOS/m-p/9713>

⁷<http://community.norton.com/>

⁸<http://www.microsofttranslator.com/dev/>

Between 20 December 2011 and 04 April 2012 94 valid ratings and 18 invalid ratings, e.g. ratings for non-machine-translated content were collected. Out of the valid ratings, 57 (61%) ratings were “yes”, i.e. the machine translated content was rated as comprehensible, and 37 (39%) were “no”, the machine-translated content was deemed incomprehensible. There were two more ratings for answers (48) than for questions (46). It is apparent from the results that, both for question and answers, “yes” was the preferred rating, as shown in Figure 3:

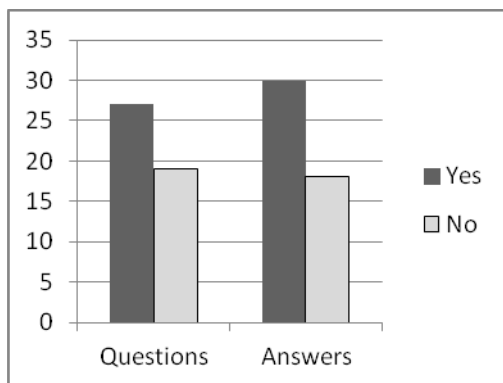


Figure 3. Ratings grouped into questions and answers

4.2 Interpretation of collected ratings

While these are only preliminary results, Figure 3 suggests that some machine-translated posts can be understood by users who do not have access to the source text. This confirms the results from Roturier and Bensadoun (2011), where on average, machine-translated posts were rated 2.6 on a scale of 5 (in terms of comprehensibility). We are interested in finding out whether some textual characteristics, such the length of a post, may have an impact of the comprehensibility ratings. For instance, the average number of words per post in those that were rated as comprehensible was 56, whereas it was 93 for those that were rated as incomprehensible. This suggests that the longer the post, the less likely it is to be comprehensible. This is only supported by a weak correlation (-0.35) between the two variables - when comprehensibility is expressed by 1 or “yes”. Thus, the relationship between length of post and comprehensibility seems to be more complex. More research needs to be conducted, e.g. on whether more context increases a post’s comprehensibility.

The users made sparse use of the other feedback options available to them (kudos, comments). Five posts received two ratings. Two

times, the users voted for the same answer, three times they voted differently. There were 210 threads (420 posts) available in the MT board. Only 88 (21%) of those posts received a rating. No kudos was given to any posts in the MT board. All of the comments (four) received in the MT board indicated that the users had not grasped the concept of the MT board, e.g. they mistook posts by other users as machine-translated content or they did not realise that the content was machine-translated. None of the comments were related to the quality of the actual MT output.

4.3 Visibility of the machine-translated posts and its impacts on rating behaviour

In the previous section, we have shown that machine-translated content could sometimes be understood by users, hence suggesting that it can be of value to these users. We are also interested in determining whether the content that is rated as comprehensible relates to important user issues. To achieve this, we analysed the top search terms on the German Norton Forum for the MT board, but found that no search queries were submitted during that time period. For the German Norton Forum in general, we found that error codes, such as “fehler 3040,20063” or “8920.201” were prevalent. While there was one MT post that had an error code in its subject “Fehler: 8.920.223”, there were no searches performed for that particular error code; however, both question and answer received a rating for this thread. This may suggest that posts including an error code (in the subject) are possibly the posts that are most accessible to the users. As the number of available searches is small for the German Norton forum (e.g. 116 single term searches within two months), we analysed the searches in the English Norton community (e.g. 52923 single term searches within two months) in order to determine possible candidates for keywords and to consequently re-rank the posts or change the way of selecting new posts. It was found, for example, that the Norton products are often searched for, as well as different browsers in connection with the Norton toolbar. This and information gained from reports on searches performed in independent search engines will be included in the selection process of threads to be machine translated in future.

Figure 5 shows the number of ratings posts collected depending on their position in the board at the time the rating was submitted. This figure

suggests that most of the ratings are likely to have been generated by users who went to the MT board deliberately, voted for some of the posts on the first page and subsequently left the board. Only six of the ratings received went below thread 9 on a page. The median number of posts voted for by a user in one session is 2, i.e. the number of posts voted for by the same user within a very short time frame. (The average number of posts voted for in one session is 3.5.)

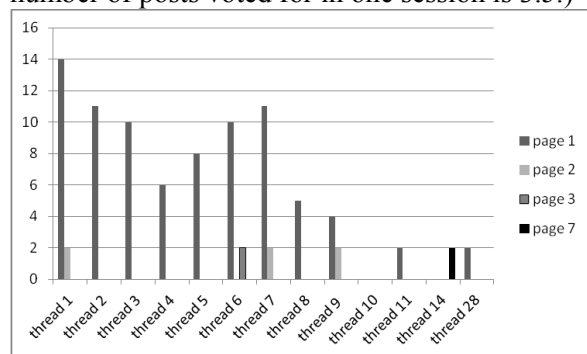


Figure 5. Position of threads voted for in MT board

5 Conclusion

This paper presented the setup and results of a pilot study focusing on the evaluation of machine translated user-generated content in an online community environment. These results point towards the content as being rated comprehensible slightly more often than not. The decision of rating a post as comprehensible may be influenced by the length of the post. The drawbacks of this study were that a limited number of ratings were collected. This is connected to the issue of motivation. It can be concluded from this study that the users need to be constantly reminded and, more importantly, motivated to vote. A possible reason for the low motivation to vote may be that a platform for German speakers is already in existence. Thus, it would be beneficial to the project, to see whether motivation to vote would increase for a language that does not have a community yet, e.g. in a Spanish board. By broadening the setup, we are hoping to receive a larger number of votes and a more general idea of whether MT content is acceptable for the users of an online community.

In addition to this, the machine-translated content could be made more relevant to the user by selecting the threads based on the findings of the analysis of search queries performed within and outside the Norton community. Some of these

issues will be tackled within the framework of an FP7-funded project, ACCEPT⁹.

References

- Banerjee, P., Naskar, S. K., Roturier, J., Way, A. and van Genabith, J. 2011. Domain Adaptation in Statistical Machine Translation of User-Forum Data using Component Level Mixture Modelling. In Proceedings of the Thirteenth Machine Translation Summit, pages 285–292, Xiamen, China.
- Flournoy, R., and Rueppel, J. 2010. One Technology: Many Solutions. Proceedings of AMTA 2010: the Ninth Conference of the Association for Machine Translation in the Americas. Denver, Colorado.
- Hovy, E., King, M. and Popescu-Belis, A. 2002. Principles of Context-Based Machine Translation Evaluation. *Machine Translation* 17 (1), 43-75.
- Roturier, J. and Bensadoun, A. 2011. Evaluation of MT Systems to Translate User Generated Content. In Proceedings of the Thirteenth Machine Translation Summit, pages 244–251, Xiamen, China.

⁹ <http://www.accept.unige.ch/Description.html>

A Machine Translation Toolchain for Polysynthetic Languages

Petr Homola

Codesign, s.r.o.

Palackého 541

252 29 Dobřichovice

phomola@codesign.cz

Abstract

We present a set of free tools for building rule-based machine translation systems for polysynthetic languages. As there are no large corpora for most of the “small” languages, it is often impossible to use statistical methods. There are some free MT tools but very little work has been done on polysynthetic languages. The aim of this project is to provide computational tools for morphological and syntactic processing for such languages.

1 Introduction

The paper describes a set of tools for natural processing of polysynthetic languages. There are quite a few definitions of polysynthesis. Baker (1996), for example, defines a ‘polysynthesis parameter’ within the Chomskyan framework. However his definition is quite strict and excludes many languages that are traditionally considered polysynthetic (such as Greenlandic). We use Mattissen’s (2006) definition which is closer to the understanding of polysynthesis of most researchers in the field. According to her, a language is polysynthetic if it contains “complex, polymorphemic verb forms which allow, within one word unit, for components in the form of non-root bound morphemes with quite ‘lexical’ meaning or optionally for the concatenation of lexical roots”.

Due to typological differences from Western languages, polysynthetic languages are quite a challenge for many theories of formal grammar. Our implementation is based on Lexical Functional Grammar (Kaplan and Bresnan, 1982; Bresnan, 2001). The system consists of a morpho-

logical analyzer, rule-based parser, transfer module and morphological generator. As an example of a polysynthetic word, consider the Aymara sentence *qullqinipachänwa* which corresponds to a complete English sentence:

- (1) *qullqi-ni-pacha-:n-wa*
 money-POSS-EVID-PAST_{3→3}-FOC
 “Apparently s/he had a lot of money.”

The paper is organized as follows: Section 2 describes how we analyze polysynthetic languages morphologically and syntactically. Section 3 gives an overview of the transfer phase. Finally we offer some conclusions in Section 4.

2 Morphological and Syntactic Analysis

2.1 Lexicon

Some polysynthetic languages, such as Aymara, have no closed morphological tagset since a stem can be nominalized and/or verbalized several times by adding various derivational suffixes recursively without any theoretical limit. The output of the morphological analyzer is a set of f(eature)-structures which contain morphosyntactic and lexico-semantic information. For example, the f-structure for the Aymara word form *uñjsma* “I see/saw you” is defined by the following morpholexical annotation:

- (2) $(\uparrow \text{PRED}) = \text{‘see}((\uparrow \text{SUBJ})(\uparrow \text{OBJ}))\text{’}$
 $(\uparrow \text{TENSE}) = \text{pres|simple_past}$
 $(\uparrow \text{SUBJ PERSON}) = 1$
 $((\uparrow \text{SUBJ PRED}) = \text{‘pro’})$
 $(\uparrow \text{OBJ PERSON}) = 2$
 $((\uparrow \text{OBJ PRED}) = \text{‘pro’})$

The functional equations in (2) encode the verb’s lemma and valency (in the PRED attribute),

polypersonal agreement (the person of SUBJ and OBJ) and the fact that both arguments can be dropped (in which case the value of the argument's PRED attribute is 'pro').

The lexicon contains an entry for the stem and a separate entry for the suffix:¹

```
(r v1 uñj uñja (v2)
  ((SUBJ ((ANIM 1)))) - V)
(s v2 sma (tf) ((TENSE nfut)
  (SUBJ ((PERS 1) (PRED pro ?))
  OBJ ((PERS 2) (PRED pro ?))))
```

Valence is defined in a separate file together with lexical rules. It contains the category, lemma, a list of grammatical functions (SUBJ and OBJ, ! means that the GF is mandatory, ? means that it is optional) and corresponding semantic roles (ACT(OR) and PAT(IENT)).

```
(V uñja (! SUBJ ACT) (! OBJ PAT))
```

2.2 Syntax

Many polysynthetic languages are nonconfigurational. Hale (1983) was the first to define and describe nonconfigurationality and its impact to syntax. The general rule which describes the structure of matrix sentences is lexocentric (see (Bresnan, 2001) for more examples):

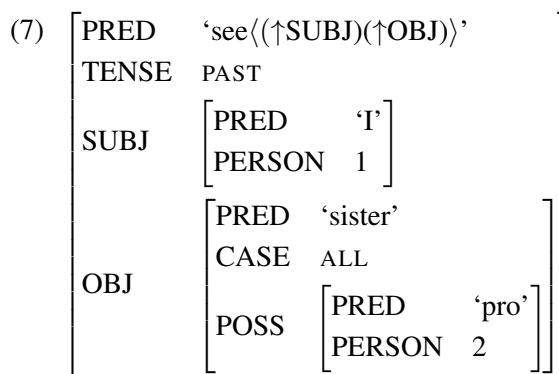
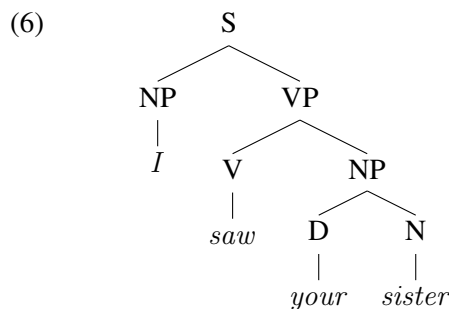
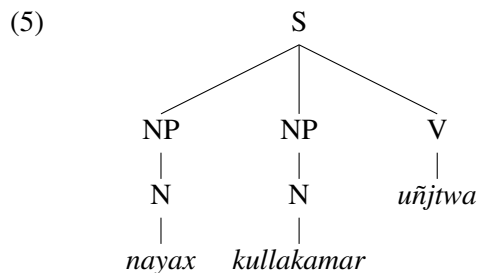
$$(3) \quad S \rightarrow C^+$$

$$\text{where } C \text{ is } \begin{array}{c} V \quad \text{or} \quad NP \mid PP \\ \uparrow=\downarrow \quad (\uparrow GF) =\downarrow \end{array}$$

As an example, compare the c(onstituent)-structure (5) of the Aymara sentence given in (4) with the c-structure (6) of its English translation. The corresponding f-structures (the English one is given in (7)) are structurally identical and differ only in the values of the PRED attributes.

(4) *Naya-x kullaka-ma-r*
 I-TOP sister-POSS2-ALL
uñj-t-wa
 see-SIMPLE_PAST1-FOC
 "I saw your sister."

¹Stem entries are denoted by **r** and contain the form as it occurs in the word (*uñj*), lemma (*uñja*), start and end state in the corresponding finite state automaton (**v1** and **v2** respectively), attribute-value pairs for the f-structure and category for c-structures (**V** for verbs etc.). Suffix entries are denoted by **s** and contain the states of the automaton (**v2** and **tf**), the form of the suffix (*-sma*) and attribute-value pairs for the f-structure associated with the suffix.



As can be seen, the c-structure of the Aymara sentence is flat since the language has no VP. As has been pointed out by Kruijff (2000), phrase structures represent the process of syntactic derivation whereas f-structures (which roughly correspond to dependency trees in dependency-based grammars) are the result of this derivation. Hale (1983) argues that in this kind of languages phrase structures do not encode syntactic relations but only word order.

However, most polysynthetic languages are discourse-configurational and if there are no articles nor other markers which would express information structure, constituency has to be used to analyze topic-focus articulation. So while the lexocentric rule given in (3) is appropriate for the analysis of languages such as Aymara and Quechua because they have topic and focus markers (the suffixes *-x* and *-wa* in (4)), languages like Abkhaz or Guaraní express information structure mainly by word order. We use a set of X' -rules that are very similar to what Meurer (2007) uses for Georgian. The core of the context-free grammar is given in (8).

- (8)
- | | | |
|----|---|-----------------|
| S | → | XP ⁺ |
| I' | → | I(S) |
| IP | → | (XP) I' |
| IP | → | XP IP |

The subtree headed by S belongs to the focus, the verb and the specifier of I' may belong to the topic or to the focus and all XPs adjoined to IP are part of the topic. Independent i(nformation)-structures introduced by King (1997) are used to capture topic-focus articulation.

2.3 Valency

Valency is very important for the correct assignment of grammatical functions. As an example, let us have a look at Abkhaz, an ergative language with transitive and intransitive bivalent verbs that have different morphosyntactic alignment. Compare, for example, the order of personal affixes in (9) and (10).

- (9) *Y-3-6-oum*
 OBJ2SG,MASC-SUBJ1SG-see-PRES
 “I see you.”

- (10) *C-y-c-yeum*
 SUBJ1SG-IOBJ2SG,MASC-hit-PRES
 “I hit you.”

As can be seen, some Abkhaz bivalent intransitive verbs such as *acɣapa* “to hit” are translated as transitive verbs in English. This situation is somewhat similar to oblique objects in Turkish (Çetinoğlu and Butt, 2008). We use a valency lexicon of verbs that contains both grammatical functions and semantic roles. The roles are used in the transfer phase.

3 Transfer

The transfer module can be used for experiments with direct (word-to-word), shallow (NPs and PPs) and deep syntactic transfer. The output of the transfer module can be used to calculate WER (word error rate) if reference translations are available.

3.1 Structural Transfer

In the LFG framework, the transfer module usually operates on f-structures (Kaplan et al., 1989). However, f-structures are too language-specific (cf. the f-structure of (10) with the f-structure of

its English translation which differ in grammatical functions). The inventory and use of grammatical functions is language-specific (there are, for example, languages without secondary objects, with double subjects etc.) which suggests that one should abstract from them and use a more general concept instead. In LFG, a-structures with thematic roles seem to be more suitable for bilingual transfer.

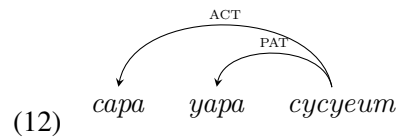
We use deep syntax trees (henceforth DSTs) in the transfer phase. DSTs can be obtained automatically from interlinked c-structures, f-structures, i-structures and a-structures using the following algorithm:

1. F-structures can be interpreted as dependency trees with autosemantic words (i.e., f-structures with the PRED attribute) as nodes and grammatical functions as edge labels.²
2. Annotate the edges of the DST with thematic roles (using the grammatical functions from the f-structure and lexical mapping).
3. Order the nodes using information structure (see (Sgall et al., 1986) for discussion).

Let us use a variant of (10) as an example:

- (11) *Capa yapa*
 I you-2SG,MASC
c-y-c-yeum
 SUBJ1SG-IOBJ2SG,MASC-hit-PRES
 “I hit you.”

Using the algorithm sketched above, the f-structure of (11) yields the DST in (12):



It is obvious that the tree in (12) is identical in Abkhaz and English (except for the PRED values) whereas the f-structures are different (PAT corresponds to OBL_θ in Abkhaz and to OBJ on English).

Table 1 summarizes what various LFG layers contribute to DSTs.

²Generally, we get a directed acyclic graph (DAG). However, edges resulting from structure sharing can be interpreted as coreferences and ignored in DSTs. Formally, we get DSTs from DAGs induced from f-structures as minimum spanning trees. We use Prim’s algorithm where the weight of edges is their distance from the root node.

LFG layer	information in DSTs
c-structure	original word order
f-structure	dependencies and coreferences
i-structure	topic-focus articulation
a-structure	thematic roles

Table 1: Information provided by LFG layers to DSTs

3.2 Lexical Transfer

Having converted f-structures to DSTs, the transfer is mostly lexical, i.e., the PRED values associated with nodes are translated to the target language. Because the translation of many words depends on the context, word sense disambiguation (WSD) is needed. Nonetheless, this is a very complicated problem itself and as a semantic and pragmatic task it is independent of the syntactic framework of LFG or any other rule-based parser.

A very simple and comparatively viable solution is the use of a statistical ranker that selects the most probable translation according to a language model. Thus in our experiments we nondeterministically generate all possible translations and select the best sentence using a trigram based target language model.

A simple bilingual entry for the pair Aymara-English looks as follows:

```
(1 V ((PRED uñja)) ((PRED see)) ())
```

It contains the category (to distinguish between identical word forms with different POS tags, such as *book* in English), a skeletal f-structure for the source language and an f-structure for the target language (most entries contain only the PRED attribute).

4 Conclusions

We have presented a set of tools developed for natural language processing of polysynthetic languages. Examples given in this paper demonstrate several typological features of polysynthetic languages which do not occur in well-researched Western languages and show how we analyze them in the LFG framework.

To test the tools we have developed an MT system from Aymara to Quechua. The WER (word error rate) measured on narrative texts is around 10%. An ongoing experiment with translation

from Aymara to English indicates a WER around 30% but final results are not available yet.

Linguistic resources used in the modules are defined in separate files, there are files for the morphological lexicon, parser rules, valence lexicon (valence frames and lexical rules) and transfer (structural and lexical rules). The code is strictly separated from data. All tools are implemented in portable C++ (using the new C++11 standard and STL) and were tested on Mac OS X (clang/LLVM) and MS Windows (Visual Studio 2010).

References

- Baker, Mark C. 1996. *The Polysynthesis Parameter*. Oxford University Press.
- Bresnan, Joan. 2001. *Lexical-Functional Syntax*. Blackwell Textbooks in Linguistics, New York.
- Çetinoğlu, Özlem and Miriam Butt. 2008. Turkish Non-canonical Objects. In Butt, Miriam and Tracy Holloway King, editors, *Proceedings of the LFG Conference*.
- Hale, Kenneth L. 1983. Warlpiri and the grammar of non-configurational languages. *Natural Language & Linguistic Theory*, 1:5–47.
- Kaplan, Ronald M. and Joan Bresnan. 1982. Lexical-Functional Grammar: A formal system for grammatical representation. In Bresnan, Joan, editor, *Mental Representation of Grammatical Relations*. MIT Press, Cambridge.
- Kaplan, Ronald M., Klaus Netter, Jürgen Wedekind, and Annie Zaenen. 1989. Translation By Structural Correspondences. In *Proceedings of 4th EACL*, pages 272–281.
- King, Tracy Holloway. 1997. Focus Domains and Information-Structure. In Butt, Miriam and Tracy Holloway King, editors, *Proceedings of the LFG Conference*.
- Kruijff, Geert-Kan. 2000. A Dependency-based Grammar. Technical report, Charles University, Prague, Czech Republic.
- Mattissen, Joanna. 2006. On the Ontology and Diachrony of Polysynthesis. In Wunderlich, Dieter, editor, *Advances in the theory of the lexicon*, pages 287–354. Walter de Gruyter, Berlin.
- Meurer, Paul. 2007. A computational grammar for Georgian. In *Proceedings of the 7th International Tbilisi Conference on Logic, Language, and Computation*.
- Sgall, Petr, Eva Hajičová, and Jarmila Panevová. 1986. *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. D. Reider Publishing Company.

EASTIN-CL: A multilingual front-end to a database of Assistive Technology products

Gregor Thurmair

Linguattec
Munich
g.thurmair
@linguatec.de

Andrea Agnoletto

Fondazione Don
Gnocchi Onlus
Milano
aagnoletto
@dongnocchi.it

Valerio Gower

Fondazione Don
Gnocchi Onlus
Milano
vgower
@dongnocchi.it

Roberts Rozis

Tilde
Riga
roberts.rozis
@tilde.lv

Abstract

The document describes an application of language technology to improve the access to a database of Assistive Technology in the EASTIN-CL project. It focuses on engineering aspects of language technology integration. The paper describes the collection of a multilingual terminology database of the domain, and its use in multilingual and multimodal frontend components, especially the design, implementation and test of the query component. The system will be online for public web access under www.eastin.eu¹.

1 Context and Task

Access to information on Assistive Technologies (AT) is a key issue in social participation and eInclusion. The UN Convention on the Rights of Persons with Disabilities declares this as a fundamental right; all UN member states are obliged to comply with this Convention.

To support people with disabilities, the single states have organised web Portals which provide information about Assistive Technology products. Portals are visited by doctors, physiotherapists and other people in the domain.

The information in the AT domain is structured along the lines of the ISO 9999 standard (*Assistive Products for Persons with Disability – Classification and terminology*). This is a classification along functional aspects; AT databases

group relevant products under each heading of this classification.

In 2005 the major European AT information providers joined in the European Assistive Technology Information Network (EASTIN). EASTIN provides a portal (www.eastin.eu) where people can access *all* databases of its national members simultaneously; the central server collects information on all products existing in one of the databases of the associated partners, so information seekers search on European level.



Fig. 1: Searching in national portals using ISO codes

However, variety of languages is still a barrier for easy access to AT information, although the portals provide at least an English translation in addition to the national languages.

To open the scope of this portal for additional user groups (end users), and to support people not familiar with the ISO 9999 classification, and speaking only their native language, a language technology front-end to the EASTIN portal was built in a project called EASTIN-CL. This front-end is supposed to be

- multilingual, i.e. users should forward information requests, and receive results, in their native language, and
- multimodal, i.e. users should be able to use a spoken channel of interaction in addition to the written one, cf. Fig. 2.

Supported languages are Danish, English, Estonian, German, Italian, Latvian, and Lithuanian.

A specific feature of the EASTIN portal is that search is not based on free text but on ISO 9999 codes.

© 2012 European Association for Machine Translation.

¹ This project is partly funded by the European Commission, ICT-PSP no 250432. Partners are Linguattec, Tilde, Fondazione Don Gnocchi / SIVA, Institut der deutschen Wirtschaft / Rehadat, and Danish Centre for Assistive Technology / HMI.

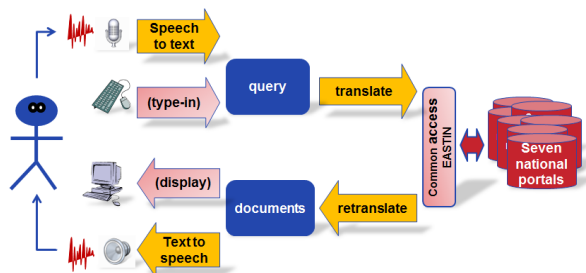


Fig. 2: EASTIN-CL frontend, EASTIN backend

So the approach is not *cross-lingual* as search terms need not to be translated, but *multilingual* as search terms in each single language point to ISO codes which can be used in search.

2 Master Term List

The first task was therefore to collect the concepts which form the AT domain, and to decide which ISO 9999 code they are linked to. The approach taken was to start with one ‘master’ language (English), and translate the resulting term list into other languages. Of course, most of the concepts are multiword terms.

2.1 Selection of Master Terms

The usual way to collect a selection of domain terms is to do corpus analysis, by collecting texts of the domain, and running term extraction tools. This way was tried first, and a list of about 120.000 term candidates was produced. However, this list was not usable, for several reasons:

- It was too large to be translated into seven languages with the resources of the project;
- Most of the high and medium frequency candidates were not specific enough to be included into the domain term list (i.e. assign a ISO 9999 code to them)
- In turn, many of the domain specific terms did not occur in the candidate list altogether.

So, the terms retrieved were not really suitable, and many good terms were not retrieved.

As a result, the approach was changed, and the domain terms were collected from existing descriptor lists: Many AT information providers, like Abledata, Rehadat etc.², offer key term lists for searchers, and so does the ISO 9999 classification itself. It was decided to base the domain terminology on such key terms, and to merge them into a common resource, resulting in a candidate list of about 17.000 terms.

Merging revealed that the different information providers had different strategies of key

term denomination: Some presented them in plural form (*‘wheelchairs’*), others in singular; some used US, others UK spelling, etc. As a result, the master list contained many pseudo-doublets. A cleanup step was needed based on principles like: use singular form as in paper dictionaries; use UK spelling (*‘tyres for wheelchairs’* instead of *‘tires for wheelchairs’*), use one term to describe one concept; i.e. split *‘backrest (bath/shower)’* into two entries; use hyphens only for particles (*‘dial-up’*) or objects of participles (*‘author-based’*)

Even after cleanup, there is significant variance in the denominations. The final list contains about 12.700 concepts, with part-of-speech annotations.

2.2 Creation of the Domain Classification

All concepts should be linked to a domain ontology. In the case of AT, the domain is structured by the ISO 9999 classification³, a three-level classification with about 800 nodes overall.

All terms of the master list were assigned one or several ISO codes; so the list forms a ‘light-weight ontology’ of the AT domain. It is expected that the term list will be fine-tuned during the test and use of the system.

2.3 Multilinguality

The task of creating the domain terminology was completed by translating the master term list into the seven languages of the EASTIN-CL partners.

Translations were carried out by domain experts, and in unclear cases the product databases could be consulted to find the best translation.

The resulting term list contains all 12.700 concepts, expressed in seven languages, about 90.000 terms altogether. This list was converted into the TBX standard, and is also offered for online access on the EASTIN-CL website.

3 Indexing and Search Preparation

3.1 Approach

In searches containing multiword terms, two indexing strategies are possible⁴: *pre-coordination* collects multiwords before searching; and *post-coordination* collects them afterwards (usually by *AND*-ing the single elements).

Nearly all search engines use post-coordination; however it can easily be seen that in multi- and crosslingual contexts, multiword terms must be

² www.abledata.com, www.rehadat.de, www.hmi.dk

³ ISO 9999: 2011

⁴ cf. Buder et al. 1990

recognised beforehand, as they may need a specific translation: if the parts of ‘*stuffed bag seat*’ are each translated in isolation, the correct German translation into ‘*Sitzsack*’ will not be found, and search results will suffer from this mistake⁵. In EASTIN-CL, the index contains multiwords, so pre-coordination is selected for indexing.

3.2 Index creation

Given the large variety in the term representations, the index terms must be considered as the target of a normalisation step, covering as many search term variants as possible.

The index in EASTIN-CL contains four fields: 1. the term in its *display form*, as it is presented to end users; 2. the term in its *normalised form*; 3. the single parts of the term as a sequence of *base forms* (lemmata), and 4. the *ISO code(s)* assigned to the term to find the real documents.

The representation of a multiword term as a list of lemmata requires lemmatisers and decomposers for the seven languages involved; tools by Linguatex and Tilde were used for this.

These tools had to be adapted to the AT domain (to decompose terms like ‘*Thorako llumballorthese*’, which would match a query for ‘*lumbale Orthese*’).

3.3 Search preparation

The query processing component must analyse the query text; it needs language resources for this. In EASTIN-CL, two considerations influenced the design of these resources: 1. It is a *runtime* component, i.e. it is time and resource critical. 2. The EASTIN *target vocabulary* is limited, and basically a fixed set: Not *all* input words but only the words of the term list need to be recognized.

Therefore, a ‘static lemmatiser’ and a ‘static’ decomposer resource were implemented, whereby in a fixed lexical resource inflected forms point to their lemma, or word parts.

4 Search

Search in EASTIN-CL consists of three steps: query analysis and translation, search proper, and result retranslation.

4.1 Query analysis

Query Analysis must map a query input to an index term. The index term is annotated with ISO 9999 codes, pointing to groups of AT products.

While the EASTIN portal is responsible for a distributed access to the national AT product databases, the frontend is responsible to produce ISO codes for searching, cf. Fig. 3.

For the language of the search, query analysis does tokenization, normalization, and lemmatisation and decomposition, by looking up the word form in the static resources. Finally, all index terms containing the single words of the query are retrieved as search candidates.

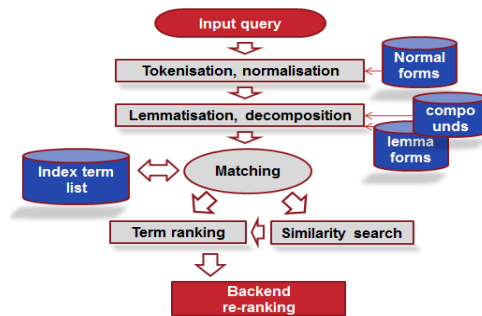


Fig. 3: Query Analysis. Resources needed: lemmatisers, decomposers, normalisers, in 7 languages

If no hit is found (typing errors), a fallback distance-based similarity search is used.

The final step of search is ranking the candidate terms. Ranking is based on the number of words in the query, the number of words of the index terms, and the number of matching terms. The terms with the highest overlap of matching terms are considered to be the best. The result is mapped on a 5-point scale, and the best ranked terms are returned with their ISO codes.

4.2 Searching

The search backend takes the candidate list of the query processing, and re-orders them as follows:

While the query processing takes care of the best matching *index term*, the main search intention is to find the best group of products, i.e. the best matching *ISO codes*.

Therefore the term list produced by the query is re-ranked based on term ranks and the ISO codes found, and the highest ranked ISO code (not necessarily the highest ranked term) is used for searching. This makes the system more robust. The search interface displays which term contributed to which ISO code (cf. Fig. 4).

To avoid a situation where users find no hits, the EASTIN portal offers additional search options, like search by navigation in the ISO classification; search for products (‘*Tigges-Lumbal-orthese*’) or manufacturers (‘*All Terrain Wheel-chairs Ltd*’) with the search term in their name.

⁵ cf. [self-cite]

Europäisches Netzwerk mit Informationen zu technischen Hilfsmitteln

gastin

Suche Was EASTIN ist Die EASTIN Partner Allgemeine Informationen

Suche → Zusammenfassung der Ergebnisse der Freitextsuche

Zusammenfassung der Ergebnisse der Freitextsuche - Suche

Ihre Suche nach "Lumbalorthese" brachte das folgende Ergebnis:

Produktgruppen: 4

- ★★★★★ Lumbo-sakrale Orthesen - ISO-Nummer: 06.03.06 - (253 Produkte)
Gefundene Schlagworte: Lumbalorthese, lumbale Orthese, Lumbalstützorthese
- ★★★★★ Lumbale Orthesen - ISO-Nummer: 06.03.04 - (8 Produkte)
Gefundene Schlagworte: Lumbalorthese, lumbale Orthese
- ★★★★★ Thorako-lumbale Orthesen - ISO-Nummer: 06.03.08 - (5 Produkte)
Gefundene Schlagworte: Thorakolumbalorthese, thorako-lumbale Orthese, Lumbalstützorthese

alle Ergebnisse ansehen...

Produkte, die das Suchwort enthalten "Lumbalorthese": 3

- Tigges-Lumbalorthese nach Krämer (2-Stufen-Therapie)
Hersteller: OZO-Zours GmbH
- T-Flex TL nach Krämer, Thorako-Lumbalorthese mit Auf-/Abbausystem
Hersteller: OZO-Zours GmbH

Fig. 4: Search in the portal for 'Lumbalorthese': Search term produces a list of ISO codes, with the terms which retrieved them underneath. Ranking is given with stars.

4.3 Retranslation

Result of a search is a list of products, grouped under a given ISO code. The product descriptions in the national EASTIN databases are stored in the national language, and in English. The multilingual front-end now must re-translate the product descriptions into the query language. This translation is done on-the-fly: The EASTIN server accesses MT web services to translate the product descriptions. Both rule-based (Linguatex's 'Personal Translator' English-> German/Italian) and SMT systems (Tilde's 'Let'sMT!' platform, English->Baltic languages) are used. The MT systems were tuned for the AT domain, using the master term list and additional corpus data. Subject of translation are the textual parts of the product descriptions.

5 Evaluation

The objective of the evaluation was to find out:

If (end) users search for a certain AT product, which query terms do they really use? How good does the terminology provided match the search interests and search profiles of the users?

5.1 Evaluation approach

Two types of tests were designed:

The first test is a test on terminology. About 100 pictures of AT products were selected randomly, and put online, asking users to enter the terms they would use to search for the type of products depicted on them. Users can input queries, which are analysed to find out if the terms used point to the right product group.

This procedure avoids to influence users by proposing terms, and allows to verify if the terminology provided by the EASTIN components is intuitive and of good coverage.

The second one is a test on usability. Users are given little tasks, and their interaction behaviour is evaluated with questionnaires: Does their search succeed? Which search tool do they use? Is MT of any help? etc.

5.2 Test Results

Tests of the term selection for pictures showed that users use terms which are recognised, and therefore lead to the right product group, in the majority of the cases (> 60%, with slight differences in the different languages); this emphasizes the good coverage of the term list. Error analysis showed that this result can be further improved by adding synonyms to the term list.

Preliminary results of the usability tests, performed with about 60 external users, show a significant increase in the acceptance of the system, mainly due to the query functionality, but also to the machine translation and speech interaction components.

Overall, the language technology front-end components are considered to be a significant improvement in the accessibility of the Assistive Technology provided by the EASTIN portal.

References

- Andrich R., 2011: Towards a global information network: the European Assistive Technology Information Network and the World Alliance of AT Information Providers, In: G.J. Gelderblom et al. (eds): Everyday technology for independence and care. pp. 190-197. IosPress
- Buder, M., Rehfeld, W., Seeger, Th., eds., 1990: Grundlagen der praktischen Information und Dokumentation. Saur.
- International standard ISO 9999:2011, Assistive Products for Persons with Disability – Classification and terminology
- Lyhne Th., 2011: The Danish National Database on Assistive Technology, In: G.J. Gelderblom et al. (eds): Everyday technology for independence and care. pp. 205-213. IosPress
- United Nations, 2007: The UN Convention on the Rights of People with Disabilities. In www.un.org/disabilities/
- Winkelmann P., 2011. REHADAT: The German information system on assistive devices, In: G.J. Gelderblom et al. (eds): Everyday technology for independence and care. pp. 205-213. IosPress

Towards the Integration of MT into a LSP Translation Workflow

David Vilar¹, Michael Schneider², Aljoscha Burchardt¹, and Thomas Wedde²

¹DFKI, Language Technology Lab, Berlin, Germany {name.surname}@dfki.de

²beo GmbH, Stuttgart, Germany {name.surname}@beo-doc.de

Abstract

This user study reports on an ongoing pilot that aims at using machine translation on a large scale, for the translation of technical documentation for a globally acting automotive supplier. The pilot is conducted by a language service provider and a research institution. First results go beyond expectations.

1 Introduction

In real-world translation environments efficiency, both in terms of cost and time, is of critical importance. Even more when the volume of texts to translate is large. Machine translation (MT) seems to be a good candidate for achieving these goals, but somehow surprisingly the economic feasibility of MT and the fitness for real-world needs of professional translators and Language Service Providers (LSPs) have been hardly analysed so far.

The MT community tries to broaden the domains the translation systems are applied to. In the early years, research on statistical machine translation concentrated on restricted domains, the touristic domain being a typical example. As the quality of the translations got better, the difficulty of the task was increased by moving to richer domains. The WMT evaluations are another example of this trend. In the first editions (Koehn and Monz, 2005) the data the systems were trained and evaluated on consisted only of the proceedings of the European Parliament. In more recent editions (Callison-Burch et al., 2011) the (parallel) training data still mostly consists of europarl data, but the evaluation has moved to the news domain, with a much wider variety of topics.

The goal of this research direction is clear: to produce an “universal translator” that is able to translate any type of text. This is however a very optimistic goal and current systems are still very far from it. And it also may not be the optimal goal for professional translators. When a LSP has a translation request, it is usually accompanied by guidelines of style, vocabulary, etc. Also, the domain is usually quite restricted. Not much topic variation can be expected from, say, user manuals of heavy machinery.

As such, the research community has perhaps overlooked a potential niche where machine translation, in its current development state, can prove to be beneficial. At the same time, potential customers are reluctant when it comes to financing the development of specialised MT engines as their idea is that MT comes for free. In this paper we present a pilot study where we analyze how a state-of-the-art machine translation system performs in a real-life environment. The work is a collaboration between a LSP (beo) providing the experience on real-life translation tasks for Bosch, and a research institution (DFKI) providing the know-how about statistical machine translation.

The paper is structured as follows: Section 2 presents the viewpoint of the LSP on the translation task, as well as the expectation of a machine translation system to be considered useful in their workflow. Section 3 describes the machine translation system adapted for the task. Conclusions are drawn in Section 4.

2 From TM to MT: The LSP’s starting point

A LSP always has to keep a good balance between prices, linguistic quality, and time, all for the benefit of the client. Especially in the area of train-

ing material the price pressure is even higher than normal, because professional translation of slides used (internally) for training is routinely omitted. Often the material is created newly in foreign languages if needed, leading to significant differences in content and quality.

The Bosch Automotive Aftermarket department (AA) decided relatively early to have training material translated to keep at least the content consistent among language versions. Price was (and is) still important: translation costs are traditionally not shared with the trainees, and were in fact in the past not part of the training budget.

The net effect was that most of the material did not get translated at all, and if so, without consistently controlled quality.

This was the point when Bosch asked beo to take over these translation tasks. Being a “preferred supplier for the Bosch group” for translation services, it was expected that beo

- keeps the price per word low, at a level of about 70% of the normal word price
- reduces the turnaround time for translations, from 3-4 Months down to ca. 2-4 weeks per unit
- raises the overall quality.

Of course, Translation Memory Technology (TM) was to be used, which helps to keep translations consistent over time and to control terminology and overall quality. But the price pressure is still on: More and more clients are not willing to pay for translation proposals coming from perfect (100%) TM matches. Still, these “synthetic” translations need to be proof read and quality checked by the translator and thus require (paid) work.

It was quite quickly clear to us that a TM alone would not be sufficient to reach the goals, especially the cost limits. When communicating such troubles to clients a common reflex is “why don’t you use machine translation?”, with the implicit assumption that MT is essentially available for free¹ and with sufficient quality to be used unchecked.

Not so with Bosch. It was known that automatic translations had to go through some sort of quality

control, and that MT itself is not free of cost (license costs, machine time, etc.). In this context we came to an agreement to use these training materials as a test case for a pilot project to integrate MT into a professional translation workflow.

The core requirements for this workflow are:

- integration of MT into a traditional TM environment. The translator should be able to use the tools and environment he is accustomed to, to keep productivity high
- no “post editing” of MT results at a large scale. Post editing poses new resource problems as there are usually not enough “post editors” at hand, and they will probably not work for free. . . Therefore the precedence is translation memory over machine translation over translate from scratch.
- “break even” point for translation costs reachable after roughly 10 months

beo’s previous experience with machine translation is limited to post-editing jobs. High volume post editing jobs for different clients lead to the insight that post editing performed as an extra work step is neither cost effective nor a guarantee for good quality. Thus, the objective of the project is to integrate MT in such a way that automatically translated content is “magically” presented to the translator just like a TM match. The translator then is responsible to accept, change or reject the translation, just like a TM match. Standard quality assurance work steps and tools can be applied, the MT is seamlessly integrated into the standard translation workflow along the TM.

3 Training an MT engine

In order to train a translation model, DFKI first had to prepare the data into a format suitable for the translation system. The original format is composed of slides translated from an original language (German) into a target language (in our experiments English and Spanish). The slides themselves could be considered as the translation unit, but we chose to work with sentence-like units. For this we firstly applied an automatic sentence splitting tool, and then proceeded to re-align the produced sentences with the Microsoft bilingual sentence aligner (Moore, 2002).

After some cleanup of the data, including removal of duplicate sentences, a special categorization step has been applied to detect tokens that can

¹In many (all?) cases “machine translation” is the same as “Google Translate” in the view of the clients.

Set	DE-EN		DE-ES	
	Segm.	Words	Segm.	Words
Original	203K	7.5M	199K	7.3M
Train	402K	3.3M	400K	3M
Dev	2 086	17 746	993	7 790
Test	2 057	16 774	1 008	8 597

Table 1: Statistics of the random split into training, development and test sets. The number of segments in the original data corresponds to slides, in the train, dev and test sets, to sentence-like units.

be directly carried over from the source language to the target language. These categories include numerical quantities, in-text references (“see Table x ”, legend of Figures, etc.) which are specially marked in the text as well as some formatting information (most notably tabular alignments).

A random split into training, development and test data was carried out. Table 1 shows the statistics of the resulting sets. As can already be seen from these statistics, the data is highly redundant. The number of segments is greatly increased when comparing the original data with the preprocessed data (train, dev and test sets), due to the sentence splitting. On the other hand the number of words is less than half, due to the removal of duplicates.

On this data a phrase-based statistical machine translation system was trained (Zens et al., 2002). We chose the Jane translation toolkit (Vilar et al., 2010) over the more widely known Moses toolkit (Koehn et al., 2007) due to its ability of handling the categories described above.² The results in terms of BLEU score are given in Table 2. As can be seen, the scores are very high, around 64%. To give a comparison, the highest scoring system in the 2011 WMT Evaluation Task scored 25% BLEU on the German-English task. For English-Spanish (there was no German-Spanish task) the best scoring system achieves a BLEU score of 35% (Callison-Burch et al., 2011).

The reason of our exceptionally good results lies of course in the nature of the data. As was pointed out before, by its nature the data is highly repetitive, even with sentence duplications removed.³

²A short note about licensing: Jane is freely available for non-commercial use. At the current stage this study is still of scientific nature. Should a commercial application arise, the licensing issue will have to be reconsidered.

³Without removal of duplicated sentences the scores go over 70% BLEU.

Language Pair	BLEU[%]
German-English	64.2
German-Spanish	63.9

Table 2: Results in terms of BLEU score on the test set.

Figure 1 shows some example translations. The first one shows an example sentence where the translation system achieved a perfect translation. The structure of this sentence allows for easy generalization (think of several connector colors) and also shows the categorization carried out when pre-processing the data, where the system detected a number and a reference.

The performance of the system is also quite good for more complicated sentences, as the second example of Figure 1 shows. Although it may sound a bit artificial at first sight due to the repetition of “side” towards the end of the sentence, the automatic translation is actually more accurate than the reference translation and in a technical domain like the one we are dealing with it may be fully acceptable.

Of course not all the translations are good, as the third example shows. Although to be fair to the translation system, this sentence does not fully conform to Bosch’s style guidelines (the passive voice should be avoided).

4 Outlook & Conclusions

We have presented a user study of applicability of (statistical) machine translation to a real-life translation task as requested from a LSP. The quality of the resulting translations is very high, well beyond our initial expectations. We consider that the quality is good enough to step to the next phase of the project, integrating the translation system into the human translator’s workflow. The goal will be to complement the currently used translation memories, which have proven to be of great assistance to the translator’s work. A straightforward application will be to use the translation system when the match of the translation memory is not good enough, but more complex interactions will be considered in a further study.

In the current study machine translation’s flexibility to translate phrases like “see Figure 5” even if the number “5” did not occur in the training data has already proven helpful as compared to standard

Source	- Anschlussstecker schwarz (Kl . \$number { 31 }) an Buchse \$ref { <1> }
Translation	- Black connector (term . 31) to socket <1>
Reference	- Black connector (term . 31) to socket <1>
Source	Werden Sollwerte erreicht , liegt ein Defekt im Airbag-Steuergerät oder im Seitenaufprall-Sensor Beifahrerseite vor .
Translation	If set values are attained , there is a fault in the airbag control unit or in the passenger 's side side impact sensor .
Reference	Airbag control unit or front passenger 's side impact sensor is defective if set values are attained .
Source	Konstruktionsbedingt können auch bei abgebautem Steuergerät keine Wicklungswiderstände gemessen werden .
Translation	The design may also be detached control unit is not winding resistances be measured .
Reference	The design is such that it is not possible to measure winding resistances even with the control unit detached .

Figure 1: Translation examples.

translation memories that present a fuzzy match in these cases. One example of a more complex interaction would be to use machine translation systems for ranking multiple 100%-matches of a translation memory according to plausibility, possibly taking context into account. Once confidence estimations of machine translation systems will get more reliable, human post-editors can be presented only material that needs to be touched or error checked.

Although BLEU scores and inspection of the translations may give a good overview of the translation quality, the final performance test will be of course to measure human performance when using the developed system. The final goal is to improve the efficiency of the whole translation pipeline.

This study may also serve as a hint for the machine translation community. The goal of creating machine translation systems that are capable of dealing with a very wide domain is certainly appealing, but ignoring smaller domains may miss important applications. Our results may seem non-conclusive to some researchers (“too similar training and test data”), but we are dealing with *real-life* data, provided by a LSP. The fact that translation memories are the most widely used computer aid by human translators is an indication that such conditions are realistic.

References

Callison-Burch, Chris, Philipp Koehn, Christof Monz, and Omar Zaidan. 2011. Findings of the 2011 work-

shop on statistical machine translation. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 22–64, Edinburgh, Scotland, July. Association for Computational Linguistics.

Koehn, Philipp and Christof Monz. 2005. Shared task: Statistical machine translation between European languages. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 119–124, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 177–180, Prague, Czech Republic, June.

Moore, Robert. 2002. Fast and accurate sentence alignment of bilingual corpora. In Richardson, Stephen, editor, *Machine Translation: From Research to Real Users*, volume 2499 of *Lecture Notes in Computer Science*, pages 135–144. Springer Berlin / Heidelberg. 10.1007/3-540-45820-4_14.

Vilar, David, Daniel Stein, Matthias Huck, and Hermann Ney. 2010. Jane: Open Source Hierarchical Translation, Extended with Reordering and Lexicon Models. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 262–270, Uppsala, Sweden, July. Association for Computational Linguistics.

Zens, Richard, Franz Josef Och, and Hermann Ney. 2002. Phrase-Based Statistical Machine Translation. In *German Conference on Artificial Intelligence*, pages 18–32, Aachen, Germany, September.

Context-Aware Machine Translation for Software Localization

Victor Muntés-Mulero

Patricia Paladini Adell

CA Technologies

WTC Almeda Park

08940 Cornellà de Llobregat, Barcelona

Victor.Muntes@ca.com

Patricia.PaladiniAdell@ca.com

Cristina España-Bonet

Lluís Màrquez

TALP Research Center

Universitat Politècnica de Catalunya

crisinae@lsi.upc.edu,

lluism@lsi.upc.edu

Abstract

Software localization requires translating short text strings appearing in user interfaces (UI) into several languages. These strings are usually unrelated to the other strings in the UI. Due to the lack of semantic context, many ambiguity problems cannot be solved during translation. However, UI are composed of several visual components to which text strings are associated. Although this association might be very valuable for word disambiguation, it has not been exploited. In this paper, we present the problem of lack of context awareness for UI localization, providing real examples and identifying the main research challenges.

1 Introduction

Due to the rapid and worldwide development of Internet and IT applications, fast software localization is becoming essential, requiring *user interfaces* (UI) to be translated to different languages. One of the main obstacles when translating UI is the *word sense disambiguation* problem since strings are usually independent from other strings in the UI and, therefore, it is not possible to infer semantic information from other parts of the text.

In this paper, we want to show that the meaning of a string in this environment varies depending on its position in the UI. For instance, a word associated to a *menu* may be interpreted as a name, but it may be an action if the same word appears on a *button*. Although enriching the translation process with this alternative contextual information would benefit quality, previous software localization techniques ignore, in general, this approach.

It is commonly accepted that the number of words processed per day by a human translator is significantly increased when an efficient machine translation (MT) engine is used and human translators intervene in the post-editing phase. Specifically, it is becoming popular to use MT engines for software localization. Unfortunately, even if contextual information about the UI components associated to strings was gathered, current localization procedures using MT engines are not devised to absorb and exploit it to improve MT quality. Visually aided translation tools, like Passolo¹ or Catalyst², leverage contextual information and show it graphically to human translators. However, they depend on specific file formats which are not always available. Improving the quality of the output of MT allows both (i) to reduce the cost of translation by increasing translator's throughput by up to 50%, based on CA Technologies³ experience, and (ii) to reduce the delay to market of the software products. The objective is simultaneous shipment.

The main contributions of this paper are as follows. Section 2 describes the most relevant state of the art. In Section 3, we define the problem of the lack of contextual information in UI localization providing examples extracted from real products of CA Technologies. In Section 4, we enumerate the main research challenges in terms of improving the quality of the output of MT by increasing the context awareness for UI localization. Finally, Section 5 concludes this paper.

¹www.passolo.com

²www.alchemysoftware.ie

³CA Technologies is a worldwide software and solutions provider that helps customers to make ICT management more agile, secure and flexible. The company localizes many of its applications to several languages, using MT techniques and human post-editing.

2 Previous Work

Incorporating MT in the software localization process has been the focus of recent projects. For instance, Ruopp (2010) adapts well-known open source translation engines. Also, Hudik and Ruopp (2011) integrate them into computer-aided translation tools. However, to our knowledge, none of these previous works make use of context extracted from UI.

In general, most MT systems, translate text sentence by sentence independently, ignoring broader contextual information. Even at sentence level, a statistical system based on segments or phrases (phrase-based SMT, (Koehn et al., 2003)) uses the source lexical context of phrases only locally, considering a limited number of words next to the phrase being translated. Because of this, the discourse at document level is not considered.

Syntax-based SMT (Chiang (2005) among many others) tries to alleviate the lack of connection between long distance phrases by considering syntactic dependencies, still within a sentence. Also, factored models (Koehn and Hoang, 2007) include linguistic information in phrase-based models as extra factors associated to words. This information can be anything that can be codified, although the most extended use is to employ morphology to generate translations from the lemmatized text. An alternative way to consider context is by using word sense disambiguation techniques to choose between possible translations of a word or a phrase. In general, these approaches use machine learning methods to learn an adequate word selection model (see for example Giménez and Màrquez (2008)). None of these advances in standard phrase-based SMT tackles the context-aware problem in UI translation.

3 Lack of Context Awareness

In this section, we describe the overall process of UI software localization in an industrial environment and describe the problem of the lack of context in UI translation.

3.1 UI Software Localization Process

Figure 1 depicts a high level overview of the process for UI localization used at CA Technologies. A first common aspect that is important to remark is that, especially in large enterprises, programming and localizing are not only performed by separate human teams, but this work is usually done in different departments in very large and complex

development organizations. As a consequence, in many cases direct collaboration between them is not straightforward due to different time zones or due to the fact that they might be using different and complex tools, highly specialized for their day-to-day tasks. Even worse, it is common that some of the UI to be translated might be coming from recently acquired software or part of the localization might be outsourced to third-parties. In addition, the skills and expertise of developers and translators are usually completely different. While developers are not expected to have comprehensive English language skills, translators are not supposed to interpret the source code of applications. As a result, development and localization are usually decoupled, their interaction is in general very complex and, in addition, translators rarely have access to the source code.

Usually, different tools are provided to help developers generating code which is compliant with internationalization requirements (step 1 in Figure 1). These tools are devised to ensure, for instance, that text appearing in the code adheres to the basic formatting rules, required in the localization process to digest and translate the text properly. Once a new product or release is ready, the source code is parsed and the text in the UI is extracted for localization (step 2 in Figure 1). First, text is run against translation memories in order to leverage previous translations (step 3). Second, the remaining strings are run through MT engines to obtain a machine translated output in the target language (step 4) that will be post-edited by human translators (step 5). This is one of the most time-consuming steps in the localization process since it consists in manually (or semi-automatically) editing the MT engine outputs in order to produce publishable content. The output is then passed to an automatic tool that prepares the new translated text to be inserted back to the original source code (step 6). Because in many cases human translators do not have access to any view of either the final layout or the source code of large and complex application, and therefore the UI components where each string is associated, they cannot guarantee a correct translation. As a consequence, it becomes necessary to perform a critical iterative step that we call *Language Quality Assurance* (LQA). This process is usually highly resource-consuming and requires programmers to generate a sample of evidences, such as screenshots, to allow translators to validate the translation in context. If errors are reported, they have to be solved by developers in an iterative

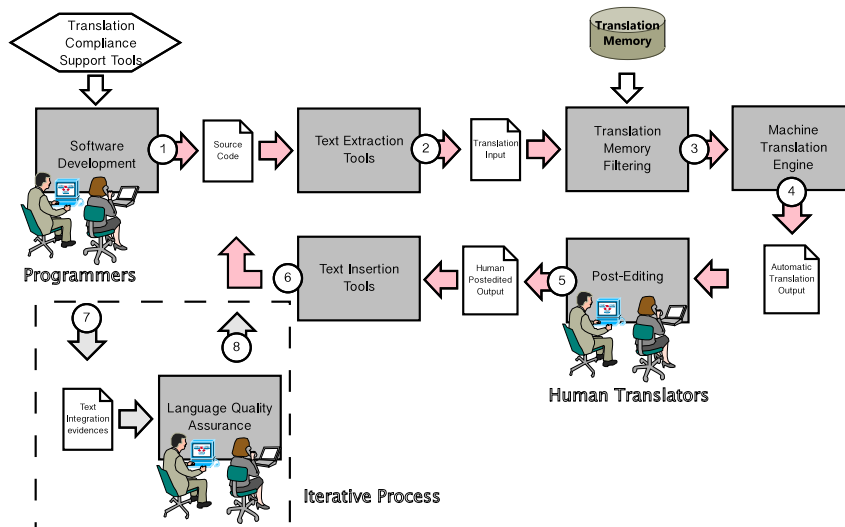


Figure 1: UI Software Internationalization and Localization Process at CA Technologies

procedure. This costly process is very inefficient and, therefore, expensive.

3.2 Context Description and Examples of Lack of Context

We define context as the minimum required information needed to solve an ambiguity. This contextual information should be added to the raw strings sent for translation. We classify the different types of ambiguities found in our scenario in four different categories: (i) *part of speech*: this is one of the ambiguities requiring solution and it is needed in order to provide an accurate translation (Figure

2.a). In this example, the source text was “Access”, which in English can be a verb (to access) or a noun (an access). However, as the text is embedded into a button, it has to be translated as a verb, in this case “Accedi” (verb in Italian) instead of “Accesso” (noun in Italian). In most cases, this ambiguity can also be solved by providing the UI element in which the text will be showing up (a button, a menu, a dialog box header, a drop down list, etc.); (ii) *gender*: this is the most difficult ambiguity to solve as the gender will always depend on a different element. For example, in some languages, the gender of a word included in a table cell will depend on the gender of the column header (Figure 2.b). In the example, you can see two overlapped ambiguities: first, the original English word “Open” could be a verb or an adjective, and it has been translated automatically as a verb (“Ouvrir”) while it should be translated as an adjective (“Ouvert”) and, second, the gender of the adjective⁴ will depend on the gender of the title in the column header (in English “Request Status” is translated into French as “Statut de la demande” which is feminine), in this case, “Ouverte”; (iii) *prepositions*: prepositions like “to” or “from” always need context information for disambiguation (Figure 2.c). For example, the word “to” has at least three possible interpretations: destination, recipient or date, and in Spanish this would be translated to “a”, “para” and “hasta”, respectively; (iv) *syntactic ambiguity*: caused by word order in English: “Display Unit” might be translated to “Unità

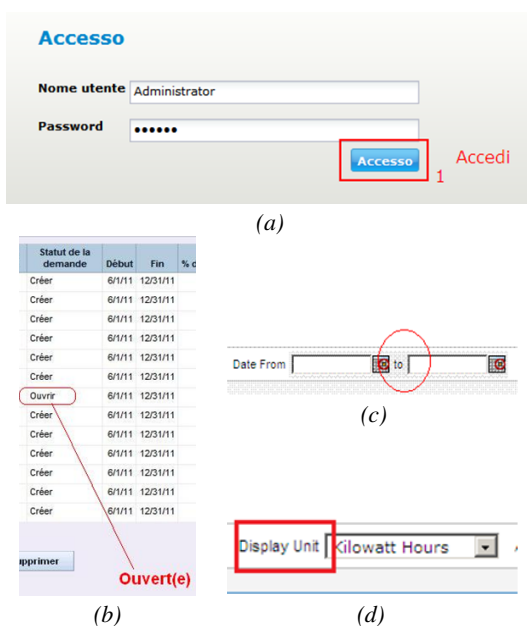


Figure 2: Examples of lack of context effects

⁴Note that adjectives in roman languages are affected by number and gender.

di visualizzazione” (unit to be displayed) or “*Visualizza unità*” (to display a unit) (Figure 2.d).

4 Research Challenges

In this section we summarize the main challenges posed by the ambiguities identified.

Adapting MT engines to exploit contextual information: MT engines must be improved in order to handle UI contextual information and improve quality. As a first approximation, we envisage writing a set of rules. This way the system is informed so that it translates, for instance, an ambiguous word as a noun in a menu and as a verb in a button. The validity of this approach depends on the degree of ambiguity and the coverage of the rules. A more competitive method could be adding the context as factors in phrase-based SMT. This way, it is different to translate (*archive, noun, menu*) from (*archive, verb, button*) or (*archive, noun, button*). Within this framework the translation is not selected by a rule, but each alternative translation has a probability estimated from frequencies in a corpus, and the translation of a word is also conditioned by the translation of the neighboring words.

Although the move towards a probabilistic approach ensures a high coverage, it might not be enough to solve some kinds of ambiguities. The information needed to properly translate the gender and number of a text might be encoded by several types of context at the same time, so it is necessary to deal with a high number of features. Factors are not appropriate for a large number of features, but machine learning techniques can be used to learn the best translation according to its context codified with those features. This methodology has been already successful as stated in Section 2.

Context extraction and internationalization-compliant programming standards: all of these approaches assume that the context can be extracted from the code. Besides, those which rely on statistical methods also need to gather an annotated corpus. A second line of research, thus, will involve establishing automatic methods to extract context information from the source code. Challenges range from parsing the information of complex UI components, such as tables, where content in the table header might affect the translation of the text in the cells for instance, to defining programming standards that make the code compliant with localization needs or creating new tools that aid developers to use writing style guidelines that make the localization process easier.

There may be several ways of including contextual information in the files sent for translation: (a) to include information of the UI components in which the ambiguous text will be embedded, next to ambiguous words; (b) for recurrent pre-defined ambiguities like “*To*” and “*From*”, provide a pre-defined standard explanation of the context, like for example, “*To*” as in “*date*”, or “*To*” as in “*e-mail*”; (c) in case none of the previous options works, to allow for a free text option to provide information necessary for disambiguation. Any of these possible solutions require establishing practical methods that do not overload developers with unnecessary extra work and, specially in the last case, sophisticated methods to extract information from free text.

5 Conclusions

Specializing MT engines used in software localization processes is vital for reducing costs both in terms of time and budget. However, to our knowledge, the problem has not been tackled yet. Reducing the mistranslations produced by the lack of context will have a direct impact on both the post-editing phase and the LQA phase, which are the most costly phases in such a process. Therefore, it is necessary to include UI components information in the localization process. We strongly believe that near future research efforts should be pointed towards these types of solution.

References

- Chiang, D. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 263–270, June.
- Giménez, J. and L. Màrquez, 2008. *Discriminative Phrase Selection for SMT*, pages 205–236. NIPS Workshop Series. MIT Press.
- Hudik, T. and A. Ruopp. 2011. The integration of mooses into localization industry. In *15th Annual Conference of the EAMT*, pages 47–53.
- Koehn, P. and H. Hoang. 2007. Factored Translation Models. In *Proceedings of the Conference on EMNLP*, pages 868–876.
- Koehn, P., F. J. Och, and D. Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the HLT/NAACL*, Edmonton, Canada, May 27-June 1.
- Ruopp, A. 2010. The mooses for localization open source project. In *Conference of the AMTA*, October.



VIRTUS™

Virtus: Translation for Structured Data

Mark Arehart, mark.arehart@ntrepidcorp.com

Ntrepid Corporation

www.virtustranslator.com

Description

This paper presents Virtus Translator¹, a new machine translation (MT) product developed by Ntrepid Corporation. Virtus is a specialized tool designed for maximum performance on structured data, such as found in spreadsheets and database tables. The Virtus engine uses statistical machine learning techniques to identify the language and category of each column in a table. The categories for which Virtus is optimized include names of people, locations and organizations as well as supplemental information such as job title, education level, nationality and religion. Based on the automatically detected language and category, Virtus recommends a translation strategy suitable to maximize performance on each type of data.

Virtus translation strategies represent an innovation in translation software that provides users unprecedented control over the handling of their structured data. Virtus strategies range in sophistication from a "do not translate" approach suitable for internet addresses and certain numeric data, to a range of linguistically appropriate translation techniques. For example, a "transliteration only" strategy might be used for person names, while street addresses would be better handled by keyword translation supplemented by automatic transliteration. More sophisticated translation strategies are recommended for more linguistically complex categories such as organization names and job titles. Knowing the category of the data in each column allows not only for intelligent algorithm selection, but also enables more accurate translation of category-specific terminology, including acronym and abbreviation expansion.

The Virtus translation engine can run in fully automatic mode; however, it is also highly customizable, allowing users to specify translation parameters for each column of data, controlling the language, the category, the translation strategy and the transliteration standard on a per-column basis. Also, recognizing the uniqueness of each user's translation requirements, Virtus provides a custom translation memory (TM) building tool to allow sophisticated users fine-grained control over Virtus Translator's output.

Virtus Translator for Chinese, Persian and Russian are currently available. Virtus Translator for Arabic, French, German and Spanish are scheduled for release in 2012. A 30-day trial version is currently available upon request.

¹ Patent Pending 12/461,574



MOLTO Enlarged EU – Multilingual On-Line Translation

EU

FP7/2007-2013

Small or medium-scale focused research project

Grant agreement No. 288317

<http://www.molto-project.eu>

List of partners
Göteborgs Universitet, Sweden (coordinator)
Helsingin yliopisto, Finland
Universitat Politècnica de Catalunya, Spain
Ontotext AD, Bulgaria
Be Informed, The Netherlands
University of Zurich, Switzerland

Project duration: March 2010 — May 2013

Summary

The MOLTO project aims to provide technology which can simultaneously tackle issues arising from real-time machine translation of web documents: localization to several languages, maintenance of their consistency in spite of asynchronous collaborative authoring with frequent edits, and grammatically and stylistically flawless text. Fifteen languages will be covered in the translations, including 12 of the 23 official languages of the European Union: Bulgarian, Danish, Dutch, English, Finnish, French, German, Italian, Polish, Romanian, Spanish, and Swedish. The 3 non-EU languages are Catalan, Norwegian, and Russian.

Two problems have slowed down the adoption of high-quality restricted language translations: development cost for a new domain or language, and learning curve for authoring texts in a restricted language. MOLTO tools will decrease the effort of developing restricted language translators radically by using the Grammatical Framework (GF) libraries. MOLTO editing tools are now available in initial prototypes for grammarians: the web-based editor and the GF plugin for the Eclipse integrated development environment. New members of the MOLTO Enlarged EU Consortium are extending the tools to a semantic wiki platform and testing the user-friendliness of these solutions for non-expert grammar writers.

MOLTO is exploring the two-way interoperability of grammars with Semantic Web conceptual models and hybrid models of combining rule-based translation systems with statistical machine translation. Data sets made available in a machine readable form, like RDF or OWL, can be used to construct a knowledge infrastructure suited to meaningful query and retrieval using natural languages. This approach is already being demonstrated with an application to the domain of cultural heritage. Combination approaches studied in MOLTO aim to integrate grammar-based and SMT models in a hybrid, robust MT system. Variants under consideration include e.g. soft integration in which phrase pairs or tree fragment pairs, generated by GF, are integrated as a discriminative probability models in a phrase-based SMT system. The testbed application for this research activity is information retrieval from patents in the pharmaceutical domain.

AIDA: Automatic Identification and Glossing of Dialectal Arabic

Heba Elfardy and Mona Diab
Center for Computational Learning Systems
Columbia University
 {heba,mdiab}@ccls.columbia.edu
<http://nlp.ldeo.columbia.edu/aida/>

Description

AIDA is a system for dialect identification, classification and glossing on the token and sentence level for written Arabic. Automatic dialect identification in Arabic is quite challenging because of the diglossic nature of the language and informality associated with the typical genres where dialectal Arabic (DA) is used. Moreover, DA lacks a standard orthography. Additionally the abundance of faux amis between the different varieties of Arabic, namely between Modern Standard Arabic (MSA) and DA, exacerbates the challenge of identifying dialectal variants. Hence identifying whether a (sequence of) token(s) is MSA or DA and providing an MSA-Gloss for the dialectal tokens in an utterance can aid Arabic MT in handling such informal genres more accurately.

AIDA aggregates several components including dictionaries and language models in order to perform named entity recognition, dialect identification & classification and MSA & English linearized glossing of the input text. The default output produces the following information for each token in the input text:

1. CLASS: this field displays whether a given word is DA, MSA or unknown (MSA, DA or UNK), and for dialectal words it identifies the class: either Egyptian, or Other (another Arabic dialect);
2. NE: Whether the word is a named-entity (NE) or not and if it is a NE, the NE class it belongs to (Person, Organization, GeoPolitical entity, Location);
3. MSA-Gloss: For dialectal tokens this field displays the MSA equivalents of a token ordered by their frequency of occurrence in Arabic Gigaword;¹
4. English-Gloss: The English equivalents of the given token.

Ex:

Input (UTF8)	ده	اللي	بيحصل	في	مصر	في	الوقت	الراهن
Input(BW)²	dh	El~y	byHSl	fy	mSr	fy	Alwqt	AlrAhn
Class	DA	DA	DA	DA/MSA	-	DA/MSA	MSA	MSA
NE:	-	-	-	-	GPE	-	-	-
MSA-Gloss:	*lk	Al*y	yHdv	fy	mSr	fy	Alwqt	AlrAhn
ENG-Gloss:	that	what	happen	in	Egypt	in	the time	the current

The output is configurable allowing the user to choose the output encoding as well as the user's preference on what tagging information to display. AIDA is accessible through a configurable web-based interface as well as a packaged pipeline that is available for offline processing.

¹ <http://www ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2009T30>

² We use Buckwalter transliteration scheme: <http://www.qamus.org/transliteration.htm>



Central and Southeast European Resources (CESAR)

European Commission
The Information and Communication Technologies Policy Support Programme
ICT-PSP PB Pilot Type B
Project ID number: 271022
<http://www.cesar-project.net>

List of partners
Nyelvtudományi Intézet, Magyar Tudományos Akadémia (HASRIL), Budapest, Hungary
Budapesti Műszaki és Gazdaságtudományi Egyetem (BME), Budapest, Hungary
University of Zagreb, Faculty of Humanities and Social Sciences (FFZG), Zagreb, Croatia
Instytut Podstaw Informatyki Polskiej Akademii Nauk (IPIPAN), Warsaw, Poland
Uniwersytet Łódzki (ULODZ), Łódź, Poland
Faculty of Mathematics, University of Belgrade (UBG), Belgrade, Serbia
Institut Mihajlo Pupin (IPUP), Belgrade, Serbia
Institute for Bulgarian language Prof Lyubomir Andreychin (IBL), Sofia, Bulgaria
Jazykovedný ústav Ľudovíta Štúra Slovenskej akadémie vied (LSIL), Bratislava, Slovakia

Project duration: February 2011 — January 2013



Summary

The main objective of the project – an integral part of META-NET – is to make available a comprehensive set of language resources and tools covering Bulgarian, Croatian, Hungarian, Polish, Serbian and Slovak. Building on a wide range of already existing resources and national or international activities, the project creates, populates and operates a comprehensive language-resource platform enabling and supporting large-scale multi- and cross-lingual products and services. The resources already involved in the project include interoperable mono- and multilingual speech databases, mono- and bilingual corpora, dictionaries, wordnets and relevant language technology processing tools such as tokenisers, lemmatisers, taggers and parsers. The main pillars of CESAR activity are seen as enhancement of resources and tools, adaptation of resources and tools to become compliant with the agreed standards for interoperability, upgrade of resources and tools by combining them with other resources and tools in order to achieve the foreseen level of interoperability and in adapting user-interfaces. A special effort is taken to achieve a common standard of involved resources and tools in order to enhance and facilitate the foreseen interoperability between them, as well as to evaluate their license schemes and IPR issues. A special track of activity is turning the linguistic development environment NooJ (www.nooj4nlp.net) into an open-source package running on most popular platforms. Key resources covered by the CESAR project will be linked and made interoperable using the facilities of the META-SHARE repository, which eventually will become an important component of a language technology marketplace for HLT researchers and developers, policy makers, language professionals (translators, interpreters, content and software localisation experts, etc.), as well as for industrial players, especially SMEs, catering for the full development cycle of HLT, from research through to innovative products and services.



Bologna Translation Service (BOLOGNA)

Funding agency: European Commission
Funding call identification: ICT-PSP 4th Call
Type of project: Theme 6: Multilingual Web
Project ID number: 270915
<http://www.bologna-translation.eu>

List of partners	
	CrossLang, Belgium (coordinator)
	Convertus, Sweden
	Applied Language Solutions, UK
	Koç University, Turkey
	Eleka Ingeniaritza Lingusitikoa, Spain

Project duration: March 2011 — February 2013

Summary

BOLOGNA (the “Bologna Translation Service” (BTS)) is an ICT PSP EU-funded project which specialises in the automatic translation of study programmes from French, German, Spanish, Portuguese, Turkish, Finnish into English, and from English into Chinese. At the core of the BTS framework are several machine translation (MT) engines through which web-based translation services are offered. The fully integrated BTS architecture includes a translation system that couples rule-based and statistical MT with automatic and human post-editing and translation memory.

The BOLOGNA project has just had its first annual review, on March 30, in Luxembourg. A first prototype of the service is now available and the initial user group has been expanded to over 50 universities and higher education institutions from 13 countries. Baseline systems have been built for all language pairs in the project and advanced systems are on their way.

Poster Session 2 – Project Papers



ACCEPT: Automated Community Content Editing PorTal

European Union

FP7 ICT-2011-7

STREP

288769

<http://www.accept.unige.ch>

List of partners
Université de Genève (Coordinator), Switzerland
University of Edinburgh, United Kingdom
Acrolinx, Germany
Symantec, Ireland
Lexcelera, France

Project duration: January 2012 — December 2014

Summary

The use of machine translation (MT) is becoming more pervasive, and at the same time Web 2.0 paradigms are democratising content creation. However, right now these two trends are fairly incompatible since current MT engines cannot produce acceptable results for community content due to the extreme variability within the content. The ACCEPT project will address this issue by developing new technologies designed specifically to help MT work better in this environment. The research consists of three main strands: (i) new paradigms for “minimally intrusive” preediting content. (ii) the development of strategies for post-editing content which, rather than fully relying on trained translators, will also leverage the monolingual skills of volunteer domain experts. (iii) the use of the insights gained in the editing process and using innovative text analytics to improve the statistical MT engines themselves. The project brings together two of the world's leading research centres in applied MT (Universities of Edinburgh and Geneva), as well as the leading provider of content editing technologies (Acrolinx). In addition, there are two extremely experienced MT users in the project: the software company Symantec and the language services provider Lexcelera. Symantec and Lexcelera will also bring their community forum experience: Symantec through its user forums and Lexcelera through Traducteurs sans Frontières, a non-profit organisation supported by Lexcelera which provides pro bono humanitarian translations via a community of translators.



PANACEA (Platform for Automatic, Normalised Annotation and Cost-Effective Acquisition of Language Resources for Human Language Technologies)

Funding agency: European Commission

Funding call identification: FP7-ICT

Type of project: STREP

Project ID number: 248064

<http://www.panacea-lr.eu>

List of partners
Universitat Pompeu Fabra, Spain (coordinator)
CNR – ILC, Italy
ILSP – R.C. “Athena”, Greece
University of Cambridge, United Kingdom
Linguattec, Germany
Dublin City University, Ireland
ELDA, France

Project duration: January 2010 — December 2012

Summary

A strategic challenge for Europe in today's globalised economy is to overcome language barriers through technological means. In particular, machine translation (MT) systems are expected to have a significant impact on the management of multilingualism in Europe.

PANACEA addresses the most critical aspect for MT: the so-called language resource (LR) bottleneck. Although MT technologies may consist of language-independent engines, they depend on the availability of language-dependent knowledge, i.e., they require LRs.

The objective of PANACEA is to build a factory of LRs that automates the stages involved in the acquisition, production, updating and maintenance of LRs required by MT systems and by other applications based on language technologies, and simplifies eventual issues regarding intellectual property rights. This automation will cut down the cost, time and human effort significantly. These reductions of costs and time are the only way to guarantee the continuous supply of LRs that MT and other language technologies will be demanding in the multilingual Europe.

In its second year, PANACEA has developed the second version of the LR factory, which allows the final user to acquire LRs by designing workflows (acquisition pipelines) that connect web services provided by the different partners. Tests for handling massive data have been successfully carried out. Some of the web services can be used for the acquisition of monolingual and parallel text and preprocessing, sentential and subsentential alignment and automated production of parallel corpora in widely used formats such as TMX. In the following months PANACEA will make available web services for the production of bilingual dictionaries, transfer grammars and acquisition of lexical information for producing information-rich dictionaries.



ATLAS: Automatic Translation into Sign Languages

Funding agency: Piedmont Region, Italy

Funding call identification: Converging Technologies - CIPE 2007

Research Sector: Cognitive Science and ICT

Website: <http://www.atlas.polito.it>

List of partners	
Politecnico of Turin, DAUIN	RAI Radiotelevisione Italiana S.p.A.
Turin University, Dep. of Computer Science	BEPS Engineering
Turin University, Dep. of Psychology	Lumiq Studios S.r.l.
CSP - ICT Innovation	Microsoft Innovation Center
Virtual Reality and Multimedia Park	University of Illinois at Chicago
Cooperativa GCS Global Communication	Alto Sistemi s.r.l.
Fondazione Bruno Kessler	

Project duration: January 2009 – July 2012

Summary

ATLAS is a three-year project, which exploits the convergence between cognitive sciences and ICT to build innovative services and tools to provide deaf people the possibility to follow and understand TV programme, media information, and entertainment channels, through the automatic translation into a sign language. This technology is a significant step toward the inclusion of deaf people in the global community and it may be considered a natural evolution of a process started by computers, internet, and mobile phones. Although the developed tools are applicable, in principle, to any broadcasted material and any language, ATLAS focuses on the Italian Sign Language (LIS - Lingua Italiana dei Segni), and on the weather forecast bulletins broadcasted by the Italian television.

The main objective of ATLAS is the distribution through various devices of the LIS animation performed by a virtual interpreter expressing the content of an Italian sentence in LIS. This goal is achieved through (i) the translation of the Italian text into the ATLAS Extended Written LIS (AEWLIS), representing the LIS into a written format, by either a statistical or a rule-based machine translation system; (ii) the conversion of the AEWLIS expression into an Animation Language (AL), able to drive the virtual actor; (iii) the generation of an AL-based animation sequence; (iv) and the delivery of the animation sequence on various user terminals (including DVB, web, mobile phones, and physical media), remotely controlled by local visualization engines and properly synchronized with audio/video contents.

Other notable outcomes of ATLAS are (i) the formal definition of the AEWLIS and AL languages; (ii) an Italian-LIS-AEWLIS parallel corpus of selected texts; (iii) an assisted editor for the AEWLIS annotation of Italian texts; and (iv) the set of developed tools distributed as open-source to allow their no-profit use by Deaf Communities.



FAUST: Feedback Analysis for User adaptive Statistical Translation

Seventh Framework Programme
Theme FP7-ICT-2009-4, Objective 2.2: Language-based interaction.
STREP
Grant agreement no. 247762
<http://faust-fp7.eu>

List of partners
Department of Engineering and Computer Laboratory, University of Cambridge, UK
Center for Language and Speech Technologies and Applications (TALP), Universitat Politecnica de Catalunya, Spain
Institute of Formal and Applied Linguistics, Charles University, Czech Republic
Language Weaver Inc., USA Language Weaver SRL, Romania
Softissimo, Inc., France

Project duration: February 2010 — January 2013

Summary

The FAUST project is developing fluent MT systems that respond to user feedback. Our objectives are to: Enhance the high-volume Reverso.net translation website with an experimental infrastructure for the study of instantaneous user feedback; Deploy novel web-oriented, feedback collection mechanisms that reduce noise and increase the utility of the web contributions; Automatically acquire novel data collections to study translation as informed by user feedback; Develop mechanisms for instantaneously incorporating user feedback into the MT engines; Create novel automatic metrics of translation quality which reflect user feedback; Develop translation models based on user feedback data and develop approaches to integrate natural language generation directly into MT to improve translation fluency and reduce negative feedback.











FAUST has now developed interactive environments for gather feedback from users. Machine translation systems developed within the project are freely available for use at the website **<http://labs.reverso.net>** . We will present the web architecture we have developed to support this collaborative research project. We will discuss some design issues in the user-facing portions of the website. We will present initial analyses of the feedback being collected, in terms of its potentially usefulness in refining MT systems. We will also describe the data sets and tools which we have developed within the project and which we have made available for public use. More information, including tools and data, is available at the project website **<http://faust-fp7.eu>** .





Bridges Across the Language Divide — EU-BRIDGE

European Union
Seventh Framework Programme FP7-ICT-2011-7-Language technologies
Integrating Project
grant agreement n°287658
<http://www.eu-bridge.eu>

List of partners	
	Karlsruhe Institute of Technology – KIT, Germany (coordinator)
	Fondazione Bruno Kessler – FBK, Germany
	Polsko Japonska Wyzsza Szkola Technik Komputerowych – PJWSTK, Poland
	RWTH Aachen University, Germany
	The University of Edinburgh, United Kingdom
	Hong Kong University of Science and Technology – HKUST, Hong Kong
	Red Bee Media Limited, United Kingdom
	Mobile Technologies GmbH, Germany
	PerVoice spa, Italy
	Accipio Projects GmbH, Germany
	Alcatel-Lucent Bell Labs France, France

Project duration: 1st February 2012 — 31st January 2015

Summary

EU-BRIDGE aims at developing automatic transcription and translation services that will permit and facilitate the development of innovative products and applications that require the transcription of and translation between languages— European as well as non-European. The project will prove the usability of the services by implementing four such applications as use cases: i) Captioning Translation for TV broadcasts, ii) University Lectures Translation, iii) European Parliament Translations, iv) Mobile Devices Communication Translation.

Therefore, EU-BRIDGE puts together academics as well as engineering and business expertise in order to create competitive offers to existing (current) needs in translation, communication, content processing and publishing. Prospective users (beneficiaries) of the project are European companies operating in a multilingual, audio-visual market (in particular TV captioning and translation).

EU-BRIDGE strives to achieve high performing speech translation technology, that will make production processes more cost-effective. Our speech translation services will be available through a network-based service infrastructure, with an easy to handle API that allows for easy integration into new products. By being able to utilize the offered services, companies will gain a distinctive advantage in a globalized, multilingual world. The project will reinforce the cooperation, dialogue and partnership between research and industry and will provide better understanding of user requirements.



GF Eclipse Plugin: an IDE for grammar development in GF

John J. Camilleri, Krasimir Angelov
john.j.camilleri, krasimir@chalmers.se
University of Gothenburg, Sweden
<http://www.grammaticalframework.org/eclipse/>

Description

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. FP7-ICT-247914.

This work introduces an Integrated Development Environment (IDE) for developing grammars using the Grammatical Framework (GF). GF is a functional programming language for writing grammars targeting multiple parallel languages simultaneously. Typical application areas of the framework are machine translation in limited domains and multilingual natural language generation from some formal representation.

The GF IDE is built on top of the Eclipse Platform and aims to provide a modern set of development tools which replace the traditional text editor and console-based approach. In doing so, we hope to make the task of grammar-writing more efficient, reduce the barrier to entry to GF and encourage new users and uses of the framework.

Apart from the standard features made available by the Eclipse platform—including multiple repositionable editor panels, syntax highlighting, code folding, automatic formatting and version control system integration—the GF Eclipse plugin also provides:

1. Immediate notification of **syntax errors** and helpful **semantic warnings** before ever reaching compilation.
2. **Module outlining**, which provides a summary view of every top-level judgement in a GF module, annotated with type information, and allowing quick navigation within source files.
3. **Identifier resolution** for both local and inter-module cross-references, allowing for instant notification of linking errors and jumping to the definition of a referenced identifier.
4. **Inline contextual documentation** for linked identifiers, including type information and listing of other overloaded forms or alternatives.
5. **Wizards** and **code generation** tools for creating new template modules when writing application grammars.
6. **Launch configurations**, allowing developers to set up multiple GF compilation configurations and run-time scripts, and run them directly from the IDE with a single click.
7. **Treebank test management**, providing a graphical interface for testing the grammar's linearisation and parsing performance against a predefined gold standard.



CrossLang Moses SMT Production System

Joachim Van den Bogaert, CrossLang / joachim@crosslang.com

Kim Scholte, CrossLang / kim.scholte@crosslang.com

<http://www.crosslang.com>

Description

Overview

CrossLang has developed an industry-grade Machine Translation (MT) and Post-Editing (PE) pipeline for the translation of high volumes of data. At its core, the system consists of a multiplexer/router, which allows source documents to be translated with any MT system connected to it. This ensures vendor independence for both CrossLang and its clients.

To facilitate the deployment in typical translation environments, the system features a SOAP XML interface and a dedicated connector to the SDL WorldServer translation management system. This standard set-up allows for very complex workflows including Translation Memory (TM) leveraging and the combination of offline and online translation.

For use in a crowd-sourcing environment, a lightweight Post-Editing platform has been added. The rationale is to allow domain-specialists, rather than translators, to rapidly review MT output for highly technical documents. This makes the acquisition of expensive CAT (Computer Aided Translation) tools and training unnecessary and speeds up the time-to-market.

The pipeline was developed to provide a cost-effective and rapid way to implement continuous localization, as opposed to project-based translation which typically involves a translation agency and the related project management overhead. Additionally, the MT-neutral implementation reduces client-side development costs and allows for multi-system scenarios.

Features

Hardened Moses SMT (Koehn, et al., 2007): for clients with full customization needs, the hardened Moses SMT implementation is probably the most attractive feature. A service layer on top of Moses SMT provides redundancy, load balancing, asynchronous processing, failover support, industry-standard document format support, alignment-based tag-handling, improved normalization and hardened (de)tokenization and (lower/real)casing.

Separation of concerns: the hardened Moses SMT set-up allows the deployment of third-party translation and language models, while still providing the text engineering capabilities built on top of the translation workflow. Named Entity Recognition (NER) and terminology management services, for example, can be added without disrupting the Moses SMT models. From a technical point of view, tagging and annotation are considered to be engineering issues, which are not allowed to interfere with the linguistic issues, as addressed by the SMT models. From a commercial point of view, clients can have third parties focus on linguistic quality, while the CrossLang system can take care of immediate production use.

Scalability and extensibility: the CrossLang system can be modified for performance or quality by adding extra hardware or processing steps through a unified API. The multiplexing capability makes it a suitable platform for MT systems combination.



Embedding Machine Translation in ATLAS Content Management System

**EU CIP-ICT Policy Support Programme Funding call identification
CIP-ICT-PSP-2009-3 Theme 3 Multilingual Web
Project ID number : 250467
<http://www.atlasproject.eu>**

List of partners	
	Tetracom Interactive Solutions Ltd., Bulgaria (coordinator)
	Institute for Bulgarian Language at the Bulgarian Academy of Sciences, Bulgaria
	Institute of Technology and Development, Bulgaria
	University of Hamburg - Research Group "Computerphilology", Germany
	German Research Center for Artificial Intelligence, Germany
	Atlantis Consulting SA, Greece
	Institute of Computer Science of the Polish Academy of Sciences, Poland
	Alexandru Ioan Cuza University of Iasi, Romania

Project duration: March 2010 — February 2013

Summary

The project aims to adjust and integrate several existing software components, assembling a platform for multilingual web content management called ATLAS, and a visualization layer called i-Publisher, which adds to the platform a powerful web-based point-and-click tool for building, reusing and managing multilingual content-driven web sites. ATLAS makes use of state-of-the art text technology methods in order to extract, translate information and cluster documents according to a given hierarchy. With the current available technology it is not possible to provide a translation system which is domain- and language variation independent and works for a couple of heterogeneous language pairs. Thus our approach envisages a system of user guidance, so that the availability and the foreseen system-performance is transparent at any time. For the MT-Engine of the ATLAS –System we decided on a hybrid architecture combining EBMT and SMT at word-based level. For the SMT-component PoS and domain factored models are used, in order to ensure domain adaptability. The document categorization module assigns to each document one or more domains. For each domain the system administrator has the possibility to store information regarding the availability of a correspondent specific training corpus. If no specific trained model for the respective domain exists, the user is provided with a warning, telling that the translation may be inadequate with respect to the lexical coverage. The output of the summarization module is processed in such way that ellipses and anaphora are omitted, and lexical material is adapted to the training corpus. The information extraction module is providing information about metadata of the document including publication age. For documents previous to 1900 we will not provide translation, explaining the user that in absence of a training corpus the translation may be misleading. The domain- and dating restrictions can be changed at any time by the system administrator if an adequate training model is provided.

TTC: Terminology Extraction, Translation Tools and Comparable Corpora

**European Community
Seventh Framework Programme (FP7/2007-2013)
STREP
248005
<http://www.ttc-project.eu/>**

List of partners
Université de Nantes, France (coordinator)
Universität Stuttgart, Germany
University of Leeds, United Kingdom
Sogitec Industries, France
Syllabs SARL, France
Tilde SIA, Latvia
Eurinnov, France

Project duration: January 2010 — December 2012

Summary

In scientific domains, resources like parallel corpora and bilingual dictionaries are often not available. As a consequence, translators spend a lot of time to create and manage terminology lists. Similarly, the lack of parallel data makes it difficult to build statistical machine translation systems.

The project TTC aims at providing data for machine translation systems, computer-assisted translation tools, and terminology management tools by automatically generating bilingual terminologies from comparable corpora. The project covers several languages of the European Union (English, French, German, Latvian and Spanish), as well as Chinese and Russian. To this end, a tool chain for compiling document collections, for terminology extraction and for bilingual term alignment is being developed, which concludes with exporting terminology data into CAT tools and MT systems.

Domain-specific corpora of several languages are collected by using a focused crawler. They then undergo pre-processing (tokenizing and POS-tagging): the project relies on flat linguistic analysis as it is available for most languages. For each language covered by the project, monolingual term extraction is performed. A part of the term extraction step consists in the identification of term variants, which provide valuable information for terminologists and can also help to deal with data sparsity.

The extracted terms are then grouped into bilingual term equivalent pairs (term alignment) using different approaches (context vector based term alignment and lexical alignment strategies). The resulting bilingual term lists can then be fed into translation systems and CAT tools. The use of terminology in machine translation tasks is also regarded as a form of extrinsic evaluation of the output of the tool chain.



Confident MT: Estimating Translation Quality for Improved Statistical Machine Translation

Irish Research Council for Science, Engineering and Technology (IRCSET)
Enterprise Partnership Scheme

<http://nclt.computing.dcu.ie/mt/confidentmt.html>

List of partners	
	Dublin City University, Ireland (coordinator)
	Symantec, Ireland
	Irish Research Council for Science, Engineering and Technology, Ireland

Project duration: November 2011 — November 2014

Summary

The aim of the Confident MT project is to develop Confidence Estimation (CE) methods to measure the reliability of Machine Translation (MT) output in the context of User-Generated Content (UGC). For localization purposes, a software company such as Symantec needs to deliver helpful content to its customers in their native languages. However, MT evaluation via automatic metrics is only possible when a reference translation is available. In the more realistic setting where no such reference is available, reliable techniques for estimating the quality of translation system output are needed. The CE methods will be applied across a range of MT systems (such as Rule-Based, Example-Based, Phrase-Based SMT and Syntax-Enhanced SMT) and the results will be used to inform the optimal combination of MT systems.

As more and more customers move away from traditional call centres and corporate websites in favour of self-service via dedicated discussion forums, there is a growing need for machine translation of UGC. Because this kind of content is an unedited mix of writing styles containing spelling mistakes, abbreviations and non-standard punctuation, it poses a particular challenge for Natural Language Processing (NLP) tools that have been trained on well-formed text.

We consider the following steps for the Confident MT project:

- Represent source and MT output text with both system-dependent and independent features.
- Adapt NLP tools (part-of-speech taggers, syntactic parsers, etc.) to UGC.
- Use particular feature classes to learn various confidence scoring models.
- Produce a confidence score to estimate machine-translated text quality.
- Combine MT systems based on CE scores.
- Examine CE scores correlation with automated and human scores.

PET: a Tool for Post-editing and Assessing Machine Translation

Wilker Aziz* and Lucia Specia**

* University of Wolverhampton, UK - W.Aziz@wlv.ac.uk

** University of Sheffield, UK - L.Specia@sheffield.ac.uk

Download: <http://pers-www.wlv.ac.uk/~in1676/pet/>

Given the significant recent improvements in Machine Translation (MT) quality and the increasing demand for cheap and fast translations, the post-editing of automatic translations is becoming a popular practice in the translation industry to save time and costs. The post-editing of automatic translations can also help identify problems in such translations and this can be used as feedback for researchers and developers to improve MT systems. Finally, post-editing can be used as a way of evaluating translations from one or more MT systems in terms of the effort required to correct them.

PET, a stand-alone **Post-Editing Tool** has two main purposes: facilitate the post-editing or revision of translations from any MT system and collect segment-level information from this process, e.g.: translation quality scores and post-editing time. In addition, it can be used to collect information for translation from scratch. PET works on any platform running a Java Virtual Machine. The interface displays source and target language texts in two columns, with many interface artefacts customisable through a configuration file.



The segment to translate or edit can be a text of any length. Segments are seen in context, between already edited segments (green) and segments to edit (red). For the active segment (yellow) it is possible to display additional information, such as the original draft translation, alternative translations (from other MT systems), or a reference translation. In fact any external textual information can be displayed on a per-segment basis, such as definitions, paraphrases or alternative translations, time or space constraints, etc. This information must be provided to the tool via XML files.

Once a segment is completed, assessment windows can be displayed to collect *explicit* feedback, e.g. translation fluency scores. PET also provides built-in *implicit* assessment indicators, such as i) time spent translating or editing a segment; ii) time spent assessing a segment; iii) assessment tags from pre-defined sets; iv) keystrokes grouped by type of keys; v) the Human Translation Edit Rate (HTER) between the draft translation and its post-edited version; vi) a time-stamped history of edit operations (i.e. insertion, deletion and substitution). Many other indicators can be added via PET's API.



LetsMT!: Do-It-Yourself Machine Translation Factory on the Cloud

**European Commission Competitiveness and Innovation Framework Programme
Call CIP-ICT-PSP.2009.5.1: Machine translation for the multilingual web
Small or medium-scale focused research project (STREP)
Project reference: 250456
<http://www.letsmt.eu>**

List of partners
Tilde, Latvia (coordinator)
University of Edinburgh, Human Communication Research Centre, UK
University of Zagreb, Faculty of Humanities and Social Sciences, Croatia
University of Copenhagen, Centre for Language Technology, Denmark
Uppsala University, Department of Linguistics and Philology, Sweden
Zoorobotics, Netherlands

Project duration: March 2010 — August 2012

Summary

LetsMT! has created a cloud-based platform for generation and running of SMT systems based on public and user-provided training data. Users can upload their parallel corpora to online repository and generate user-tailored SMT systems based on user selected data. LetsMT! hides complexity of MT generation by providing a cloud-based infrastructure and easy user interface to manage data, create and run multiple customized MT engines and use them for various translation needs. LetsMT! includes such features as storing of public and private training data, automated training of SMT systems from specified data, facilities for automated MT evaluation, facilities for running MT systems and web-based translation, API for integration of MT services, user and platform management facilities.

Publicly available parallel resources, such as OPUS, DGT and JRC-Acquis, supplemented by user-provided data, are used in LetsMT! as training data for development of SMT systems. Users can upload their data in a variety of formats (e.g. TMX, XLIFF and Moses formats, parallel documents in PDF, text and DOC formats, compressed gzip, zip and tar archives) that are automatically processed by validation and conversion tools. The system also includes a sentence alignment module for creation of new parallel resources for SMT training from scratch.

LetsMT! uses Moses as a language independent SMT solution and integrates it as a cloud-based service into the LetsMT! online platform. Moses toolkit has been adapted to fit into the rapid training, updating, and interactive access environment. The Moses SMT training pipeline involves a number of steps that each require a separate program to run. In the framework of LetsMT! this process is streamlined and made automatically configurable given a set of user-specified variables (training corpora, data for language model, dictionaries, tuning sets).

LetsMT! translation services can be used in several ways: through the web portal letsmt.com, through a widget for web-page translation, through browser plug-ins, and through integration in computer-assisted translation (CAT) tools SDL Trados and Kilgray memoQ. The platform provides API level access through a web service that (i) provides information about available SMT systems, their metadata and status, (ii) performs translation of text, (iii) allows SMT systems to be managed (load or unload), (iv) authenticates users and controls user access rights.

Oral Session 3 – Research Papers

Cross-lingual Sentence Compression for Subtitles

Wilker Aziz and Sheila C. M. de Sousa

University of Wolverhampton
Stafford Street, WV1 1SB
Wolverhampton, UK
W.Aziz@wlv.ac.uk
sheilacastilhoms@gmail.com

Lucia Specia

Department of Computer Science
University of Sheffield
211 Portobello, S1 4DP
Sheffield, UK
L.Specia@sheffield.ac.uk

Abstract

We present an approach for translating subtitles where standard time and space constraints are modeled as part of the generation of translations in a phrase-based statistical machine translation system (PB-SMT). We propose and experiment with two promising strategies for jointly translating and compressing subtitles from English into Portuguese. The quality of the automatic translations is measured via the human post-editing of such translations so that they become adequate, fluent and compliant with time and space constraints. Experiments show that carefully selecting the data to tune the model parameters in the PB-SMT system already improves over an unconstrained baseline and that adding specific model components to guide the translation process can further improve the final translations under certain conditions.

1 Introduction

The increasing demand for fast and cheap generation of audiovisual content is pushing research and development in the automatic translation of subtitles. Several attempts have been made in recent years to translate subtitles automatically by using different Machine Translation (MT) approaches (see Section 2). Overall, it has been shown that translation tools can be very helpful in producing adequate and fluent translations of subtitles, yielding significant time (and cost) reductions when compared to manually translating subtitles. However, subtitling has other important constraints in addition to translation quality: translations must fit

the space available on the screen and time slot so that they can be read by viewers. None of the existing approaches to translating subtitles considers these constraints.

When generating or translating subtitles from audio transcripts, human subtitlers should follow several conventions. Especially due to the advent of the DVD and the increasing use of smaller and smaller screens, norms and conventions in subtitling evolve quickly (Cintas and Remael, 2007). Currently, a norm of 40 characters per line, with two lines per screen, seems to be the most widely accepted for television screen, with common variants reaching up to 50 characters per line. Regarding time, a subtitle should remain in the screen for at least 1 second and at most 6 seconds if it contains two full lines.

It is important to make a distinction between translating directly from an audio transcript and translating from a subtitle in the source language. An audio transcript is likely to breach the time/space constraints simply because of the differences between human listening and reading rates. Therefore, some compression is usually necessary when generating monolingual subtitles. Producing subtitles in a second language however may require a second level of compression: even if the source language subtitle observes the time/space constraints, depending on the language-pair, a translation can be considerably longer than the source subtitle. This is particularly the case for translation between languages with significant structural differences such as English and the Romance languages. Additionally, lower quality source subtitles may already violate the time/space constraints.

We propose an approach for joint translation and compression that can be applied to translating from both transcripts and source language subtitles. We

experiment with the translation of English subtitles from a few popular TV series, taken from the OpenSubtitle section of the Opus corpus,¹ which contains both transcripts and translations by amateur subtitlers. As we discuss in Section 3.1, this corpus is particularly appealing for compression, since even manually produced translations violate the time/space constraints: 33.5% of the translations are longer than the recommended standard, with an average of 10 ± 7 additional characters.

Since compression may incur some loss of information, it should only be performed when necessary. The proposed approach *dynamically* defines the need for compression for every source subtitle and uses this information to bias the system to produce translations with the appropriate length. In order to do so, it exploits two main strategies for joint translation and compression in Statistical MT (SMT): the tuning of the SMT model parameters using a carefully selected dataset where space/time constraints are observed and the addition of explicit model components to guide the compression of the source subtitles via the selection of translation options that globally optimize the length of the target subtitle.

Our approach brings the following main contributions to previous work: (i) it takes advantage of the paraphrases that naturally occur in SMT systems, as opposed to resorting to artificially generated and potentially noisy paraphrases, or to the deep language processing techniques required by other sentence compression approaches; (ii) it is cross-lingual and therefore aims at ensuring that the target subtitle is compressed as required, as opposed to compressing the source subtitle, which could later get de-compressed as a consequence of an automatic translation, or directly compressing the target subtitle, which would require a sentence compression method for each target language; (iii) it dynamically identifies the need for compression as a function of the time/space available for the source subtitle, avoiding unnecessary compression, which could lead to inadequate translations; (iv) it yields a more efficient method for correcting both translation and compression in a single step. Additionally, it allows a more objective way of evaluating compression and translation based on these corrections, as opposed to commonly used subjective evaluation metrics based on human judgments for adequacy and fluency.

¹<http://opus.lingfil.uu.se/>

2 Related work

Several attempts have been made to translate subtitles automatically using Rule-Based (RBMT), Example-Based (EBMT), Statistical (SMT) and also Translation Memory (TM) tools. The first attempt by Popowich et al. (2000) use a number of preprocessing steps in order to improve the accuracy of an RBMT system and report 70% accuracy in a manual evaluation. In (Armstrong et al., 2006) an EBMT system is built using a corpus of subtitles. A comparison using a larger heterogeneous corpus including sentences from Europarl shows that a homogeneous setting leads to better translations. Volk (2008) uses an SMT system trained on a corpus of 5 million subtitle sentences and reports that SMT outputs can still be acceptable translations as long as they lie within 5 keystrokes from a reference translation. Sousa et al. (2011) presents an objective way of measuring translation quality for subtitles in terms of post-editing time. Experiments with a number of MT/TM approaches show that post-editing draft subtitles is consistently faster than translating them, and that post-editing time can be used to compare alternative TM/MT systems.

None of these approaches considers time/space constraints to generate or assess translations. On the other hand, a number of approaches have been proposed to compress subtitles. Most work is related to the ATrANoS² and MUSA³ projects. These projects focused on the monolingual compression of audio transcripts based on handcrafted deletion and substitution rules and statistics extracted from a parallel corpus of original transcripts and their compressed version (Daelemans et al., 2004; Vandeghinste and Pan, 2004). Piperidis et al. (2004) use TM and RBMT systems to translate the compressed subtitles. Glickman et al. (2006) contrast context-independent and context-dependent models to replace words in subtitles by shorter synonyms. Context models based on distributional similarity provided useful estimates, but they resulted in an accuracy of only 60%.

Previous work on general monolingual text compression can also be mentioned (Knight and Marcu, 2000; Cohn and Lapata, 2009). However, these works do not model time/space constraints explicitly and are rather aimed at compressing every input sentence. A closely related work on

²<http://atranos.esat.kuleuven.ac.be/>

³<http://sifnos.ilsp.gr/musa/>

monolingual compression is that by Ganitkevitch et al. (2011). The authors generate sentential paraphrases from phrasal paraphrases using the syntax-based SMT framework with two additional features to explicitly model compression. However, a fixed, pre-defined compression rate is used for all input sentences, as opposed to a dynamic rate that depends on the input segment and the need for compression given time/space conventions.

3 Cross-lingual sentence compression

3.1 Motivation

In what follows, we illustrate the need for compression in subtitles taking as example the English-Portuguese language pair and manually translated subtitles from 3 recent episodes of 6 popular TV series, amounting to 8,144 pairs of subtitles. Here a *subtitle* refers to the sequence of words appearing in one screen before an end-of-sentence marker.

For this analysis, we define the notion of *ideal length* as a function of the duration of the source language subtitle. More specifically, we consider the amount of time the source language subtitle is shown on the screen to define the ideal length of its translation. We follow the conventions in (Cintas and Remael, 2007) to identify the expected number of characters given a time slot and the frame rate. For example, if the source segment remains on the screen for 1 second, given the frame rate under consideration (25 frames per second), the number of characters in the translation (as well as in the source) subtitle should not exceed 17 characters.

By looking at the manually produced target side of this corpus, we found that 33.5% of the translations do not respect this ideal length, containing an average of 10 ± 7 additional characters. This may be a consequence of the fact that the source subtitles are sometimes too lengthy, since they were mostly generated by amateur subtitlers and are often merely transcriptions from the audio. In fact, 36.28% of the source subtitles are on average 8.85 ± 6.73 characters longer than expected. Nevertheless, 45.2% of the target subtitles are longer than the source subtitles by an average of 5 ± 4.5 characters, showing the natural difference in length between the two languages.

In order to show that standard MT tools will also fail to generate time/space compliant translations, we used Google Translate, a freely available translation tool, to translate the original subtitles. We found that 42.3% of the translations do not ob-

serve the ideal length, containing an average of 11.6 ± 8.7 additional characters. Interestingly, 63% of the translations are longer than the sources, with an average of 5.5 ± 4.3 additional characters. This seems to confirm the expected tendency: longer Portuguese translations are produced from English texts. It also shows that a general purpose MT system performs worse than the average amateur subtitler, producing even longer translations.

3.2 Rationale

We propose a joint approach to sentence translation and compression. The approach is based on a modification of the standard PB-SMT framework to include time/space constraints based on the input text. While in this paper we apply this approach to the translation of subtitles, it could be used for other applications that also require dynamically compressing translations.

In a nutshell, PB-SMT learns a bilingual dictionary of phrases (the *phrase table*) and their associated translation probabilities from a parallel corpus. It is not unusual that a given phrase in the source language is assigned a number of possible phrases in the target language, to accommodate for phenomena such as the ambiguity and paraphrasing in translation. During the translation process (*decoding*), the system chooses the translation that best fits the context based on a number of model components, among which are the phrase probability to indicate how common that translation is for the source phrase. Hence, a sizeable phrase table will contain many paraphrases, some of which will be shorter than others, particularly if this phrase table is generated from a corpus where the target language may require some compression. Different from previous work where monolingual paraphrases need to be externally generated, we focus on using these naturally occurring paraphrases in the phrase table. This approach has the advantages of providing a natural filter on the quality of the paraphrases as well as allowing the control of translation quality and compression rate in a single step. Additional paraphrases generated by any means could also be added to the phrase table, for example, following the method in (Ganitkevitch et al., 2011).

Compression may incur some loss of information. To prevent unnecessary and excessive compression, we treat compression as a less deterministic process by dynamically modeling the need for

compression as a function of the time/space constraints of each specific source segment. Our approach models time/space constraints by (i) adding model components to the Moses PB-SMT system (Koehn et al., 2007) to control the need of compression, and (ii) guiding the tuning process to prefer shorter translations. Each of these strategies is described in what follows.

3.3 Dynamic length penalty

Time and space constraints can be represented as a function of the time available for the source subtitle, as described in Section 3.1. In practice, these constraints will affect the length of the target subtitle, and therefore hereafter we refer to them as a *length constraint*. To incorporate this constraint into the Moses decoder, we define a character-based length penalty to adjust translations so that they meet this constraint as the difference between an *expected length* and the *length of the current translation hypothesis*. A length constraint is thus set individually for each segment to be translated.

As typical of PB-SMT, our length penalty component h_{lp} is incrementally computed in a per-phrase basis, that is:

$$h_{lp}(\bar{f}_1^K, \bar{e}_1^K, c) = \sum_{k=1}^K \hat{h}_{lp}(\bar{f}_k, \bar{e}_k, c)$$

where \bar{f}_1^K denotes a source sentence f broken into K contiguous phrases, \bar{e}_1^K denotes the K target phrases that make up the hypothesised translation e , and c is the expected length constraint.

The character length penalty models how much the translation hypothesis deviates from the expected length constraint c , that is: $h_{lp}(\bar{f}_1^K, \bar{e}_1^K, c) \equiv c - \text{length}(\bar{e}_1^K)$, where $\text{length}(x)$ is the number of characters of the sequence x including a space between every adjacent token. Every target segment spans a portion of text that is proportional to the source phrase being covered, therefore the length constraint can be adjusted to the segment level as \hat{h}_{lp} in:

$$\hat{h}_{lp}(\bar{f}, \bar{e}, c) = c \times \frac{\text{length}(\bar{f})}{\text{length}(f)} - \text{length}(\bar{e})$$

where \bar{f} is a source phrase, \bar{e} is its hypothesised translation, and $\frac{\text{length}(\bar{f})}{\text{length}(f)}$ is a scaling factor that allows computing h_{lp} in a per-phrase basis.

In order to define the *expected length* constraint, c , for a given subtitle translation, we consider the following sources of information (in characters):

```
<s id="15" lp::ideal="23" lp::input="19"
  lp::min="19">I never felt this .</s>
```

Figure 1: Example of constraints.

- $lp::ideal$ is the ideal length given the duration of the subtitle and the conventions in (Cintas and Remael, 2007), as outlined in Section 3.1;
- $lp::input$ is the length of the source subtitle;
- $lp::min$ is the minimum of the 2 above values.

We use the decoder’s XML mark-up scheme to assign the length constraints to the source subtitles as shown in Figure 1. Based on these types of information we build two variations of our approach:

LP₂) Two model components: We add the constraint $lp::ideal$ that represents a theoretically supported value based on the source subtitle duration. That is, with $lp::ideal$ the system is trained to produce translations that can be read given the time slot of the source subtitle. However, sometimes a subtitle is shown for a long time, although it contains a very short string, and therefore $lp::ideal$ can lead the decoder to produce translations that are longer than necessary simply because there is space left for it. To compensate for this issue, we add a second model component: $lp::input$, which may differ significantly from the former.

LP₁) One model component: An alternative approach adds a single model component, $lp::min$, which puts the two above mentioned components together. If the ideal length is longer, the model targets the input length. If instead the source subtitle is longer, the model targets the ideal length, aiming at producing a translation that observes the time and space constraints even though the original text is too lengthy.

3.4 Tuning process

Adding a new component to the model requires learning its contribution and its interaction with the other components. These model parameters are adjusted in a process often referred to as *tuning*. In this process a dataset for which gold translations are known is used to incrementally tune the model parameters towards improving a measure of quality, traditionally BLEU (Papineni et al., 2002).

In order to guide the model to select translation candidates that are likely to be good while complying with the length constraint, at tuning time,

when compression is necessary the model must reward phrases that are shorter. This can be done by i) biasing the evaluation metric towards shorter translations (Ganitkevitch et al., 2011); ii) using evaluation metrics that go beyond string matching, such as METEOR (Lavie and Agarwal, 2007), which also matches synonyms and paraphrases; iii) adding multiple reference translations that vary in length; or (iv) filtering the tuning set so that it contains only pairs of segments that comply with the length constraint. These strategies do not necessarily exclude each other, and can rather complement each other. An evaluation metric that rewards compression in general does not suit our application to subtitle translation, where segments should only be compressed when necessary. As for tuning with metrics like METEOR, the lack of quality in-domain Portuguese paraphrases for the subtitle domain is an issue.⁴ Since having multiple references is expensive, we opted for filtering the tuning set so that it contains only subtitle pairs that comply with the length constraint, i.e. subtitles whose target sides are equal or shorter than the source sides and equal or shorter what is expected given the duration of the sources (ideal length).

The tuning of the proposed systems is performed using these controlled datasets and the standard MERT procedure in Moses.

4 Experimental settings

4.1 Corpus

We use the most recent version of the parallel corpus of subtitles distributed as part of the Opus Project (Tiedemann, 2009). The parallel corpus is made up of freely available fan-made subtitles⁵ for a large variety of TV series, movies and other audiovisual materials. The English-Brazilian Portuguese portion of the corpus amounts to 28 million subtitle pairs. We selected the top 14 million pairs to build a translation model, which we judged to be enough for a PB-SMT system. The data is already automatically pre-processed: tokenized, truecased and word-aligned.

To generate the tuning and test sets we took the most recent episodes of three TV series from the same source of fan-made subtitles, which were not included in the Opus release: Dexter (D), How I

⁴Experiments with popular methods to generate paraphrases such as (Bannard and Callison-Burch, 2005) resulted in very poor paraphrases for this domain, most likely due to the highly non-literal nature of translations.

⁵<http://www.opensubtitles.org>

Met Your Mother (H) and Terra Nova (T). A tuning set and a test set was created for each of these series. These were pre-processed as the training data using the tools and methods provided by Opus.

After filtering the tuning sets according to the restrictions defined in Section 3.4, the resulting sets contained 1900 (D), 1130 (H) and 1320 (T) English subtitles and their single reference translations. For testing the models, a test set containing 400 source subtitles from 2 recent episodes of each series (200 per episode, in their original sequence) was compiled, amounting to 1200 subtitles. No filtering was applied to the test sets.

4.2 Models and baselines

We experiment with the two variations of the length constrained models (Section 3.3), LP₂ and LP₁. Additionally, we consider three baselines:

Baseline 1 (B₁) Google Translate, an off-the-shelf SMT system known to be often used by amateur subtitlers to generate translations.

Baseline 2 (B₂) A PB-SMT system built using Moses and the same corpus as our proposed models, but tuned on unconstrained tuning sets (2000 subtitles per series), i.e., without selecting only subtitles that are compliant with time/space constraints.

Baseline 3 (B₃) A PB-SMT system built using Moses trained on the same corpus as our proposed models, and tuned on the same tuning set (only space/time compliant subtitles), but without any length penalty.

In all cases, the tuning of the systems was performed individually for each TV series.

4.3 Evaluation

In order to objectively evaluate our approach for both translation and compression, we have human translators post-editing the machine translations and collect various information from this process. Meta-information from post-editing has been successfully used in previous work to avoid the subjective nature of explicit scoring schemes (Specia, 2011; Sousa et al., 2011).

We use a post-editing tool⁶ that gathers post-editing effort indicators on a per-subtitle basis, including keystrokes, time spent by translators to post-edit the subtitle and the actual post-edited

⁶<http://pers-www.wlv.ac.uk/~in1676/pet/>

subtitle (Aziz et al., 2012). The tool allows the specification of the length constraints and renders the tasks differently according to how well the translation observes time/space constraints. It uses colors to facilitate the visualization of the compression needs and indicates the number of characters that need to be compressed or remain to be used in the translation.

Each test set was given to human translators along with the post-editing tool and guidelines for translation correction and compression. Eight Brazilian Portuguese native speakers and fluent speakers of English with significant experience in English-Portuguese translation post-edited the MT outputs. We base our evaluation on the computation of automatic metrics such as HTER (Snover et al., 2006) between the machine translation and its post-edited version (Section 5).

4.3.1 Post-editing guidelines and task design

Guidelines and examples of translations were given to the translators and adapted after a pilot experiment with 150 subtitles post-edited per translators. In a nutshell, translators should minimally correct translations to make them fluent and adequate (style and consistency should be disregarded) and compress them when necessary. The following instructions summarise the guidelines:

- If the translation is fluent, adequate and follows the length constraint: do not post-edit it.
- If the translation observes the length constraint but is not fluent and/or is not adequate: perform the minimum necessary corrections to make it fluent and adequate, trying to keep it within the length limit as much as possible.
- If the translation is fluent and adequate but does not observe the length constraint: compress it towards the ideal length, preserving as much as possible the meaning of the source subtitle and keeping the translation fluent.

For the final evaluation, we split each test set in batches of 50 subtitles and distributed them among the eight translators in a way that the same annotator would never see the same source subtitle more than once and would post-edit target subtitles from randomly selected systems. Subtitles in a batch were shown in their original sequence so that the translators could rely on previous and posterior contexts for both compression and correction. Annotators post-edited 200 subtitles a day.

5 Results

In this section we discuss the performance of the systems in terms of automatic metrics computed using the human post-edited translations for the 3 test sets (i.e. D, H and T). Note that translation quality and compression are jointly evaluated. We use the *multeval* toolkit (Clark et al., 2011) to score the systems and test them for statistical significance.⁷ We report BLEU, TER and the hypothesis length over the reference length in percentage (LENGTH).⁸

To make the reference set we put together all post-edited translations that were length compliant. In addition, references longer than the ideal length were kept only if no compliant paraphrase was produced by any of the annotators (we observed only 5 of those cases).

For all test sets (Tables 1 to 3), systems trained using subtitles data outperform B₁ (Google) by a large margin, which shows that parallel subtitles provide phrase pairs that are naturally better/shorter than those typical of general purpose parallel data. Additionally, simply constraining the tuning set to space compliant subtitles (B₃) already yields significant improvement over B₂ (unconstrained tuning).

System	BLEU \uparrow	TER \downarrow	LENGTH
B ₃	61.7	30.3	116.0
B ₁	43.6 ⁻	63.6 ⁻	156.5 ⁻
B ₂	58.1 ⁻	35.7 ⁻	127.3 ⁻
LP ₂	62.2	29.5	115.5
LP ₁	64.6 [†]	28.3 [†]	115.8

Table 1: Metric scores for the dataset D: p-values are computed with respect to B₃.

Table 1 shows that LP₁ outperforms B₃ in terms of both BLEU and TER. It suggests that the length penalty contributes to producing subtitles that require less post-editing. On the other hand, Tables 2 and 3 show no statistically significant differences between B₃ and the systems with length penalties (except for LP₂ on test set H). Moreover, while Table 2 suggests that LP₁ produces translations slightly longer than necessary (LP₁'s LENGTH is larger than B₃'s), Table 3 shows that LP₂ compresses the translations slightly more than

⁷Hereafter [†], [‡] and ^{*} denote results that are significantly better than a baseline ($p < 0.01$, 0.05 and 0.10 , respectively). ⁻, ⁼ and [≡] denote results that are significantly worse than a baseline ($p < 0.01$, 0.05 and 0.10 , respectively).

⁸The closer a system is to 100%, the closer its outputs are in length to what human translators produce as final subtitles.

B_3 (LP_2 's LENGTH is smaller than B_3 's). These somewhat conflicting results suggest that characteristics of the dataset may affect the generalization power of the length penalty (see Table 4).

System	BLEU \uparrow	TER \downarrow	LENGTH
B_3	70.8	20.0	108.5
B_1	47.0 ⁻	52.8 ⁻	144.3 ⁻
B_2	60.6 ⁻	31.3 ⁻	126.9 ⁻
LP_2	70.3	21.0 ⁼	109.1
LP_1	70.6	20.7	110.0 ⁼

Table 2: Metric scores for the dataset H: p-values are computed with respect to B_3 .

System	BLEU \uparrow	TER \downarrow	LENGTH
B_3	60.0	33.8	120.2
B_1	41.0 ⁻	63.1 ⁻	152.1 ⁻
B_2	52.7 ⁻	44.1 ⁻	135.8 ⁻
LP_2	60.4	33.4	119.3 [‡]
LP_1	57.9 ⁼	34.8	119.8

Table 3: Metric scores for the dataset T: p-values are computed with respect to B_3 .

Table 4 shows the distribution of the input and ideal lengths in our test sets. While the average input length is almost constant across datasets, the other two constraints show that the datasets H and T require more compression than D.

Finally, although over 36% of the source subtitles in our datasets are not time/space compliant, Table 5 shows that our systems decrease this non-compliance in 10% by either filtering the tuning set (B_3) or modelling length penalties (LP_2 and LP_1). Moreover, even if the automatic compression is not enough, models LP_2 and LP_1 make manual compression easier, as the lower percentage of malformed PEs suggests.

5.1 Further improvements

The human post-editing produced 5 reference translations for a set of 1200 sentences (400 per series). We used these sentences altogether to experiment with an alternative tuning approach: a tuning set with explicit human-made, mostly length compliant, paraphrases (see Section 3.4). In Table 6 the

Set	lp::input	lp::ideal	lp::min
D	28.82 \pm 15.43	36.99 \pm 14.40	26.03 \pm 12.86
H	28.40 \pm 13.81	33.25 \pm 13.77	25.97 \pm 12.20
T	28.34 \pm 15.22	30.14 \pm 11.47	24.61 \pm 11.93

Table 4: Average length constraints (in number of characters) in source subtitles.

Malformed	B_1	B_2	B_3	LP_2	LP_1
MT	44.15	34.41	25.40	24.57	25.65
PE	8.50	9.08	7.0	5.65	5.65

Table 5: Percentage of MT and human post-edited translations that are longer than the ideal length.

superscript m denotes a system that was retrained using this multiple-reference tuning set. We kept B_3 in the comparison to measure whether the new tuning set brings up any significant performance gain.

System	BLEU \uparrow	TER \downarrow	LENGTH
B_3^m	63.2	26.8	103.8
B_3	62.1 ⁼	27.0	106.1 ⁻
LP_2^m	63.8	26.0 [‡]	103.3 [*]
LP_1^m	64.1 [*]	25.9 [†]	103.6

Table 6: Metric scores for a dataset of 600 unseen sentences (200 from each series) post-edited by 4 translators following the guidelines presented in Section 4.3.1: p-values are computed with respect to B_3^m .

Adding multiple references in the tuning phase yields consistent and significant gains in performance. The new systems significantly outperform B_3 in terms of both BLEU and TER. Furthermore, B_3 is the system which is the farthest from the 100% LENGTH, that is, the improved systems produce translations that are closer in length to what human translators produce as final subtitles, with LP_2^m having the closest length. Finally, LP_1^m and LP_2^m are both significantly better than B_3^m in terms of TER.

6 Conclusions

We have presented an approach to successfully compress subtitles in a multilingual scenario by i) adequately choosing tuning data and ii) giving a PB-SMT model the capability of controlling the length of its hypotheses. Moreover, we have shown that in the presence of reliable, often shorter, paraphrases in the tuning set, more promising length-constrained models can be produced.

In future work we plan to further evaluate the model by trying to isolate edits due to translation quality from edits due to compression needs. Besides we must consider other indicators of post-editing effort such as post-editing time and keystrokes.

References

- Armstrong, Stephen, Colm Caffrey, Marian Flanagan, Dorothy Kenny, Minako O'Hagan, and Andy Way. 2006. Leading by Example: Automatic Translation of Subtitles via EBMT. *Perspectives*, 3(14):163–184.
- Aziz, Wilker, Sheila Castilho Monteiro de Sousa, and Lucia Specia. 2012. PET: a tool for post-editing and assessing machine translation. In *The Eighth International Conference on Language Resources and Evaluation*, LREC '12, Istanbul, Turkey, May. To appear.
- Bannard, Colin and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 597–604, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Cintas, Jorge Díaz and Aline Remael. 2007. *Audio-visual Translation: Subtitling. Translation Practice Explained*. St Jerome Publishing.
- Clark, Jonathan, Chris Dyer, Alon Lavie, and Noah Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the Association for Computational Linguistics*.
- Cohn, Trevor and Mirella Lapata. 2009. Sentence compression as tree transduction. *Journal of Artificial Intelligence Research*, 34:637–674.
- Daelemans, Walter, Anja Höthker, and Erik Tjong Kim Sang. 2004. Automatic sentence simplification for subtitling in dutch and english. In *4th International Conference on Language Resources and Evaluation*, pages 1045–1048, Lisbon, Portugal.
- Ganitkevitch, Juri, Chris Callison-Burch, Courtney Napoles, and Benjamin Van Durme. 2011. Learning sentential paraphrases from bilingual parallel corpora for text-to-text generation. In *Conference on Empirical Methods in Natural Language Processing*, pages 1168–1179, Edinburgh, Scotland, UK., July.
- Glickman, Oren, Ido Dagan, Mikaela Keller, Samy Bengio, and Walter Daelemans. 2006. Investigating lexical substitution scoring for subtitle generation. In *10th Conference on Computational Natural Language Learning*, pages 45–52, New York City, New York.
- Knight, Kevin and Daniel Marcu. 2000. Statistics-based summarization - step one: Sentence compression. In *17th National Conference of the American Association for Artificial Intelligence*, pages 703–710, Austin, USA.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics: Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.
- Lavie, A. and A. Agarwal. 2007. METEOR: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *2nd Workshop on Statistical Machine Translation*, pages 228–231, Prague.
- Papineni, K., S. Roukos, T. Ward, and W. Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Morrystown.
- Piperidis, Stelios, Iason Demiros, Prokopis Prokopidis, P. Vanroose, A. Hoethker, Walter Daelemans, E. Sklavounou, M. Konstantinou, and Y. Karavidas. 2004. Multimodal multilingual resources in the subtitling process. In *4th International Conference on Language Resources and Evaluation*, pages 205–208, Lisbon, Portugal.
- Popowich, Fred, Paul Mcfetridge, Davide Turcato, and Janine Toole. 2000. Machine translation of closed captions. *Machine Translation*, 15:311–341.
- Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *7th Conference of the Association for Machine Translation in the Americas*, pages 223–231.
- Sousa, Sheila C. M., Wilker Aziz, and Lucia Specia. 2011. Assessing the post-editing effort for automatic and semi-automatic translations of DVD subtitles. In *Recent Advances in Natural Language Processing Conference*, Hissar, Bulgaria.
- Specia, Lucia. 2011. Exploiting objective annotations for measuring translation post-editing effort. In *15th Conference of the European Association for Machine Translation*, pages 73–80, Leuven.
- Tiedemann, Jörg. 2009. News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In Nicolov, N., K. Bontcheva, G. Angelova, and R. Mitkov, editors, *Recent Advances in Natural Language Processing (vol V)*, pages 237–248. John Benjamins.
- Vandeghinste, Vincent and Yi Pan. 2004. Sentence compression for automated subtitling: A hybrid approach. In *ACL-04 Workshop Text Summarization Branches Out*, pages 89–95, Barcelona, Spain.
- Volk, Martin. 2008. The automatic translation of film subtitles. a machine translation success story? In *Resourceful Language Technology: Festschrift in Honor of Anna*, volume 7, Uppsala, Sweden.

Can Automatic Post-Editing Make MT More Meaningful?

Kristen Parton¹ Nizar Habash¹ Kathleen McKeown¹ Gonzalo Iglesias² Adrià de Gispert²

¹Columbia University, NY, USA

{kristen, kathy, habash}@cs.columbia.edu

²University of Cambridge, Cambridge, UK

{gi212, ad465}@eng.cam.ac.uk

Abstract

Automatic post-editors (APEs) enable the re-use of black box machine translation (MT) systems for a variety of tasks where different aspects of translation are important. In this paper, we describe APEs that target adequacy errors, a critical problem for tasks such as cross-lingual question-answering, and compare different approaches for post-editing: a rule-based system and a feedback approach that uses a computer in the loop to suggest improvements to the MT system. We test the APEs on two different MT systems and across two different genres. Human evaluation shows that the APEs significantly improve adequacy, regardless of approach, MT system or genre: 30-56% of the post-edited sentences have improved adequacy compared to the original MT.

1 Introduction

Automatic post-editors (APEs) seek to perform the same task as human post-editors: correcting errors in text produced by machine translation (MT) systems. APEs have been used to target a variety of different types of MT errors, from determiner selection (Knight and Chander, 1994) to grammatical agreement (Mareček et al., 2011). There are two main reasons that APEs can improve over decoder output: they can exploit information unavailable to the decoder, and they can carry out deeper text analysis that is too expensive to do in a decoder.

We describe APEs that target three types of adequacy errors: deleted content words, content words that were translated into function words, and mistranslated named entities. These types of errors are common across statistical MT (SMT) systems and can significantly degrade translation *adequacy*, the amount of information preserved during translation. Adequacy is critical to the success of many cross-lingual applications, particularly cross-lingual question answering (CLQA),

where adequacy errors can significantly decrease task performance. The APEs utilize word alignments, source- and target-language part-of-speech (POS) tags, and named entities to detect phrase-level errors, and draw on several external resources to find a list of corrections for each error.

Once the APEs have a list of errors with possible corrections, we experiment with different approaches to apply the corrections: an approach that uses phrase-level editing rules, and two techniques for passing the corrections as feedback back to the MT systems. The *rule-based APE* uses word alignments to decide where to insert the top-ranked correction for each error into the target sentence. This approach rewrites the word or phrase where the error was detected, but does not modify the rest of the sentence. We test these MT system-independent rules on two MT systems, MT A and MT B (described in more detail in section ??).

The *feedback APE* passes multiple suggestions for each correction back to the MT system, and allows the MT decoder to determine whether to correct each error and how to correct each error during re-translation. Many MT systems have a mechanism for “pre-editing,” or providing certain translations in advance (e.g., for named entities and numbers). We exploit this mechanism to provide post-editor feedback to the MT systems during a second-pass translation. While post-editing via feedback is a general technique, the mechanism the decoder uses is dependent upon the implementation of each MT system: in our experiments, MT A accepts corpus-level feedback from the APE, while MT B can handle more targeted, phrase-level feedback from the APE.

Our evaluation using human judgments shows that the APEs always improve the overall translation adequacy: across all conditions, whether rule-based or feedback, MT A or MT B, newswire or web genre, adequacy improved in 30-56% of post-edited sentences, and the improved sentences significantly outnumbered sentences that got worse. We also collected judgments on fluency, which highlighted the relative advantages of each APE

approach. The rule-based approach affords more control for error correction, at the expense of fluency. The feedback approach improves adequacy only when it can maintain some level of fluency, which results in more fluent post-edits than the rule-based approach. Due to the fluency constraints, the feedback APEs do not modify as many sentences as the rule-based APE, and therefore improve fewer sentences. Our analysis suggests ways in which feedback may be improved in the future.

2 Motivation

As MT has increased in quality and speed, its usage has gone beyond open-ended translation towards a variety of applications: cross-lingual subjectivity analysis, cross-lingual textual entailment, cross-lingual question-answering, and many others. Open-ended MT systems are task-agnostic, so they seek to balance fluency and adequacy. Depending on the task, however, adequacy may take precedence over fluency (or vice versa). We propose using the framework of automatic post-editing (Knight and Chander, 1994) to detect and correct task-specific MT errors at translation time. (In this paper, we use the term “post-editing” to refer to automatic post-editing only.)

The advantage of post-editing is that the APE can adapt any MT output to the needs of each task without having to re-train or re-tune a specific MT system (Isabelle et al., 2007). Acquiring parallel text, training and maintaining an SMT system is time-consuming and resource-intensive, and therefore not feasible for everyone who wishes to use MT in an application. Ideally, an APE can adapt the output of a black-box MT system to the needs of a specific task in a light-weight and portable manner. Since APEs are not tied to a specific MT system, they also allow application developers flexibility in switching MT systems as better systems become available.

Our focus on adequacy in automatic post-editing is motivated by CLQA with result translation. In this task, even when the correct answer in the source language is retrieved, it may be perceived as irrelevant in the target language if not translated correctly. The MT errors that have the biggest impact on CLQA include missing or mistranslated named entities and missing content words (Parton and McKeown, 2010; Boschee et al., 2010).

Manual error analysis of MT has shown that missing content words produce adequacy errors across different language pairs and different types of SMT systems. Condon et al. (2010) found that 26% of their Arabic-English MT errors were verb,

noun or pronoun deletions. Similarly, Vilar et al. (2006) found that 22% of Chinese-English MT errors were content deletion. Popović and Ney (2007) reported that 68% deleted tokens from their Spanish-English MT system were content words. We address these errors via automatic post-editing, with the ultimate goal of improving MT output for adequacy-oriented tasks.

3 Related Work

The goal of APE is to automatically correct translated sentences produced by MT. Adaptive APEs try to learn how to improve the translation output by adapting to the mistakes made by a specific MT system. In contrast, general APEs target specific types of errors, such as English determiner selection (Knight and Chander, 1994), certain types of grammar errors in English (Doyon et al., 2008) and Swedish (Stymne and Ahrenberg, 2010), and complex grammatical agreement in Czech (Mareček et al., 2011). The APEs in this paper are more similar to general APEs, since they target specific kinds of adequacy errors.

APEs may utilize information unavailable to the decoder to improve translation output. Previous task-based MT approaches have used task context to select verb translations in CLQA at query time (Ma and McKeown, 2009) and to identify and correct name translations in CLIR (Parton et al., 2008). The rule-based APE we describe extends those APEs to cover additional types of adequacy errors. The feedback APEs are most similar to (Suzuki, 2011), which uses confidence estimation to select poorly translated sentences and then passes them to an adaptive SMT post-editor. Other work in confidence estimation (Specia et al., 2011) aims to predict translation adequacy at runtime without using reference translations, which is similar to our error detection step.

Many APEs use sentence-level analysis tools to make improvements over decoder output. Since these tools rely on having a fully resolved translation hypothesis (and since they are expensive), they are infeasible to run during decoding. The DepFix post-editor (Mareček et al., 2011) parses translated sentences, and uses the bilingual parses to correct Czech morphology. While syntax-based MT systems use POS and parses, most systems do not use other types of annotations (e.g., information extraction, event detection or sentiment analysis). An alternative approach would be to incorporate these features directly into the MT system; the focus of this paper is on adapting translations to the task without changing the MT system.

4 Post-Editing Techniques

Our APEs carry out three steps: 1) detect errors, 2) suggest and rank corrections for the errors, and 3) apply the suggestions. All the APEs use identical algorithms for steps 1 and 2, and only differ in how they apply the suggestions. The algorithms are language-pair independent, though we carried out all of our experiments on Arabic-English MT.

4.1 Pre-Processing

The Arabic source text was analyzed and tokenized using MADA+TOKAN (Habash et al., 2009). Each MT system used a different tokenization scheme, so the source sentences were processed in two separate pipelines. Separate named entity recognizers (NER) were built for each pipeline using the Stanford NER toolkit (Finkel et al., 2005), by training on CoNLL and ACE data. Each translated English sentence was re-cased using Moses and then analyzed using the Stanford CoreNLP pipeline to get part-of-speech (POS) tags (Toutanova et al., 2003) and NER (Finkel et al., 2005).

4.2 Detecting Errors and Suggesting Corrections

The APEs address specific adequacy errors that we have found to be most detrimental for the CLQA task: content words that are not translated at all, content words that are translated to function words, and mistranslated named entities. In the error detection step, these types of errors are detected via an algorithm from prior work that uses bilingual POS tags and word alignments (Parton and McKeown, 2010). Each flagged error consists of one or more source-language tokens and zero or more target-language tokens. In the error correction step, the source and target sentences and all the flagged errors are passed to the suggestion generator, which uses the following three resources.

Phrase Table: The phrase table from MT B is used as a phrase dictionary (described in more detail in ??).

Dictionaries: We also use a translation dictionary extracted from Wikipedia, a bilingual name dictionary extracted from the Buckwalter analyzer (Buckwalter, 2004) and an English synonym dictionary from the CIA World Factbook.¹ They are high precision and low recall: most errors do not have matches in the dictionaries, but when they do, they are often correct, particularly for NEs.

¹<http://www.cia.gov/library/publications/the-world-factbook>

Background MT corpus: Since our motivation is CLQA, we also draw on a resource specific to CLQA: a background corpus of about 120,000 Arabic newswire and web documents that have been translated into English by a state-of-the-art industry MT system. Ma and McKeown (2009) were able to exploit a similar pseudo-parallel corpus to correct deleted verbs, since words deleted in one sentence are frequently correctly translated in other sentences.

For each error, the source-language phrase is converted into a query to search all three resources. Then the target-language results are aggregated and ranked by overall confidence scores. The confidence scores are a weighted combination of phrase translation probability, number of dictionary matches and term frequencies in the background corpus. The weights were set manually on a development corpus.

4.3 Rule-Based APE

Table 1 shows examples of sentences post-edited by the different APEs. For each error, the rule-based post-editor applies the top-ranked correction using one of two operations: *replace* or *insert*. An error can be replaced if there is an existing translation, and all of the source- and target-language tokens aligned to the error are flagged as errors. (This is to avoid over-writing a correct partial phrase translation, as in example 2a where the word “their” is not replaced.) If the error cannot be replaced, the new correction is inserted.

During *replace*, all the original target tokens are deleted, and the correction is inserted at the index of the first target token. For *insert*, the algorithm first chooses an insertion index, and then inserts the correction. The insertion index is chosen based on the indices of the target tokens in the error. If there are no target tokens, the insertion index is determined by the alignments of the neighboring source tokens. If they are aligned to neighboring translations, the correction is inserted between them. Or, if only one of them is aligned to a translation, the correction is inserted adjacent to it. If an insertion index cannot be determined via rules, the error is not corrected.

These editing rules are MT system-independent, language-independent and relatively simple. The word order is copied from the original translation or from the source sentence. This simple model worked for (Parton et al., 2008) because they were rewriting mistranslated NEs that were already present in the translation. Similarly, Ma and McKeown (2009) successfully re-inserted deleted verbs into English translations us-

	Sentence	Sentence
Reference	Vanunu was released in April, 2004 ...	Why does Aramco donate 8 thousand dollars ...
MT A orig.	And was released in April, 2004 ...	Why ARAMCO to \$ thousands ...
Rule-Based	And was vanunu released in April, 2004 ...	He donates why ARAMCO the amount of dollars to \$ thousands ...
Corpus-Level	Vanunu was released in April, 2004 ...	Why Aramco donate \$ 8 of thousands of dollars ...
	<i>1a) Both APEs re-insert the deleted name, but the rule-based version has poor word order.</i>	<i>1b) Both APEs re-insert the deleted verb, but the feedback word order is better. \$ is incorrectly detected as a function word, and both APEs incorrectly re-insert "dollars". The feedback APE avoids adding the redundant "the amount of".</i>
Reference	... in proportion to the efforts they make.	... Ministry of Interior Starts to Define Committee's Authority!!
MT B orig.	... commensurate with their.	... The Ministry of Interior started to define the terms of the !
Rule-Based	... commensurate with effort exert their.	... The Ministry of Interior started to define the terms of body !
Phrase-Level	... commensurate with the work they do.	... The Interior Ministry started the authority of the board !
	<i>2a) The rule-based APE makes two separate edits to insert "effort" and "exert." The feedback APE produces a more fluent sentence by handling both at once.</i>	<i>2b) The original sentence deletes the noun Committee. The rule-based version has the wrong translation and is ungrammatical. The phrase-level feedback selects a better translation, but the verb (define) is now deleted.</i>

Table 1: Examples of the kinds of edits (both good and bad) made by different APEs.

ing only word alignments, assuming that local Chinese SVO word order would linearly map to English word order.

However, our APEs need to deal with a much wider range of error types, including phrases that were mistranslated, partially translated or never translated; and content words of any POS, not just NEs or verbs. Since Arabic word order differs from English, these rules often produce poorly ordered words: verbs may appear before their subjects, and adjectives may appear after their nouns. In this case, we are explicitly trading off fluency for adequacy, under the assumption that the end task is adequacy-oriented. In example 1a, the subject comes after the auxiliary verb, but the sentence can still be understood. On the other hand, since adequacy and fluency are not independent, degrading the fluency of a sentence can often negatively impact the adequacy as well.

Even when the error detection and correction steps work correctly, not all errors can be fixed with these simple operations. The original MT may be too garbled to correct, or may have no place to insert the corrected translation so that it carries the appropriate meaning.

4.4 Feedback APEs

To mitigate the problems of the rule-based APE, we developed an approach that is more powerful and flexible. The feedback APEs take as input the same list of errors and corrections as the rule-based APE, and then convert the corrections into feedback for the MT system. Sentences with detected errors are decoded a second time with feedback. Passing feedback to the MT system is a general technique: many MT systems allow users to specify certain fixed translations ahead of time, such as numbers, dates and named entities. The underlying implementation of how these fixed translations are

handled by the decoder is MT system-specific, and we describe two such implementations in section 4.5: corpus-level feedback and phrase-level feedback.

The difference between pre-editing and post-editing in this case is that the post-editor is *reactive* to the first-pass translation. The APE only passes suggestions to the MT system when it detects an error in the first-pass translation, and has some confidence that it can provide a reasonable correction. Since the post-editing is actually done by the decoder, the effectiveness of the feedback APE will vary across different MT systems.

This is similar to the error correction approach described in (Parton and McKeown, 2010), where sentences with detected errors are re-translated using a much better (but slower) MT system. They found that the second-pass translations were much better than the first-pass translations, but most of the detected errors were still present. The feedback post-editor allows us to pass specific information about which errors to correct and how to correct them to the original MT system. Unlike adaptive post-editors, where the second translation step translates from "bad" target-language text to "good" target-language text, the feedback APEs re-translate from the source text, and only one MT system is needed.

The biggest advantage the feedback APEs have over the rule-based APE is that the MT system can modify the whole sentence during re-translation, while taking the feedback into account, rather than just replacing or inserting a single phrase at a time. The decoder will not permit local disfluencies that might occur from a simple insertion (e.g., "they goes" or "a impact"), and will often prefer the correct word order, as in example 1a in Table 1. Furthermore, the decoder can take all of the feedback into account at once, whereas the rule-based ap-

proach makes each correction in the sentence separately, as in example 2a. Finally, the rule-based approach always picks the top-ranked correction for each error, and almost always edits every error. The feedback APEs can pass multiple corrections to the MT system, often along with probabilities, which proves helpful in example 2b. One drawback of the feedback APEs is that they are slower than the rule-based APE since they require a second-pass decoding. Also, the decoder may ultimately decide not to use any of the corrections, which may be an advantage if low-confidence suggestions are discarded, or could be a disadvantage, since fewer errors will get corrected.

4.5 Corpus-Level vs. Phrase-Level Feedback

Each of our MT systems has a different mechanism for accepting feedback on-the-fly, and handles the feedback differently. MT A allows *corpus-level feedback* without translation probabilities. In other words, the APE passes all of the translation suggestions for the entire corpus back to the MT system during re-translation. MT B allows *phrase-level feedback* with translation probabilities. Each source phrase flagged as an error is annotated with the list of possible corrections and their translation probabilities. Both MT systems allow multiple corrections for each detected error, unlike the rule-based APE. Both also allow the post-edited corrections to compete with existing translations in the system, so the re-translation may not use the suggested translations. Note that both forms of feedback are used in an online manner by the SMT systems; no re-training or re-tuning is done.

Overall, the phrase-level feedback mechanism is more fine-grained because corrections are targeted at specific errors. On the other hand, the coarser, corpus-level feedback could result in unexpected improvements in sentences where errors were not detected, since the translation corrections can be used in any re-translated sentence.

5 Experiments

We tested our APEs on two different MT systems using the NIST MT08 newswire (nw) and web (wb) testsets, which had 813 and 547 sentences, respectively. The translations were evaluated with multiple automatic metrics as well as crowd-sourced human adequacy judgments.

5.1 MT Systems

We used state-of-the-art Arabic-English MT systems with widely different implementations. MT A was built using HiFST (de Gispert et al.,

2010), a hierarchical phrase-based SMT system implemented using finite state transducers. It is trained on all the parallel corpora in the NIST MT08 Arabic Constrained Data track (5.9M parallel sentences, 150M words per language). The first-pass 4-gram language model (LM) is trained on the English side of the parallel text and a subset of Gigaword 3. The second-pass 5-gram LM is a zero-cutoff stupid-backoff (Brants et al., 2007) estimated using 6.6B words of English newswire text.

MT B was built using Moses (Koehn et al., 2007), and is a non-hierarchical phrase-based system. It is trained on 3.2M sentences of parallel text (65M words on the English side) using several LDC corpora including some available only through the GALE program (e.g., LDC2004T17, LDC2004E72, LDC2005E46 and LDC2004T18). The data includes some sentences from the ISI corpus (LDC2007T08) and UN corpus (LDC2004E13) selected to specifically add vocabulary absent in the other resources. The Arabic text is tokenized and lemmatized using the MADA+TOKAN system (Habash et al., 2009). Lemmas are used for Giza++ alignment only. The tokenization scheme used is the Penn Arabic Treebank scheme (Habash, 2010; Sadat and Habash, 2006). The system uses a 5-gram LM that was trained on Gigaword 4. Both systems are tuned for BLEU score using MERT.

5.2 Automatic and Human Evaluation

We ran several automatic metrics on the baseline MT output and the post-edited MT output: BLEU (Papineni et al., 2002), Meteor-a (Denkowski and Lavie, 2011) and TERp-a (Snover et al., 2009). BLEU is based on n-gram precision, while Meteor takes both precision and recall into account. TERp also implicitly takes precision and recall into account, since it is similar to edit distance. Both Meteor and TERp allow more flexible n-gram matching than BLEU, since they allow matching across stems, synonyms and paraphrases. Meteor-a and TERp-a are both tuned to have high correlation with human adequacy judgments.

In contrast to automatic system-level metrics, human judgments can give a nuanced sentence-level view of particular aspects of the MT. In order to compare adequacy across APEs, we used human annotations crowd-sourced from CrowdFlower.² Since our annotators are not MT experts, we used a head-to-head comparison rather than a 5-point scale. Adequacy scales have been shown

²<http://www.crowdflower.com>

MT set	APE	sents w/err.	sents mod.	
A	nw	rule-based	48%	41%
		corpus feed.	48%	40%
	wb	rule-based	69%	64%
		corpus feed.	69%	62%
B	nw	rule-based	24%	24%
		phrase feed.	24%	15%
	wb	rule-based	34%	34%
		phrase feed.	34%	25%

Table 2: The percentage of all sentences with errors detected, and the percentage of all sentences modified by each APE.

to have low inter-annotator agreement (Callison-Burch et al., 2007). Each annotator was asked to select which of two sentences matched the meaning of one reference sentence the best, or to select “about the same.” The tokens that differed between the translations were automatically highlighted, and their order was randomized. The instructions explicitly said to ignore minor grammatical errors and focus only on how the meaning of each translation matched the reference, and included a number of example judgments.

We compared each post-edited sentence to the baseline MT. For each comparison, we collected five “trusted” judgments (as defined by Crowd-Flower) according to how well they did on our gold-standard questions. For clarity, we are reporting results using macro aggregation, in other words, the number of times overall that a particular APE was voted better than, worse than, or about the same as the original MT.

6 Results

Table 2 shows the percentage of sentences with detected errors for which the correction algorithm found a suggested solution. These sentences were passed to each APE, which could then decide to modify the sentence or leave it unchanged. The percentage of all sentences that were changed by each APE is also shown in Table 2.

The web genre has more errors than the newswire genre, likely because informal text is more difficult for both MT systems to translate. MT A has twice as many sentences with detected errors as MT B. This is not a reflection of relative MT quality (both systems have comparable BLEU scores), but rather a limitation of the error detecting algorithm. When MT A deletes a word, it is frequently dropped as a single token, which is simple to detect as a null alignment. Missing words in MT B are frequently deleted as part of a phrase, so they are more difficult to detect (e.g., mistranslat-

MT set		Δ BLEU			Δ TERp-adeq			Δ Meteor-adeq		
		base MT	rule based	feed back	base MT	rule based	feed back	base MT	rule based	feed back
A	nw	51.32	-0.91	-0.41	37.49	-0.54	-0.74	69.48	+0.15	+0.32
	wb	36.15	-1.41	+0.03	60.66	-1.34	-2.69	55.24	+0.15	+0.88
B	nw	51.23	-0.49	+0.05	35.31	-0.22	-0.26	70.38	+0.00	+0.17
	wb	37.60	-0.50	-0.12	55.97	-0.26	-0.23	57.06	-0.07	+0.13

Table 3: The effect of APEs on automatic metric scores. Base columns show the score for the original MT and the other columns show the difference between the post-edited MT and the original MT. The rule-based APE is the same for both systems, and the feedback APE is corpus-level for MT A and phrase-level for MT B.

ing “white house” as “white” does not get flagged).

The impact of the APEs also varies depending on how many sentences with detected errors were actually changed by the APE. The rule-based APE almost always applies the edits. The corpus-level APE also modified most of the sentences, since all of the corrections were applied to all of the re-translated sentences. However, the phrase-level feedback APE frequently retained the original translation.

Both of these factors mean that the potential improvement from post-editing varies significantly by experimental setting, from only 15% of the sentences by the phrase-based feedback (MT B) on the news corpus, up to 64% of the corpus by the rule-based APE for MT A on the web corpus.

6.1 Automatic Metric Results

Table 3 shows the automatic metric scores for both MT systems, across both datasets. For the baseline MT output, the raw score is shown, and for the APEs, the change in score between the post-edited MT and the baseline MT is shown. (Since post-editing only changes a fraction of sentences in the corpus, the score changes are generally small.)

All APEs improve the TERp-a score across all conditions³, with the feedback APEs often outperforming the rule-based APE. The feedback APEs also improve the Meteor-a score across all conditions, while the rule-based APE has mixed Meteor results. None of the APEs improve the BLEU score: the rule-based APE is always significantly worse than the original MT, while the feedback APEs have either a negative or negligible impact.

The positive improvements in TERp-a and Meteor-a suggest that the APEs are improving adequacy. In general, the feedback APEs improve the automatic scores more than the rule-based APE, although the rule-based APE actually edits more sentences in the corpus than the feedback APEs.

³Since TERp is an error metric, smaller scores are better.

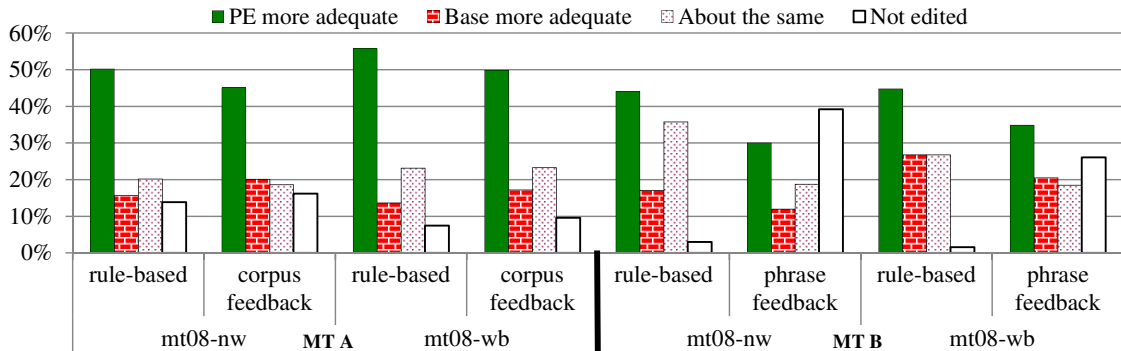


Figure 1: Percentage of post-edited sentences that were judged more adequate, less adequate or about the same as the original MT. “Not edited” is the percentage of sentences with errors that the APE decided not to modify.

The feedback APEs also always have better BLEU scores than the rule-based APE. The negative impact of APEs on BLEU score is not surprising, since they work by adding content to the translations, which is more likely to improve translation recall than precision.

6.2 Human-Annotated Adequacy Results

Figure 1 shows the percentage of post-edited sentences that were judged more adequate, less adequate or the same as the original MT, and the percentage of sentences with errors that the APE did not edit. Of the sentences that were post-edited, the APEs improved adequacy 30-56% of the time. Across both MT systems and both datasets, post-editing improved adequacy much more often than it degraded it: the ratio of improved sentences to degraded sentences varied from 1.7 to 4.1. For both MT systems, the APEs had a larger impact on the web corpus than the newswire corpus, both because more errors were detected in the web corpus and because the APEs edited errors more often in the web corpus.

We were surprised to find that the rule-based APE improved adequacy more often than the feedback APEs, across both MT systems and genres, especially given that the automatic metrics favored the feedback APEs. To understand the results better, we did another crowd-sourced evaluation, comparing the fluency of the rule-based and feedback post-edited sentences (when both APEs made changes). The sentences produced by the feedback APEs were judged more fluent than the rule-based APE sentences across all conditions.

The fluency evaluation shows the relative advantages of the different approaches. The rule-based APE does introduce new, correct information into the translations, but at the expense of fluency. With extra effort, the meaning of these sentences can usually be inferred, especially when the rest of the sentence is fluent (as in example 1a).

On the other hand, the feedback APEs try to balance the post-editor’s request to include more information in the sentence against the goal of the decoder to produce fluent output. But the need for fluency also led to fewer modified sentences, particularly for phrase-level feedback. In cases where both APE approaches improve the adequacy, the feedback approach is better because it produces more fluent sentences. But in cases where the feedback approach does not modify the sentence, the rule-based approach can often still improve the adequacy of the translation at the expense of fluency.

7 Conclusions and Future Work

We described several APE techniques: rule-based in addition to corpus-level and phrase-level feedback. Whereas previous APEs focused primarily on translation fluency and grammaticality, our APEs targeted adequacy errors. Manual analysis showed that post-editing was effective in improving the adequacy of the original MT output 30-56% of the time, across two MT systems and two text genres. The APEs had a larger impact on the web text than the newswire, indicating that they are particularly useful for hard-to-translate genres.

Manual evaluation of the APEs revealed a trade-off between fluency and control. The rule-based APE allowed control over which errors to correct and exactly how to correct them, but was limited to two basic edit operations that often led to disfluent sentences. The feedback APEs produced sentences that were more fluent, but they relied on MT decoders that might or might not carry out the corrections. The corpus-level feedback APE was the least targeted, because suggestions passed to the MT system could affect any re-translated sentence, even those where the phrase was translated correctly. Surprisingly, it was still able to improve adequacy. The phrase-level feedback APE allowed more targeted error correction, yet had the least

impact because it often ignored the corrections.

In future work, we plan to improve the error detection module to handle additional types of adequacy errors, in order to detect more of the adequacy errors made by MT B. We would also like to encourage the phrase-level APE to carry out our corrections more often. Another direction for research is including syntactic information in the rule-based APE, for more fluent translations.

The APEs were motivated by the CLQA task, where adequacy errors can make correct answers appear incorrect after translation. We believe that APE is particularly suitable for task-oriented MT, where black box MT systems must be adapted to the needs of a specific task. We plan to do a task-based evaluation of the adequacy-oriented APEs, to measure their impact on CLQA relevance.

Acknowledgments

This material is based upon work supported by DARPA under Contract Nos. HR0011-12-C-0016 and HR0011-12-C-0014. Any opinions, findings, and conclusions expressed in this material do not necessarily reflect the views of DARPA. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7-ICT-2009-4) under grant agreement number 247762.

References

- Boschee, Elizabeth, Marjorie Freedman, Roger Bock, John Graettinger, and Ralph Weischedel. 2010. Error analysis and future directions for distillation. In *Handbook of Natural Language Processing and Machine Translation*.
- Brants, Thorsten, Ashok C. Popat, Peng Xu, Franz J. Och, and Jeffrey Dean. 2007. Large language models in machine translation. In *EMNLP-CoNLL*, pp. 858–867.
- Buckwalter, Tim. 2004. Buckwalter arabic morphological analyzer version 2.0. *LDC2004L02, ISBN 1-58563-324-0*.
- Callison-Burch, Chris, Cameron Forgy, Philipp Koehn, Christof Monz, and Josh Schroeder. 2007. (meta-) evaluation of machine translation. In *StatMT '07: Proc. of the Second WMT*, pp. 136–158.
- Carpuat, Marine, Yuval Marton, and Nizar Habash. 2012. Improved arabic-to-english statistical machine translation by reordering post-verbal subjects for word alignment. *Machine Translation*, 26:105–120.
- Condon, Sherri L., Dan Parvaz, John S. Aberdeen, Christy Doran, Andrew Freeman, and Marwan Awad. 2010. Evaluation of machine translation errors in English and Iraqi Arabic. In *LREC*.
- de Gispert, Adrià, Gonzalo Iglesias, Graeme Blackwood, Eduardo R. Barga, and William Byrne. 2010. Hierarchical phrase-based translation with weighted finite-state transducers and shallow-n grammars. *Computational Linguistics*, 36(3):505–533.
- Denkowski, Michael and Alon Lavie. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. In *EMNLP 2011: Proc. of the Sixth WMT*.
- Doyon, Jennifer, Christine Doran, C. Donald Means, and Dominique Parr. 2008. Automated machine translation improvement through post-editing techniques: analyst and translator experiments. In *AMTA*, pp. 346–353.
- Elming, Jakob. 2006. Transformation-based corrections of rule-based MT. In *EMT*, pp. 219–226.
- Finkel, Jenny Rose, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by Gibbs sampling. In *ACL*, pp. 363–370.
- Habash, Nizar, Owen Rambow, and Ryan Roth. 2009. MADA+TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. *Proc. of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR)*, pp. 242–245.
- Habash, Nizar. 2010. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.
- Isabelle, Pierre, Cyril Goutte, and Michel Simard. 2007. Domain adaptation of MT systems through automatic post-editing. *MT Summit XI*.
- Knight, Kevin and Ishwar Chander. 1994. Automated post-editing of documents. In *AAAI '94*, pp. 779–784.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *ACL '07: Interactive Poster and Demonstration Sessions*, pp. 177–180.
- Ma, Wei-Yun and Kathleen McKeown. 2009. Where's the verb?: correcting machine translation during question answering. In *ACL-IJCNLP*, pp. 333–336.
- Mareček, David, Rudolf Rosa, Petra Galuščíková, and Ondřej Bojar. 2011. Two-step translation with grammatical post-processing. In *Proc. of the Sixth WMT*, pp. 426–432.
- Papineni, Kishore, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*, pp. 311–318.
- Parton, Kristen and Kathleen McKeown. 2010. MT error detection for cross-lingual question answering. In *COLING (Posters)*, pp. 946–954.
- Parton, Kristen, Kathleen McKeown, James Allan, and Enrique Henestroza. 2008. Simultaneous multilingual search for translingual information retrieval. In *CIKM*, pp. 719–728.
- Popović, Maja and Hermann Ney. 2007. Word error rates: Decomposition over POS classes and applications for error analysis. In *Proc. of the Second WMT*, pp. 48–55.
- Sadat, Fatiha and Nizar Habash. 2006. Combination of arabic preprocessing schemes for statistical machine translation. In *Proceedings of the Conference of the Association for Computational Linguistics*, Sydney, Australia.
- Simard, Michel, Cyril Goutte, and Pierre Isabelle. 2007. Statistical phrase-based post-editing. In *HLT-NAACL*, pp. 508–515.
- Snover, Matthew, Nitin Madnani, Bonnie J. Dorr, and Richard Schwartz. 2009. Fluency, adequacy, or HTER?: exploring different human judgments with a tunable MT metric. In *StatMT '09: Proc. of the Fourth WMT*, pp. 259–268.
- Specia, Lucia, Najeh Hajlaoui, Catalina Hallett, and Wilker Aziz. 2011. Predicting machine translation adequacy. In *MT Summit XIII*.
- Stymne, Sara and Lars Ahrenberg. 2010. Using a grammar checker for evaluation and postprocessing of statistical machine translation. In *Proc. of the Seventh International Conference on Arabic Language Resources and Tools*.
- Suzuki, Hirokazu. 2011. Automatic post-editing based on SMT and its selective application by sentence-level automatic quality evaluation. *MT Summit XIII*.
- Toutanova, Kristina, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL-HLT*, pp. 173–180.
- Vilar, David, Jia Xu, Luis Fernando D'Haro, and Hermann Ney. 2006. Error analysis of machine translation output. In *LREC*, pp. 697–702.

Evaluating User Preferences in Machine Translation Using Conjoint Analysis

Katrin Kirchhoff

Department of Electrical Engineering
University of Washington
Seattle, WA, USA
katrin@ee.washington.edu

Daniel Capurro, Anne Turner

Department of Medical Education
and Biomedical Informatics
University of Washington
Seattle, WA, USA
dcapurro@u.washington.edu
amturner@u.washington.edu

Abstract

In spite of much ongoing research on machine translation evaluation there is little quantitative work that directly measures users' intuitive or emotional preferences regarding different types of machine translation errors. However, the elicitation and modeling of user preferences is an important prerequisite for future research on user adaptation and customization of machine translation engines. In this paper we explore the use of conjoint analysis as a formal quantitative framework to gain insight into users' relative preferences for different translation error types. Using English-Spanish as the translation direction we conduct a crowd-sourced conjoint analysis study and obtain utility values for individual error types. Our results indicate that word order errors are clearly the most dispreferred error type, followed by word sense, morphological, and function word errors.

1 Introduction

Current work in machine translation (MT) evaluation research falls into three different categories: *automatic evaluation*, *human evaluation*, and *embedded application evaluation*. Much effort has focused on the first category, i.e. on designing evaluation metrics that can be computed automatically for the purpose of system tuning and development. These include e.g. BLEU (Papineni et al., 2002), position-independent word error rate (PER), METEOR (Lavie and Agarwal, 2007), or translation error rate (TER) (Snover et al., 2006). Human

evaluation (see (Denkowski and Lavie, 2010) for a recent overview) typically involves rating translation output with respect to fluency and adequacy (LDC, 2005), or directly comparing and ranking two or more translation outputs (Callison-Burch et al., 2007). All of these evaluation techniques provide a *global* assessment of overall translation performance without regard to different error types.

More fine-grained analyses of individual MT errors often include manual or (semi-) automatic error annotation to gain insights into the strengths and weaknesses of MT engines (Vilar et al., 2006; Popovic and Ney, 2011; Condon et al., 2010; Farreus et al., 2012). There have also been studies of how MT errors influence the work of post-editors with respect to productivity, speed, etc. (Krings, 2001; O'Brien, 2011) or the performance of back-end applications like information retrieval (Parton and McKeown, 2010).

In contrast to this line of research, there is surprisingly little work that directly investigates which types of errors are intuitively the most disliked by users of machine translation. Although there is ample anecdotal evidence of users' reactions to machine translation, it is difficult to find formal, quantitative studies of how users perceive the severity of different translation errors and what trade-offs they would make between different errors if they were given a choice. User preferences might sometimes diverge strongly from the system development directions suggested by automatic evaluation procedures. Most automatic procedures do not take into consideration factors such as the cognitive effort required for the resolution of different types of errors, or the emotional reactions they provoke in users. For example, errors that are inadvertently comical or culturally offensive might provoke strong negative user reac-

tions and should thus be weighted more strongly by system developers when user acceptance is a key factor in the intended application. On the other hand, most users might expect, and thus be forgiving of, minor grammatical errors. A deeper insight into which errors are perceived as the most egregious for a particular machine translation application (depending on language pair, domain, etc.) is therefore crucial for improving user acceptance. In addition, user adaptation and customization of MT engines are emerging as important future directions for machine translation research, and it is necessary to develop principled strategies for eliciting and modeling user preferences. However, despite a wealth of existing research on computational preference elicitation techniques little of it has been applied to machine translation evaluation research.

In this paper we explore the use of *conjoint analysis* (CA) to gain knowledge of users' preferences regarding different types of machine translation errors. Conjoint analysis is a formal framework for preference elicitation that was originally developed in mathematical psychology and is widely used in marketing research (Green and Srinivasan, 1978). Its typical application is to determine the reasons for consumers' purchasing choices. In conjoint analysis studies, participants are asked to choose from, rate, or rank a range of products characterized by different combinations of attributes. Statistical modeling, typically some form of multinomial regression analysis, is then used to infer the values ("utilities" or "part-worths") consumers attach to different attributes. In a typical marketing setup the attributes might be price, packaging, performance, etc. In our case the attributes represent different types of machine translation errors and their frequencies. The outcome of conjoint analysis is a list of values attached to different error types across a group of users, along with statistical significance values.

In the remainder of this paper we will first give an overview of the basic techniques of conjoint analysis (Section 2), followed by a description of the data set (Section 3) and experimental design (Section 4). Results and discussion are provided in Section 5. Section 6 concludes.

2 Conjoint Analysis

Conjoint analysis is based on discrete choice theory and studies how the characteristics of a prod-

uct or service influence users' choices and preferences. It is typically used to evaluate and predict purchasing decisions in marketing research but has also been used in analyzing migration trends (Christiadi and Cushing, 2007), decision-making in healthcare settings (Philips et al., 2002), and many other fields. The assumption is that each product or "concept" can be described by a set of discrete attributes and their values or "levels". For example, a laptop can be described by CPU type, amount of RAM, price, battery life, etc. CA generates different concepts by systematically varying the combination of attributes and values and letting respondents choose their preferred one. Clearly, the most preferred and least preferred combinations are known (e.g. a laptop with maximum CPU power, RAM and battery life at the minimum price would be the most preferred). The value of CA derives from studying intermediate combinations between these extremes since they shed light on the trade-offs users are willing to make. In an appropriately designed CA study, each attribute level is equally likely to occur. For a small number of attributes and levels, the total number of possible concepts (defined by different combinations of attributes) is generated and tested exhaustively; if the number of possible combinations is too large, sampling techniques are used. The total set of responses is then evaluated for main effects (i.e. the relative importance of each individual attribute) and for interactions between attributes.

Various different approaches to CA have been developed. The traditional full-profile CA requires respondents to rate or rank all concepts presented. In choice-based conjoint analysis (CBC) (Louviere and Woodworth, 1983) several different concepts are presented, and respondents are required to choose one of them. Finally, adaptive conjoint analysis dynamically adapts and changes the set of concepts presented to respondents based on their previous choices. CBC is currently the most widely used method of conjoint analysis, due to its simplicity: respondents merely need to choose one of a set of proposed concepts, as task which is similar to many real-life decision-making problems. The disadvantage is that the elicitation process is less efficient: respondents need to process the entirety of information presented before making a choice; therefore, it is advisable to only include a small number of concepts to choose from in any given task. CBC is thus appropriate for con-

cepts involving a small number of attributes.

The most frequently-used underlying statistical model for CBC is McFadden’s conditional logit model (McFadden, 1974). The conditional logit model specifies the n possible concept choices as a categorical dependent variable Y with outcomes $1, \dots, n$. The decision of an individual respondent i in favor of the j ’th outcome is based on a utility value u_{ij} , which must exceed the utility values for all other outcomes $k = 1, \dots, n, k \neq j$. It is assumed that u_{ij} decomposes into a systematic or representative part v_{ij} and a random part ε_{ij} ; $u_{ij} = v_{ij} + \varepsilon_{ij}$. A further assumption is that the random components are independent and identically distributed according to the extreme value distribution with cumulative density function

$$F(\varepsilon_{ij}) = e^{-e^{-\varepsilon_{ij}}} \quad (1)$$

The systematic part v_{ij} is modeled as a linear combination $\beta' \mathbf{X}$, where $\mathbf{X} = \{x_1, \dots, x_m\}$ is a vector of m observed predictor variables (the attributes of the alternatives) and β is a vector of coefficients indicating the importance of the attributes. Then, the probability that the i ’th individual chooses the j ’th outcome, $P(j|i)$, can be defined as:

$$P(j|i) = \frac{e^{\beta' \mathbf{X}_{ij}}}{\sum_{k=1}^n e^{\beta' \mathbf{X}_{ik}}} \quad (2)$$

The β parameters are typically estimated by maximizing the conditional likelihood using the Newton-Raphson method. For basic CBC an aggregate logit model is used, where responses are pooled across respondents. In this case a single set of β parameters is used to represent the average preferences of an entire market, rather than individuals’ preferences. This implicitly assumes that respondents form a homogeneous group, which is typically not correct. This oversimplification can be circumvented by applying latent class analysis (Goodman, 1974), which groups respondents into homogeneous subsets and estimates different utility values for each one.

There are numerous advantages to using a formal analysis framework of this type rather than simply questioning users about their experience. First, for a complex “product” like machine translation output, users are notoriously poor at analyzing their own judgments and stating them in explicit terms, especially when they lack linguistic training. It has been noted in the past that it is often difficult for human evaluators to assign consistent

ratings for fluency and adequacy, leading to low inter-annotator agreement (Callison-Burch et al., 2007). Requiring users to rank the output from different systems has proven easier but, as discussed in (Denkowskie and Lavie, 2010), it is still difficult for evaluators to produce consistent rankings. By contrast, the CA framework used here only requires the choice of one out of several possibilities. Users are not asked to provide an objective ranking of several translation possibilities but a single, personal choice, which is an easier task. Furthermore, the choice-based design provides a way of observing trade-offs users make with respect to different types and numbers of errors. For instance, from the user’s point of view, do three morphological errors in one sentence count as much, more, or less than a single word-sense error? Second, CA provides numerical values (“utilities” or “part-worths”) indicating the relative importance of different features of a machine translation output. These might be helpful in machine translation system tuning provided that different error types can be classified automatically. Third, it is also possible to analyze interactions between different attributes, e.g. the effect that a certain combination of errors (e.g. both word order and word sense error present in one sentence) has vs. other combinations. Fourth, different techniques exist to segment the population into different user types (or ‘market segments’) and estimate different utility values for each. However, in this paper only aggregate conjoint analysis will be used, where preferences are analyzed for the entire population surveyed.

2.1 Conjoint analysis for eliciting machine translation user preferences

When applying the conjoint analysis framework to machine translation evaluation we treat different machine translations as different products or “concepts” between which users may choose. We assume that users clearly prefer some machine translations over others, and that these preferences are dependent on the types and frequencies of the errors present in the translation. Thus, error types serve as the attributes of our concepts and the (discretized) error frequencies (e.g. high, medium, low) are the levels. Note that there may be other features of a translation (e.g. sentence length) that may affect a user’s choice – these are not considered in this study but they could easily be included in future studies.

In contrast to most standard applications of conjoint analysis a particular combination of attributes defines not only a single concept but a large set of concepts (alternative translations of a single sentence, or multiple sentences). It is therefore useful to consider a representative sample of sentences for each combination of attributes. Thus, compared Eq. 2 we have another conditioning variable s ranging over sentences:

$$P(j|i, s) = \frac{e^{\beta' \mathbf{X}_{ijs}}}{\sum_{k=1}^n e^{\beta' \mathbf{X}_{iks}}} \quad (3)$$

Our procedure for this study is as follows. First, we select the error types to be investigated. This is done by manually annotating machine translation errors in our data set and selecting the most frequent error types. The different error frequencies are quantized into a small number of levels for each error type. We then generate different profiles (combinations of attributes/levels) and group them into choice tasks – these are the combinations of profiles from which respondents will choose one. Respondents’ choices are gathered through Mechanical Turk. Finally, we estimate a single set of model parameters, aggregating over both respondents and sentences, and compute statistical significance values. Additionally, we perform prediction experiments, using the estimated utility values to predict users’ choices on held-out data.

3 Data

The data used for the present study was collected as part of a research project on applying machine translation to the public health domain. It consists of information materials on general health and safety topics (e.g. HIV, STDs, vaccinations, emergency preparedness, maternal and child health, diabetes, etc.) collected from a variety of English-language public health websites. The documents were translated into Spanish by Google Translate (<http://www.google.com/translate>). 60 of these documents were then manually annotated for errors by two native speakers of Spanish. Our error annotation scheme is similar to other systems used for Spanish (Vilar et al., 2006) and comprises the following categories:

1. **Untranslated word.** These are original English words that have been left untranslated by the MT engine and that are not proper names or English words in use in Spanish.

Type	%	Subtypes	%
Morphology	28.2	Verbal Nominal	15.8 12.4
Missing word	16.7	Function word Content word	12.6 4.1
Word sense error	16.1		
Word order error	9.7	short range long range	8.0 1.7
Punctuation	9.1		
Other	5.9		
Spelling	5.1		
Superfluous word	4.7	Function word Content word	3.8 0.9
Capitalization	2.7		
Untranslated word	1.1	medical term proper name other	0.0 0.2 0.9
Pragmatic	1.0		
Diacritics	0.2		
Total	100.0		

Table 1: Error statistics from manual consensus annotation of 25 documents. The two right-hand columns show error subtypes.

2. **Missing word.** A word necessary in the output is missing – a further distinction is made between missing function words and missing content words.
3. **Word sense error.** The translation reflects a word sense of the English word that is wrong or inappropriate in the present context.
4. **Morphology.** The morphological features of a word in the translation are wrong.
5. **Word order error.** The word order is wrong – a further distinction is made between short-range errors (within a linguistic phrase, e.g. adjective-noun ordering errors) and long-range errors (spanning a phrase boundary).
6. **Spelling.** Orthographic error.
7. **Superfluous word.** A word in the translation is redundant or superfluous.
8. **Diacritics.** The diacritics are faulty (missing, superfluous, or wrong).
9. **Punctuation.** Punctuation signs are missing, wrong, or superfluous.
10. **Capitalization.** Missing or superfluous capitalization.
11. **Pragmatic/Cultural error.** The translation is unacceptable for pragmatic or cultural reasons, e.g. offensive or comical.
12. **Other.** Anything not covered by the above categories.

Annotators were linguistically trained and were supervised in their annotation efforts.

For a subset of 25 of these documents (1804 sentences), the annotators were instructed to create

a consensus error annotation, and to subsequently correct the errors, thus producing consensus reference translations. Computing BLEU/PER scores against the corrected output yields a BLEU score of 65.8 and a PER of 19.8%. Unsurprisingly, these scores are very good since the reference translations are corrections of the original output rather than independently created translations – however, annotators independently judged the overall translation quality as quite good as well. The detailed errors statistics computed from the 25 documents is shown in Table 1. The most frequent error types are, in order: morphological errors, word sense errors, missing function words, and word order errors. Based on this we defined four error types to be used as the attributes in our conjoint analysis study: word sense errors (S), morphology errors (M), word order errors (O) and function word errors (F) – the latter includes both missing and superfluous function words. For word sense, word order, and function word errors we defined two values (levels): high (H) and low (L). Since morphology errors are much more frequent than others we use a three-valued attribute in this case (high, medium (M), and low).

From these documents we selected 40 sentences, each of which contained a minimum of one instance each of sense, order and function word errors, and a minimum of two instances of morphological errors. Based on the error annotations and their manual corrections, each sentence can be edited selectively to reflect different attribute levels, i.e. different numbers of errors of a given type. For example, different versions of a sentence are created that exhibit a high, medium, or low level of morphological errors. The variable number of errors are mapped to the discrete attribute levels as follows: If the total number of errors for a given type is ≤ 2 , then $H = 2$ errors and $L = 0$ errors for the binary attributes, and $H=2$, $M=1$, $L=0$ for the three-valued attribute. When the number of errors is larger than 2, the interval size for each level is defined by the number of errors divided by the number of levels, rounded to the nearest integer.

The number of all possible different combinations of attributes/levels is 24; thus, for each sentence, 24 concepts or “profiles” are constructed. A partial example is shown in Table 2.

4 Experiments

We chose a full factorial experiment design, i.e. each of the 24 possible profiles was utilized for each of the 40 sentences. Each partially-edited sentence represents a different profile. However, not all 24 profiles can be presented simultaneously to a single respondent – typically, CBC surveys need to be kept as small and simple as possible to prevent respondents from resorting to simplification strategies and delivering noisy response data. Profiles were grouped into choice tasks with three alternatives each, representing a balanced distribution of attribute levels.

For each survey, 4 choice tasks were randomly selected from the total set of choice tasks. The questions in the survey thus included profiles pertaining to different sentences, which was intended to avoid respondent fatigue. Surveys were presented to respondents on the Amazon Mechanical Turk platform. For each choice task, Turkers were instructed to carefully read the original source sentence and the translations provided, then choose the one they liked best (an obligatory choice question with the possibility of choosing exactly one of the alternatives provided), and to state the reason for their preference (an obligatory free-text answer). The latter was included as a quality control step to prevent Turkers from making random choices. The set of Turkers was limited to those who had previously delivered high-quality results in other Spanish translation and annotation HITs we had published on Mechanical Turk. In total we published 240 HITs (surveys) with 4 choice tasks and 3 assignments each, resulting in a total of 2880 responses. A total of 29 workers completed the HITs, with a variable number of HITs per worker. The responses were analyzed using the conditional logit model implementation in the R package.¹

5 Results and Discussion

We first measured the overall agreement among the three different responses per choice task using Fleiss’s Kappa (Fleiss, 1971). The kappa coefficient was 0.35, which according to (Landis and Koch, 1977) constitutes “fair agreement” but does indicate that there is considerable variation among workers regarding their preferred translation choice. We next estimated the coefficients of the conditional logit model considering main ef-

¹<http://www.r-project.org>

No.	Attributes	Sentence
1	S=H:M=H:O=H:F=H	Planear con anticipación y tomar un atajo pocos ahorrar su tiempo y su dinero para alimentos.
2	S=H:M=H:O=H:F=L	Planear con anticipación y tomar un atajo le pocos ahorrar su tiempo y su dinero para la alimentos.
3	S=H:M=H:O=L:F=H	Planear con anticipación y tomar un pocos atajo ahorrar su tiempo y su dinero para alimentos.
4	S=H:M=H:O=L:F=L	Planear con anticipación y tomar un pocos atajo le ahorrar su tiempo y su dinero para la alimentos.
5	S=H:M=M:O=H:F=H	Planear con anticipación y tomar un atajo pocos ahorrar su tiempo y su dinero para alimentos.
6	S=H:M=M:O=H:F=L	Planear con anticipación y tomar un atajo le pocos ahorrar su tiempo y su dinero para la alimentos.
7	S=H:M=M:O=L:F=H	Planear con anticipación y tomar un pocos atajo ahorrará su tiempo y su dinero para alimentos.
8	S=H:M=M:O=L:F=L	Planear con anticipación y tomar un pocos atajo le ahorrará su tiempo y su dinero para la alimentos.
9	S=H:M=L:O=H:F=H	Planear con anticipación y tomar unos atajos pocos ahorrará su tiempo y su dinero para alimentos.
10	S=H:M=L:O=H:F=L	Planear con anticipación y tomar unos atajos le pocos ahorrará su tiempo y su dinero para la alimentos.
	etc.	etc.
24	S=L:M=L:O=L:F=L	Planear con anticipación y realizar unos pocos recortes le ahorrará su tiempo y su dinero para la comida.

Table 2: Examples of the 24 attribute combinations and corresponding partially-edited translations for the English input sentence *Planning ahead and taking a few short cuts will save both your time and your food dollars.*

Variable	β	$\exp(\beta)$	α
O	-1.125	0.3246	0.001
S	-0.6302	0.5325	0.001
M	-0.4034	0.6680	0.001
F	-0.1211	0.8859	0.001

Table 3: Estimated coefficients in the conditional logit model and associated significance levels (α) – main effects. O = word order, S = word sense, M = morphology, F = function words.

fects only. The model’s β coefficients, exponentiated β ’s, and significance values are shown in Table 3. It is easiest to interpret the exponentiated β coefficients: these represent the change in the odds (i.e. odds ratios) of the error type being associated with the chosen translation, for each unit increase in the error level and while holding other error levels constant. For example, if the level of word sense errors is increased by 1 (i.e. goes from low to high) while other error types are being held constant, the odds of the corresponding translation being chosen decrease by a multiplicative factor of 0.5325 (i.e. roughly 50%). Overall we see that word order errors are the most dispreferred, followed by word sense, morphology, and function word errors. All values are highly significant ($p < 0.001$, two-sided z-test). We next tested all pairwise interactions between individual attributes. An interaction between two attributes means that the impact of one attribute on the outcome is dependent on the level of the other attribute. We found two statistically significant interactions, between word sense and function word

Variable	β	$\exp(\beta)$	α
O	-1.149e+00	3.169e-01	0.001
S	-1.079e+00	3.398e-01	0.001
M	-6.971e-01	4.980e-01	0.001
F	-8.932e-01	4.094e-01	0.001
M:F	2.081e-01	1.231e+00	0.001
S:F	2.649e-01	1.303e+00	0.01

Table 4: Estimated coefficients in the conditional logit model and associated significance values (α) – interactions. O = word order, S = word sense, M = morphology, F = function words. Variables containing “:” denote interaction terms.

errors, and between morphological and function word errors. The meaning of the coefficients in Table 4 changes with the introduction of interaction terms, and they cannot directly be compared to those in Table 3. In particular, the $\exp(\beta)$ for M:F and S:F now need to be interpreted as *ratios of odds ratios* for unit increases in the attribute levels. The values (> 1) indicate that the odds ratio of a positive choice associated with a unit increase in function word error level actually increases as the level of M or S errors rises – e.g. the odds ratio for S=*high* is 0.4462 ($\exp(\beta_S + \beta_{S:F})$) vs. 0.3398 for S=*low*). This means that function word errors have a stronger impact on respondents’ choices at low levels of morphological or word sense errors; by contrast, when the level of the latter is high, respondents are less sensitive to function word errors. This effect is also observable for word order and function word errors but it is not statistically significant.

	Accuracy (%)	Stddev
Clogit	54.68	1.99
Fewest errors	49.49	2.70
Random	33.33	0.0

Table 5: Average cross-validation accuracy and standard deviation of conditional logit model, fewest-errors-baseline, and random baseline.

A standard way of validating the overall explanatory power of the model is to perform prediction on a held-out data set. To this end we compute the probability of each choice in a set according to Eq. 3 by inserting the estimated β coefficients and take the max over j , which can be simplified as:

$$j^* = \max_j \beta^t X_{ijs} \quad (4)$$

$$(5)$$

The percentage of correctly identified outcomes (the “hit rate” or accuracy) is then used to assess the quality of the model.

We performed 8-fold cross-validation. For each fold one eighth of the data for each sentence was assigned to the test set; the rest was assigned to the training set. Table 5 shows the average accuracies for our conditional logit model as well as two baselines. The first is the random baseline – each training/test sample is a choice task with 3 alternatives; thus, choosing one alternative randomly results in a baseline accuracy of 33.3%. The second baseline consists of choosing the alternative with the lowest number of errors overall. This leads to accuracies ranging from 45.75%-53.75%, with an average of 49.59%. The accuracies obtained by our model with the fitted coefficients range from 53.00%-58.75%, with an average of 54.06%. This is significantly better than the random baseline and clearly better (though not statistically significant) than the fewest-errors baseline. Nevertheless there clearly is room for improvement in the predictive accuracy of the model. The model shows virtually the same performance (54.04% accuracy on average) on the training data; thus, generalization ability is not the problem here. Rather, the difficulty lies in the underlying variability of the data to be modelled, in particular the diversity of the user group and the sentence materials. For example, no distinction has been made between short-range and long-range word order errors, although it may be assumed that long-range word order errors are considered more severe by users than short-range

errors. Another source of variability is the respondent population itself – since we only used aggregate conjoint analysis in this study, preferences are averaged over the entire population, ignoring potential sub-classes of users. It may well be possible that some user types are more accepting of e.g. word-order errors than word sense errors, or vice versa – recall that the agreement coefficient on the top choice was only 0.35. Finally, another confounding factor might be the quality of the Mechanical Turk data. Although we took several steps to ensure reasonable results, responses may not be as reliable as in a face-to-face study with respondents.

6 Conclusions and Future Work

We have studied the use of conjoint analysis to elicit user preferences for different types of machine translation errors. Our results confirms that, at least for the language pair and population studied, users do not necessarily rely on the overall number of errors when expressing their preferences for different machine translation outputs. Instead, some error types affect users’ choices more strongly than others. Of the different error types considered in this study, word order errors have the lowest frequency in our data but are the most dispreferred error type, followed by word sense errors. The most frequent error type in our data, morphology errors, is ranked third, and function word errors are the most tolerable. The viability of the conjoint analysis framework was demonstrated by showing that the prediction accuracy of the fitted model exceeds that of a random or fewest-errors baseline.

In future work the overall predictive power of the model could be improved by more fine-grained modeling of different sources of variability in the data. Specifically, we plan to compare the present results to results from face-to-face experiments, in order to gauge the reliability of crowd-sourced data for conjoint analysis. In addition, latent class analysis will be used in order to obtain preference models for different user types. In the long run, such models could be exploited for rapid user adaptation of machine translation engines after eliciting a few basic preferences from the user. Utility values obtained by conjoint analysis might also be used in MT system tuning, by appropriately weighting different error types in proportion to their utility values; however, this would require high-accuracy

automatic classification of different error types.

Another way of extending the present analysis is to elicit user preferences in the context of a specific task to be accomplished; for instance, users could be asked to indicate their preferred translation when faced with the tasks of postediting or extracting information from the translation. Finally, it is also possible to investigate a larger set of error types than those considered in this study. These may include different types of word order errors (long-range vs. short-range), consistency errors (where a source term is not translated consistently in the target language throughout a document), or named-entity errors.

Acknowledgments

We are grateful to Aurora Salvador Sanchis and Lorena Ruiz Marcos for providing the error annotations and corrections. This study was funded by NLM grant 1R01LM010811-01.

References

- Callison-Burch, C., C. Fordyce, P. Koehn, C. Monz, and J. Schroeder. 2007. (Meta-)evaluation of machine translation. In *Proceedings of WMT*, pages 136–158.
- Christiadi and B. Cushing. 2007. Conditional logit, IIA, and alternatives for estimating models of interstate migration. In *Proceedings of the 46th Annual Meeting of the Southern Regional Science Association*.
- Condon, S., D. Parvaz, J. Aberdeen, C. Doran, A. Freeman, and M. Awad. 2010. Evaluation of machine translation errors in English and Iraqi Arabic. In *Proceedings of LREC*.
- Denkowskie, M. and A. Lavie. 2010. Choosing the right evaluation for machine translation: an examination of annotator and automatic metric performance on human judgment tasks. In *Proceedings of AMTA*.
- Farreus, M., M.R. Cosa-Jussa, and M. Popovic Morse. 2012. Study and correlation analysis of linguistic, perceptual, and automatic machine translation evaluations. *Journal of the American Society for Information Science and Technology*, 63(1):174–184.
- Fleiss, J.L. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.
- Goodman, L.A. 1974. Exploratory latent structure analysis using both identifiable and unidentifiable models. *Biometrika*, 61(2):215–231.
- Green, P. and V. Srinivasan. 1978. Conjoint analysis in consumer research: Issues and outlook. *Journal of Consumer Research*, 5:103–123.
- Krings, H. 2001. *Empirical Investigations of Machine Translation Post-Editing Processes*. Kent State University Press.
- Landis, J.R. and G.G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33:159–174.
- Lavie, A. and A. Agarwal. 2007. Meteor: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 28–231.
- LDC. 2005. Linguistic data annotation specification: Assessment of fluency and adequacy in translations. revision 1.5. Technical report, LDC.
- Louviere and Woodworth. 1983. Design and analysis of simulated consumer choice experiments: an approach based on aggregate data. *Journal of Marketing Research*, 20(4):350–67.
- McFadden, D.L. 1974. Conditional logit analysis of qualitative choice behavior. In Zarembka, P., editor, *Frontiers in Econometrics*, pages 105–142. Academic Press: New York.
- O’Brien, S., editor. 2011. *Cognitive Explorations of Translation: Eyes, Keys, Taps*. Continuum.
- Papineni, K., S. Roukos, and T. Ward. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318.
- Parton, K. and K. McKeown. 2010. MT error detection for cross-lingual question answering. In *Proceedings of Coling*.
- Philips, K., T. Maddala, and F.R. Johnson. 2002. Measuring preferences for health care interventions using conjoint analysis. *Health Services Research*, pages 1681–1705.
- Popovic, M. and H. Ney. 2011. Towards automatic error analysis of machine translation output. *Computational Linguistics*, 37(4):657–688.
- Snover, M., B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of AMTA*.
- Vilar, D., J. Xiu, L.F. D’Haro, and H. Ney. 2006. Error analysis of statistical machine translation output. In *Proceedings of LREC*.

Poster Session 3 – Research and Project Papers

Cascaded Phrase-Based Statistical Machine Translation Systems

Dan Tufiş

Research Institute for Artificial Intelligence,
Romanian Academy
Bucharest, Romania
tufis@racai.ro

Ştefan Daniel Dumitrescu

Research Institute for Artificial Intelligence,
Romanian Academy
Bucharest, Romania
sdumitrescu@racai.ro

Abstract

Statistical-based methods are the prevalent approaches for implementing machine translation systems today. However the resulted translations are usually flawed to some degree. We assume that a statistical baseline system can be re-used to automatically learn how to (partially) correct translation errors, i.e. to turn a “broken” target translation into a better one. By training and testing on initial bilingual data, we constructed a system S1 which was used to translate the source language part of the training corpus. The new translated corpus and its reference translation are used to train and test another similar system S2. Without any additional data, the chain S1+S2 shows a sensible quality increase against S1 in terms of BLEU scores, for both translation directions (English to Romanian and Romanian to English).

1 Introduction

The paper presents a cascaded phrase based translation system that obtains improved translation scores using no additional data compared to the standard single-step translation system.

The first challenge of our research was to obtain the best standard translation system possible. We experimented with different factored models that include surface form, lemmas and different part of speech tag sets in various combinations to confirm the assumption that translation accuracy is improved over a surface form only baseline model.

The second objective of our work was to validate our intuition that a statistical baseline system can be re-used (cascaded) to automatically

learn how to (partially) correct its own translation errors, i.e. to turn an initially “broken” translation into a better one.

The phrase-based translation approach has overcome several drawbacks of the word-based translation methods and proved to significantly improve the quality of translated output. The morphology of a highly inflected language permits a flexible word order, thus shifting the focus from long-range reordering to the correct selection of a morphological variant.

Morphologically rich languages have a large number of surface forms in the lexicon to compensate for a flexible word order.

Both Transfer and Interlingua MT employ a generation step to produce the surface form from a given context and a lemma of the word. In order to allow the same type of flexibility in using the morpho-syntactic information in translation, factored translation models (Koehn and Hoang, 2007) provide the possibility to integrate the linguistic information into the phrase-based translation model.

Most of the statistical machine translation (SMT) approaches that have a morphologically rich language as target employ factored translation models. Our approach is similar to several other factored machine translation experiments such as adding the morphological features as factors (Avramidis and Koehn, 2008). Our results confirm findings of other researchers, namely that when very large parallel corpora are available, minimal pre-processing is sufficient to get better results than the baseline (raw data); however, when only a limited amount of training data is available, better results are achieved with part-of-speech tags and complex morphological analysis (Habash and Sadat, 2006).

Romanian is a morphologically rich language which needs more than 1200 lexical tags in order to be compliant with the Multext-East lexical specifications (Erjavec and Monachini, 1997).

Czech and Slovene require more than 2000 such morpho-lexical descriptors (MSDs). These descriptors encode detailed linguistic information (gender, case, modality, tense etc.) which can be extremely useful for an accurate translation based on factored models. The set of MSDs can be reduced without information loss by exploiting the redundancy between various feature-value combinations in these descriptors. Yet, the resulting tagsets are too large and thus the data-sparseness hampers the reliability of automatic assignment of MSDs to arbitrary new texts.

Tiered tagging (Tufiş, 1999) is a two-stage technique addressing the issue of training data sparseness. It uses an automatically induced intermediary tag-set, named CTAG tagset, of a smaller size on the basis of which a common POS tagging technique can be used. In a second phase, it replaces the tags from the small tag-set with tags from the fully-specified morpho-syntactic tag-set (MSD tag-set) also taking into consideration the context. The second phase of tiered tagging relies on a lexicon and a set of hand-written rules. The original idea of tiered tagging has been extended in (Ceaşu, 2006), so that the second phase is replaced with a maximum entropy-based MSD recovery. In this approach, the rules for CTAG to MSD conversion are automatically learnt from the corpus. Therefore, even the CTAG labels assigned to unknown words can be converted into MSD tags. If an MSD-lexicon is available, replacing the CTAG label for the known words by the appropriate MSD tags is almost 100% accurate.

2 System overview

Factored translation models extend the phrase-based translation by taking into account not only the surface form of the phrase, but also additional information like the dictionary form (lemma), the part-of-speech tag or the morpho-syntactic specification. It also provides, on the target side, the possibility to add a generation step. All these new features accommodate well in the log-linear model employed by many decoders:

$$P(e|f) = \exp \sum_{i=1}^n \lambda_i h_i(e, f) \quad (1)$$

where $h_i(e, f)$ is a function associated with the pair e, f and λ_i is the weight of the function.

To improve the translation into morphologically-rich languages, the multitude of options provided by the factored translation can help validate the following assumptions:

- a) Aligning and translating *lemma* could significantly reduce the number of translation equivalency classes, especially for languages with rich morphology;
- b) *Part of speech affinities*. In general, the translated words tend to preserve their part of speech and when this is not the case, the part-of-speech chosen is not random;
- c) The *re-ordering* of the target sentence words can be improved if language models over POS or MSD tags are used.

In order to test the improvement of the factored model over the phrase-based approach, we built strong baseline systems for the RO-EN language pair (Ceaşu and Tufiş, 2011).

The intuition that motivated our experiments is that the same methodology used in translating from language A into language B could be applied for (partially) correcting the initial translation errors. We wanted to validate this idea without recourse to additional resources. To this end, we built a two – layered cascaded translation system.

The first step was to create the best possible direct translation system S1 for $A \rightarrow B$. For this we started from a parallel corpus: $\{C_A, C_B\}$. Using this corpus we trained a factored phrased-based translation model. Having the $A \rightarrow B$ system obtained (Ceaşu and Tufiş, 2011), we prepared for the second system S2 by translating the entire training corpus C_A into language B, obtaining $T_{S1}(C_A)$. Using the new parallel corpus $\{T_{S1}(C_A), C_B\}$ we trained the second system S2.

At this point we chained the two systems together: we give an input text I_A (in language A), the first system translates I_A to $T_{S1}(I_A)$ which is the input for the second system. Thus, the chained system receiving the input I_A produces the output $O_B: T_{S2}(T_{S1}(I_A))$.

We further present the steps taken to build this cascaded system and compare the translation performance against the direct, S1 one-step system.

3 Data Preparation

The corpus used to train any SMT system has the biggest influence on translation quality, so special attention is given to its preparation. For the purposes of this paper we used the bilingual parallel corpus (Romanian-English) that had been developed during the ACCURAT FP-7 research project. We chose this resource because it is a reasonably large parallel corpus between a highly inflectional language (Romanian) and a less inflectional reference language (English).

The content of the corpus is drawn from several other corpora:

- 1) DGT-TM¹, law and juridical domain, approx. 650,000 sentences;
- 2) EMEA (Tiedemann, 2009), medical corpus, approx. 994,000 sentences;
- 3) Romanian-English part of the multilingual thesaurus Eurovoc², (1-5 words), approx. 6,500 bilingual terms, treated as short sentences;
- 4) PHP³, translation of the PHP software manual, approx. 30,000 sentences;
- 5) KDE⁴, translation of the Linux KDE interface, approx. 114,000 sentences;
- 6) SETIMES⁵, news corpus, approx. 170,000 sentences.

In total, the source Romanian – English corpus has over 1,950,000 sentences. However, the corpus needed to be cleaned and annotated. This was performed in three steps:

1) Step 1 – Initial corpus cleaning – We created a cleaning application that removes duplicate lines (ex: the PHP corpus contains many identical lines), lines that contain only/mostly numbers (such as lines that consist only of telephone numbers), lines that contain no Latin characters, lines that contain less than 3 characters and other similar heuristics. Additionally, there are three specific types of text distortions occurring in Romanian texts: (i) missing diacritical characters, (ii) different encoding codes for the same diacritical characters, and (iii) different orthographic systems. When ignored, they have a negative impact on the quality of translation and language models and, thus, on the translation results. For details on the process of diacritics restoration, see (Tufiş and Ceauşu, 2008).

2) Step 2 – Corpus annotation – The parallel corpus was annotated using our NLP tools (Tufiş et al., 2008) that tokenize, lemmatize and tag the input text. The tagger does its job both in terms of CTAG and MSD tagsets. This annotation was performed for both Romanian and English sides of the corpus. The annotation has the Moses (Koehn et al., 2007) input file structure. For example, the sentence: “Store in the original package.” has been annotated as shown in Table 1, one token per line followed by three additional fields, separated by “|”:

<i>English</i>	<i>Romanian</i>
Store store ^{Nc} NN Ncns	A avea ^{Va} VA3S Va--3s
in in ^{Sp} PREP Sp	se sine ^{Px} PXA Px3--a-----w
the the ^{Dd} DM Dd	pastra pastra ^{Vm} V3 Vmii3s
original original ^{Af}	în în ^{Sp} S Spsa
package package ^{Nc} NN Ncns	ambalajul ambalaj ^{Nc} NSRY Ncmsry
	original original ^{Af} ASN Afpms-n

Table 1. EN-RO annotated sentence pair

- 0 – surface form – the token itself;
- 1 – lemma of the token, trailed (^) with the grammar category;
- 2 – CTAG – tag from the reduced tagset;
- 3 – MSD – Morpho-Syntactic Annotation tag.

3) Step 3 – Final cleaning – The last step involved using the Moses cleaner, a Perl script that ensured that the corpus did not contain illegal characters, spaces, etc. and that the two corpus sides (Romanian – English) had an equal number of sentences.

After these cleaning steps the RO-EN corpus was reduced to around 1,250,000 sentences. Finally, the corpus was randomized and 1200 sentence-pairs ($T_{RO}-T_{EN}$) were extracted that represent the RO-EN test files.

4 Translation experiments

5.1 First layer translation system (S1)

The first step was to decide on a model for the direct Romanian ↔ English translation. Several models have been proposed and tested. Using the Moses SMT software, we have created the following models (we have experimented with several more models, but kept here only the top performers for reference):

<i>Model #</i>	<i>Details</i>
#1	t0-0 m0
#2	t1-1 g1-0 m0
#3	t1-1 g1-3 t3-3 g1,3-0 , m0m3
#4	t1-1 g1-3 t3-3 g1,3-0 , m0m3 r0
#5	t1-1 g1-3 t3-3 g1,3-0 , m0m3 r3

Table 2. Models description for the first layer

Notation: t = translation step, g = generation step, m = language model, r = reordering model. The first model (#1) simply translates surface forms in language A to surface forms in language

¹ <http://langtech.jrc.it/DGT-TM.html>

² <http://eurovoc.europa.eu/>

³ <http://www.php.net/>

⁴ <http://docs.kde.org/>

⁵ <http://www.setimes.com/>

B (t0-0). The second model (#2) first translates lemmas in language A to lemmas in language B (t1-1) and then employs a generation step to generate surface forms in language B from lemmas in language B (g1-0). The third, fourth and fifth models (#3, #4, #5) follow a more complex path. They first start with a lemma-lemma translation (t1-1), followed by a lemma to MSD generation in language B (g1-3), a translation of MSDs in language A to MSDs in language B (t3-3) and finally generating surface forms from the previously translated lemmas and MSDs in language B (g1,3-0). They use two language models. While models #1 and #2 use just a surface language model, models #3, #4 and #5 additionally use a MSD language model. The difference between models #3, #4 and #5 is that model #4 uses a reordering model based on surface forms while model #5 uses reordering based on MSDs. Table 3 presents the BLEU scores (Papineni et al., 2002) obtained testing the five proposed models.

For the Romanian \rightarrow English direction, model #3 was the best performing of the five, with a BLEU score of 57.01. For the English \rightarrow Romanian direction, scores were a bit lower, model #2 having the highest 53.94 BLEU points.

Interestingly, the large size of the corpus shows its power, bringing the score of the unfactored model #1 very close to the factored models.

The next step was to estimate the translation time of the corpus. This was necessary because of the size of the training corpus: approx. 1.25 million sentences. Moses offers two different translation options: the default translation search and the cube pruning search algorithm. There are two adjustable parameters: the stack size and beam search. These parameters have been manually specified to obtain insights about their influence on translation speed and quality. We present only model #3 for the RO \rightarrow EN direction.

The translation time includes language model and translation/generation tables loading time. The test machine is a dedicated 16 core (8 physical + 8 virtual, running at 2.6GHz), 12 GM RAM server.

<i>RO \rightarrow EN</i>		<i>EN \rightarrow RO</i>	
<i>Model #</i>	<i>BLEU</i>	<i>Model #</i>	<i>BLEU</i>
#1	56.31	#1	52.43
#2	56.49	#2	53.94
#3	57.01	#3	49.97
#4	56.79	#4	49.12
#5	56.89	#5	48.70

Table 3. S1: Model scores

<i>Stack Size</i>	<i>Beam</i>	<i>Translation</i>	<i>BLEU</i>
<i>Param.</i>	<i>Search</i>	<i>Time (s)</i>	<i>Score</i>
	<i>Param.</i>		
(default)	(default)	3074	57.01
100	(default)	1611	56.69
50	(default)	831	56.05
20	(default)	391	54.97
15	(default)	307	54.36
10	(default)	229	53.16
5	(default)	144	51.35
(default)	100	83	39.17
(default)	10	83	43.29
(default)	2	87	47.17
(default)	1	93	49.63
(default)	0.5	151	51.80
(default)	0.1	169	55.84
100	1	106	49.63
Cube pruning algorithm with stack size 2000		167	56.29

Table 4. S1: Parameter variation, translation time and BLEU scores.

Table 4 shows measurements for the translation times and BLEU scores (RO \rightarrow EN direction) of the test files (1200 sentences), for different settings of the Stack Size and Beam Search.

Even though the best performing translation was achieved using the default parameters (BLEU score: 57.01), due to the very long translation time, we found that the best compromise was to use the cube pruning algorithm with the stack size 2000 that obtains a marginally lower BLEU score of 56.29. When using the cube pruning algorithm, we found that, for our test set, increasing the stack size to more than 2000 does not generate any noticeable score improvements.

Based on these results, we have used the two best performing models (model #3 for the RO \rightarrow EN direction and model #2 for the EN \rightarrow RO direction) with the cube pruning search algorithm to translate both languages of the parallel corpus $\{C_{RO}, C_{EN}\}$. We obtained two new corpora: for the RO \rightarrow EN direction we obtained the $\{T_{SI}(C_{RO}), C_{EN}\}$ corpus, and for the EN \rightarrow RO direction we obtained the $\{C_{RO}, T_{SI}(C_{EN})\}$ corpus.

After the translation, the final phase of this step was to process the two newly obtained corpora. Using the same NLP tool we used to annotate the original corpus we annotated the translated corpora with lemma, CTAGs and MSDs. Finally, the annotated corpora were cleaned again, but using only step 3 (the Moses cleaning script) of the cleaning process described in section 3. The cleaning yielded for the RO \rightarrow EN direction a corpus of around 1,110,000 sentences (losing in this second cleaning process about

140,000 sentences - around 11% - from the initial 1,250,000), while for the EN→RO direction the corpus lost almost 240,000 sentences resulting in a corpus of 1,010,000 sentences.

5.2 Second layer translation system (S2)

For this step, using the intermediary corpus, we trained 9 models to see which one would perform best. Table 5 shows the models chosen and table 6 shows the translation and BLEU scores using the cube pruning and default translation algorithms. The same models were used for both translation directions.

<i>Model</i>	<i>Details</i>
#1	t0-0 m0
#2	t1-1 g1-0 m0
#3	t1-1 g1-2 t2-2 g1,2-0 m0,m2
#4	t1-1 g1-3 t3-3 g1,3-0 m0,m3
#5	t1-1 g1-3 t3-3 g1,3-0 m0,m3 r3
#6	t1-1 g1-2 t2-2 g2-3 t3-3 g1,3-0 m0,m2,m3
#7	t0,1-0,1 m0
#8	t0,1,2-0,1,2 m0,m2
#9	t1,2-t1,2 m0,m2

Table 5. S2: Models description

Translating was performed with both default parameters and using the cube pruning search with stack size 2000. The reordering model is the Moses default, with the only difference that in model 5 we have used MSDs as the reordering factor.

For testing S2 we used the same test files as for S1, but translated with the best S1 models: the model #3 for RO→EN direction and the model #2 for the EN→RO direction (see Table 3). The reference translations for the two directions were T_{EN} and T_{RO} respectively (1200 sentences each).

For the RO→EN direction the BLEU translation score of the S1+S2 system has been improved from the best S1 model (57.01) to a new BLEU score of 60.90.

The fact that S2 translation based on model #7 (surface form & lemma to surface form & lemma using only the surface language model) was the fastest and most accurate is not surprising: we “translated” from partly broken English into presumably better English.

Generation steps were not necessary and the information on the lemma eliminated some candidates from the search space.

Interestingly, the translation time the using default Moses parameters is very close to the cube

<i>Model</i>	<i>Transl. time (s)</i>	<i>BLEU with cube pruning</i>	<i>Transl. time (s) with default params.</i>	<i>BLEU with default params.</i>
#1	195	60.42	257	60.65
#2	186	59.59	4745	60.12
#3	175	55.68	4129	56.12
#4	281	55.50	3994	56.18
#5	221	55.45	4104	56.20
#6	244	55.16	5016	55.98
#7	108	60.74	143	60.90
#8	144	58.50	254	58.61
#9	136	58.50	249	58.61

Table 6. RO→EN: $S2(SI(T_{RO}))$

pruning search (because the chosen model has just phrase translation and no generation component), but yields approximately 0.14 BLEU point increase.

Table 7 shows that for the EN→RO direction, the S2 system models #7 and #8 have a similar performance, increasing the BLEU score from the original 53.94 points to 54.44 (0.5 BLEU point net increase). As with the RO→EN direction, the S2 models that employ generation steps actually slightly decrease the score.

<i>Model</i>	<i>Transl. time (s)</i>	<i>BLEU with cube pruning</i>	<i>Transl. time (s) with default params.</i>	<i>BLEU with default params.</i>
#1	254	54.41	154	54.42
#2	1443	52.14	556	52.55
#3	1051	53.50	594	53.50
#4	543	53.59	798	53.59
#5	530	53.59	613	53.59
#6	805	53.56	997	53.56
#7	282	54.43	167	54.44
#8	417	54.41	287	54.44
#9	403	54.40	280	54.42

Table 7. EN→RO: $S2(SI(T_{EN}))$

6 Evaluation procedure and discussion

After the original corpus was annotated and cleaned, it was split into two separate files for each language: training set and test set. The test file $T_{EN-T_{RO}}$ contains 1200 aligned sentences. Since the sentences were extracted from the randomized corpus after cleaning, the test files contain sentences from all genres that make up the original corpus, so they represent **in-domain** data.

In Tables 6 and 7 we showed that the cascaded factored SMT (S1+S2) performs better than the baseline system (S1) for both translation directions, in terms of BLEU scores. We were inter-

ested to see which were the most distant translations from the reference, assuming that these were bad translations. We computed for each sentence I the similarity scores SIM between its translations and the reference translation. These scores were computed with the same BLEU-4 function used for bitexts. Similarly to the BLEU score applied to a bitext, 100 means perfect match and 0 means complete mismatch. Thus, we obtained 1200 pairs of scores SIM_{S1}^I and SIM_{S1+S2}^I . We also compute the average similarity scores as $\frac{1}{1200} \sum_{I=1}^{1200} SIM_{S\alpha}^I$ where S_α is S1 or S1+S2. As expected, the average SIM scores make the same ranking as the BLEU scores, although they are a bit higher (ex: 61.18 for S1 and 63.58 for S1+S2 for the RO→EN direction).

We briefly comment on the results of this analysis for the Romanian-English translation direction. We manually analysed the test set translations. We identified 3 sentences with their translations having a zero SIM score for both systems. The explanation was that the reference translation was wrongly aligned to the source sentence.

S1 produced 72 perfect translations (score 100) while S1+S2 produced 105. Only 57 perfect translations were common to S1 and S1+S2, meaning that S1+S2 actually deteriorated a few of the original perfect translations. By analyzing the 15 translations that were “deteriorated” we noticed that they were identical, except that unlike S1+S2, S1 and Reference translations either had a differently capitalized letter that marginally lowered the score or had multiword units joined by underscores (e.g. *as well as* vs. *as_well_as*). This was a small bug which has been removed and which, overall, brought a 0.05 increase in the BLEU score. One of the “degraded” translation pair is given below:

RO: *după examinarea problemelor și consecințelor posibile , Uniunea Democrată Croată a Primului Ministru Ivo Sanader și aliații săi parlamentari au decis să sprijine amânarea .*

S1: *after examination problems and possible consequences , the Democratic Union of Croatian Prime Minister Ivo Sanader and his allies lawmakers decided to support the postponement .* (score 0.1794)

S1+S2: *after examination problems and possible consequences , the Croatian Democratic Union of Prime Minister Ivo Sanader and his allies lawmakers decided to support the postponement .* (score 0.1695)

EN_{REF}: *after considering possible issues and consequences , Prime Minister Ivo Sanader 's Croatian Democratic Union and its parliamentary allies decided to support a delay . "*

If one ignores the underscore issue in the S1+S2 translation, then this translation is better than the one of S1. A frequent translation difference with respect to the reference translations is illustrated by the example above: the Saxon genitive construction for noun phrases is replaced by a prepositional genitival construction (in this case the word order is closer to the Romanian word order).

The capitalization and punctuation are other sources of lower scoring against the reference. All these examples show the sensitivity of the BLEU scoring method, especially for very short sentences.

Another important variable to note is the amount of change from one layer to the other: out of all sentences, around 37% had a BLEU increase while around 20% had a BLEU decrease (but see the comment on the underscore difference), the rest 43% have not been changed in any way.

Overall, we obtain a 3.89 BLEU point increase for the RO→EN direction and a smaller 0.5 BLEU point increase for the more difficult EN→RO direction using our cascaded system.

Another interesting result was to evaluate the simple cascading systems without feature models, that is (S1=t0-0m0)+(S2=t0-0m0) and compare their performances with the direct translations and the best feature-models cascaded systems. The results are shown in Table 8.

RO → EN' → EN		EN → RO' → RO	
Model #	BLEU	Model #	BLEU
#1+#1	60.47	#1+#1	54.29
#3+#7	60.90	#2+#7	54.44

Table 8. $S_2(SI(T_{source}))$

The increased accuracy due to various feature combinations versus the baseline system has been apparent from Tables 6 and 7 compared to the results in Table 3. Table 8 shows that the direct translations (S1 with any model) for both directions have BLEU scores lower than the cascaded system (S1+S2) even when feature models were not used (model #1+#1).

Thus, we can support the statement that the morphological features and the cascading idea are beneficial to the overall accuracy of translations (at least between Romanian and English).

<i>S1 SIM</i>	<i>S2 SIM</i>	<i>Romanian Source</i>	<i>S1 Translation</i>	<i>S2 Translation</i>	<i>English Reference</i>
<i>Difference</i>					
0.397		bun , și-acuma să revenim	good , and now <i>to revenim</i>	good , and now <i>let us go to</i>	and now let us get back to
0.492		la problema lui cum și de	to the problem of how and	the problem of how and	the question of how and
0.095		ce.	why.	why.	why.
0.392		spune-mi ce crezi tu că-ți	tell me what <i>believe you</i>	tell me what <i>you think</i> that	tell me what you think you
0.660		amintești.	that you remember.	you remember.	remember.
0.268					
0.213		În primul rând , pentru că	firstly , because confes-	firstly , because <i>the</i> con-	in the first place , because
0.316		mărturisirile pe care le	sions <i>on which</i> they made	fessions they made were	the confessions that they
0.104		faceau erau evident	were obviously clean and	obviously clean and jerk	had made were obviously
		smulse și neadevărate.	jerk and untrue.	and untrue.	extorted and untrue.
0.447					
0.376		cum ar putea muri ?	how <i>could die</i> ?	how <i>to die</i> ?	how could he die ?
-0.071					
0.256		cei trei nu făcuseră nici o	the three not <i>făcuseră</i> any	the three not <i>to make</i> any	the three men never
0.216		mişcare.	movement.	movement.	stirred.
-0.039					

Table 9. Out-of-domain text S1 / S1+S2 translation improvement / degradation examples for RO→EN

Given the corpus is almost entirely composed of juridical and medical texts, we were anxious to see how the second translation step would perform on **out-of-domain** texts.

To make things even harder, we chose a different genre: literary fiction. We extracted 1000 sentences between 3 and 40 words long from Orwell’s “1984” novel. This test text is challenging because it contains many out of vocabulary words, new senses, frequent subject-elided constructions (Romanian is a pro-drop language), verbal tenses specific to literary narratives which are practically absent from the training data. Another challenge was due to the Romanian translation of Orwell’s original, which is not a word-for-word translation, but a literary one.

We tested only the RO→EN direction with the following results: the first translation system (S1) obtained a score of 27.53 BLEU points (model #3), while the second system (S2) marginally improved the translation to 27.70.

Out of the 1000 sentences, 69 have had their scores properly increased and 76 slightly “decreased”. However, even if the overall BLEU score increase was minimal, we observed that the translation quality has improved from a human analysis point of view. The positive and negative examples (Table 9) show that even though the changes in SIM score are minimal, the text produced by S2 corrects some of the unknown words of S1 (by synonyms or paraphrases, not matching the reference) as well as phrase structure by better word choice and word reordering (corrections missed by the BLEU/SIM scores).

Finally, we took the cascading idea one step further by repeating the entire train-translate

process (step 2), obtaining $S3(S2(S1(T_{source})))$. We observed that the translation stabilized, with very few sentences being changed (around 1%), and with the changes being minor (increasing or even decreasing the BLEU score by less than ~0.05 points). We concluded that further cascading would not bring significant improvements.

7 Conclusions and future work

This article presented a simple but effective way of further improving the quality of a phrased-based statistical machine translation system, by cascading translators. We are not aware of better translation scores for the Romanian-English pair of languages. The idea of post-processing the output of a SMT system is not new but, this step was most often than not based on hand-crafted rules or other knowledge intensive methods. A similar idea was recently reported in (Ehara, 2011) but, their EIWA ensemble is based on a commercial rule-based MT (specialized in patent translation) for the first step and a MOSES-based SMT for the second phase (named statistical post-editing). There are several other methodological differences between our system and the one described in (Ehara, 2011). EIWA does not work in real time because before proper translation of a text T, the SMT post-editor is trained on a text similar to T. The similar text is constructed from a large patent parallel corpus (3,186,284 sentence pairs) by selecting for each sentence in T an average number of 127 similar sentences.

We use the same SMT system trained on different parallel data. The first system S1, trained

on parallel data $\{C_A, C_B\}$ learnt to produce draft translations from L_A to L_B . The second translation system S2, trained on the “parallel” data $\{S1(C_A)-C_B\}$, learnt how to improve the draft translations. Except for the training data and the different parameter settings, the two systems are incarnations of the same basic system. Contrary to Ehara (2011), we found that setting the distortion parameter to a non-null value improves the translation quality. Translation of a new, unseen text is achieved in real time (no retraining at the translation time). While in (Ehara, 2011) improvements were reported for two language pairs (Japanese to English and Chinese to English), we showed that our approach, for the present moment, works only for one language pair (Romanian and English) but in both translation directions. We also showed that the cascaded approach improves the translation quality for both in-domain and out-of-domain texts, although not to the same degree.

As future research, we are considering extending the factored experiment with comparable parallel data. The comparable data is available through the ACCURAT project. The aim of the ACCURAT project, to be finalized in June this year, is to research methods and techniques to overcome one of the central problems of machine translation (MT) – the lack of linguistic resources for under-resourced areas of machine translation. Within this context various narrow domain adaptation techniques will be evaluated and experiments will be conducted for several other language pairs.

Acknowledgments. This work has been supported by the ACCURAT project (www.accurat-project.eu/) funded by the European Community’s Seventh Framework Program (FP7/2007-2013) under the Grant Agreement n° 248347.

References

Avramidis E., Koehn, P. 2008. Enriching morphologically poor languages for statistical machine translation. In: *Proceedings of Association for Computational Linguistics / HLT*, pp. 763–770, Columbus, Ohio

Ceaușu Alexandru. 2006. Maximum Entropy Tiered Tagging, Janneke Huitink & Sophia Katrenko (eds), *Proceedings of the Eleventh ESSLLI Student Session*, ESSLLI 2006, pp. 173-179

Ceaușu, A., Tufiș, D. 2011. Addressing SMT Data Sparseness when Translating into Morphologically-Rich Languages. In Bernadette Sharp, Mi-

chael Zock, Michael Carl, and Arnt Lykke Jakobsen (eds.) *Human-machine interaction in translation*, Copenhagen Business School, pp. 57-68.

Ehara T. 2011. Machine translation system for patent documents combining rule-based translation and statistical postediting applied to the PatentMT Task, *Proceedings of NTCIR-9 Workshop Meeting*, December 6-9, 2011, Tokyo, Japan, pp. 623-628.

Erjavec, T., Monachini, M. (Eds.). 1997. *Specifications and Notation for Lexicon Encoding*. Deliverable D1.1 F. Multext-East Project COP-106. <http://nl.ijs.si/ME/CD/docs/mte-d11f/>

Habash, N., Dorr, B., Monz, C. 2006. Challenges in Building an Arabic-English GHMT System with SMT Components. In *Proceedings of AMTA’06*, Cambridge, MA, USA.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, demonstration session, Prague.

Koehn, P., Hoang, H. 2007. Factored Translation Models. In: *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 868–876, Prague.

Papineni, K., Roukos, S., Ward, T., Zhu W.J. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 20th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, pp. 311-318.

Tiedemann, J. 2009. News from OPUS - *A Collection of Multilingual Parallel Corpora with Tools and Interfaces*. In: N. Nicolov and K. Bontcheva and G. Angelova and R. Mitkov (eds.) *Recent Advances in Natural Language Processing (vol V)*, pp. 237-248.

Tufiș, D. 1999. Tiered Tagging and Combined Classifiers. In: F. Jelinek, E. Nth (eds) *Text, Speech and Dialogue LNCS vol. 1692*, pp. 28-33 Springer-Verlag Berlin Heidelberg.

Tufiș, D., Ceașu, A. 2008. DIAC+: A Professional Diacritics Recovering System, in *Proceedings of LREC 2008*, May 26 - June 1, Marrakech, Morocco. ELRA - European Language Resources Association.

Tufiș, D., Ion, R., Ceașu, A., Ștefănescu, D. 2008. RACAI’s Linguistic Web Services, in *Proceedings of LREC 2008*, May 26 - June 1, Marrakech, Morocco. ELRA - European Language Resources Association.

Hybrid Parallel Sentence Mining from Comparable Corpora

Dan Ștefănescu

RACAI

Calea 13 Septembrie, 13
Bucharest, Romania

danstef@racai.ro

Radu Ion

RACAI

Calea 13 Septembrie, 13
Bucharest, Romania

radu@racai.ro

Sabine Hunsicker

DFKI

Stuhlsatzenhausweg 3, 66123
Saarbrücken, Germany

sabine.hunsicker@dfki.de

Abstract

This paper presents a fast and accurate parallel sentence mining algorithm for comparable corpora called LEXACC based on the Cross-Language Information Retrieval framework combined with a trainable translation similarity measure that detects pairs of parallel and quasi-parallel sentences. LEXACC obtains state-of-the-art results in comparison with established approaches.

1 Introduction

Mining for parallel sentences in comparable corpora is much more difficult than aligning sentences in parallel corpora. Sentence alignment in parallel corpora usually exploits simple empirical evidence (turned into assumptions) such as (i) the length of a sentence is proportional with the length of its translation and (ii) the discourse flow is necessarily the same in both parts of the bi-text (Gale and Church, 1993). Thus, the extraction tools search for parallel sentences around the same (relative) text positions, making sentence alignment a much easier task when compared to kind of work undertaken here.

For comparable corpora, the second assumption does not hold. Parallel sentences, should they exist at all, are scattered all around the source and target documents, and so, any two sentences¹ have to be processed in order to determine if they are parallel or not. Also, we aim at finding pairs of quasi-parallel sentences that are not entirely parallel but contain spans of contiguous text that is parallel. Thus, finding parallel sentences in comparable corpora is confronted

with the vast search space one has to consider since any positional clues indicating parallel or partially parallel sentences are not available.

The brute force approach is to analyze every element of the Cartesian product between the two sets containing sentences in the source and target languages. This approach is clearly impractical since the resulting algorithm would be very slow and/or memory consuming.² To reduce the search space, we turned to a framework that belongs to Information Retrieval: Cross-Language Information Retrieval (CLIR). The idea is simple: use a search engine to find sentences in the target corpus that are the most probable translations of a given sentence from the source corpus. The first step is to consider the target sentences as documents and index them. Then, for each sentence in the source corpus, one selects the content words and translates them into the target language according to a given dictionary. The translations are used to form a Boolean query which is then fed to the search engine. The top hits are considered to be translation candidates.

Using the CLIR approach to select a set of candidate target sentences (out of all target sentences) for the input source sentence is one way to dramatically reduce the search space. The reduced search space will serve another practical concern: the execution time. Thus, each candidate target sentence can be compared with the input sentence using a computationally much more complex translation similarity measure that would otherwise require an unacceptable amount of time to finish analyzing all possible pairs.

In what follows, we present our own adaptation of the hybrid CLIR/translation similarity measure approach to parallel sentence mining from comparable corpora called “Lucene-based Parallel Sentence Extraction from Comparable Corpora” (LEXACC). We describe the indexing

¹ Or a carefully selected set of sentence pairs as we will see in the next sections.

² With the possible exception of the parallelizing the computations but this issue is beyond the scope of this paper.

of the target corpus in subsection 3.1, the Boolean query generation for the input sentence in subsection 3.2, an additional filtering step on the output of the Lucene search engine in subsection 3.3 and our design of the translation similarity measure in section 4. We present a host of experiments aimed at assessing the performance of LEXACC from both the CLIR perspective (precision, recall and F1-measure) and practical SMT experimenting with data produced by LEXACC.

2 Related Work

Parallel data mining from comparable corpora receives its share of attention from the statistical machine translation scientific community, one of the major reasons being the fact that the Web can be seen as a vast source of comparable corpora.

The CLIR approach to finding translation candidates for sentences (reducing the search space) has received significant attention. While Rauf and Schwenk (2011) index the target sentences directly, Munteanu and Marcu (2005) index target documents for retrieving similar ones.

Another approach to cutting the search space is to perform document alignment inside the comparable corpus first and then to attempt extracting parallel sentences by inspecting the constructed document pairs only. This road has been taken by Fung and Cheung (2004) who perform document alignment using a simple dictionary-based, translation similarity measure. Recently, Ion (2011) proposes an EM algorithm that finds document alignments in a comparable corpus.

The way a pair of sentences is deemed parallel or not is usually specified with three different approaches: binary classifiers (Munteanu and Marcu, 2005; Tillman, 2009), translation similarity measures (Fung and Cheung, 2004) and generative models (Quirk et al., 2007). Our approach is somewhat similar to that of Munteanu and Marcu (2005) who used a dictionary to translate some of the words of the source sentence, and then used these translations to query a database for finding matching translation candidates. The difference resides in the fact that they choose candidate sentences based on word overlap and the decision whether a sentence pair is parallel or not is performed by a Maximum Entropy classifier trained on parallel sentences. With respect to Rauf and Schwenk (2011) who also index target sentences, our approach benefits of some filtering steps, the query is formulated using additional fields and we use a much more elaborated translation similarity measure.

3 Indexing, Searching and Filtering

3.1 Indexing target sentences

Our goal is to implement a simple yet effective solution, easily replicable. First, we split the target corpus into sentences and transform them so that we keep only stemmed non-functional words.³ We also compute the average length in words (μ) and the standard deviation (σ) for target sentences. We consider a sentence s to be short if $length(s) \leq \mu + \sigma$ and long if $length(s) \geq \mu - \sigma$. We consider the medium-sized sentences for which $\mu - \sigma \leq length(s) \leq \mu + \sigma$, to be both short and long.

Following the general description presented in the introduction, we use the C# implementation of Lucene⁴ to index the target sentences as Lucene documents. For each such document, we introduce three additional searchable fields, two of them corresponding to the sentence length:

- (i) a field specifying if the sentence is *small*;
- (ii) a field specifying if the sentence is *long*;
- (iii) a field specifying the document where the target sentence belongs; this field is based on the document alignment information of the comparable corpus being processed and it is optional if such alignment information is not supplied.

3.2 Finding translation candidates for source sentences

Given an input source sentence (out of the total S source sentences), the role of the search engine is to return a list of translation candidates that are to be further analyzed. The number of hits h we take into account regulates the size of the new search space: $h * S$. The larger it is, the higher the number of candidates which can potentially increase the recall but also the computational complexity. For each sentence in the source corpus, we generate a Lucene query as follows:

(i) We employ a GIZA++ (Och and Ney, 2000) dictionary previously created from existing parallel documents. This dictionary is expected to be small due to the lack of necessary resources. For each content word we keep the best 50 translation equivalents, which are also content words, having translation probabilities above 0.1. Each of them is stemmed and added as an disjunctive query term (SHOULD occur);

(ii) We add two disjunctive query terms (SHOULD occur) standing for the length of the source sentence: *short* and *long*. Each of these

³ We keep functional words lists for all languages.

⁴ <http://incubator.apache.org/lucene.net/download.html>

terms can be boosted according to the importance one wants to give to matching source and target lengths. In our implementation, the value of the boosting factor is 2;

(iii) We add a compulsory query term (MUST occur) specifying the target document where the source sentence translation should be searched. However, this term can be added only if the document alignment information exists and it has been used at index creation as well.

After the query is constructed, we use it to interrogate the default Lucene search engine (no modifications on the relevance method) in order to get the best h hits.

3.3 Filtering

The filtering step is designed to further reduce the new search space, selecting only the best candidates for the final stage in which the translation similarity measure (Section 4) is applied. Filtering must be very fast and good enough not to filter out parallel data. We do this by computing a viability score for each candidate sentence pair and then keeping only those above the average. For a candidate pair formed by a source sentence s and a target sentence t , the formula is:

$$viabilityScore = \alpha * \beta * se * sim \quad (1)$$

where se represents the score returned by the search engine and sim is a similarity score we will come back to later. The other factors are aiming at favoring high scores for sentences with similar (α) and large (β) lengths. In our implementations they are computed as:

$$\alpha = 1 - \frac{abs(|s| - |t|)}{\max(|s|, |t|)} \quad (2)$$

$$\beta = \frac{\min(|s|, |t|)}{\lambda} \quad (3)$$

where abs is the absolute value, $|s|$ is the length in words of sentence s and λ is an integer constant representing the length threshold from which we consider a sentence to be very long ($\lambda=100$ in our implementation, but it can be chosen depending on the given corpora).

The similarity score (sim) from equation 1 is calculated according to the formula:

$$sim = \frac{2 * teFound * te}{|s| + |t|} * \frac{1}{\sqrt{coh}} \quad (4)$$

where $teFound$ is the total number of words in s for which we found translation equivalents in t , coh is the *cohesion score* computed as the average distance between the sorted positions of the

translation equivalents found in t (the lower the better)⁵ and te is calculated as:

$$te(s, t) = \sum_{w_s \in S} \max_{w_t \in T} dicScore(w_s, w_t) \quad (5)$$

where $dicScore$ is the translation probability score from the dictionary. The rationale behind equation 5 is induced by the assumption that a word w_s is translated by only one word w_t and so, $dicScore(w_s, w_t) \geq dicScore(w_s, w_i)$ for any w_i in t .

We should note that since we aim at gathering parallel data which is not already in the dictionary with started with, we are more interested in finding long parallel texts. It is more probable that such texts would contain (beside already known translations) unknown parallel data.

4 The Translation Similarity Measure

The binary classifier of Munteanu and Marcu (2005) associate a confidence probability with its decision but setting this confidence at 0.5 or 0.7 as they do, is equivalent to saying that sentence pairs with a score below the confidence level are not interesting for SMT.⁶ Our view is that whatever sentence pairs actually improve the output of an SMT system are important and we found that these range from parallel, quasi-parallel to strongly comparable.

We modeled our translation similarity measure as a weighted sum of feature functions that indicate if the source piece of text is translated by the target. Given two sentences s in the source language and t in the target language, then the translation similarity measure $P(s, t)$ is

$$P(s, t) = \sum_i \theta_i f_i(s, t) \quad (6)$$

such that $\sum_i \theta_i = 1$. Each feature function $f_i(s, t)$ will return a real value between 0 (s and t are not related at all) and 1 (t is a translation of s) and contributes to the overall parallelism score with a specific fraction θ_i that is language-pair dependent and that will be automatically determined by training a logistic regression classifier on existing parallel data (see next subsection).

Each of the feature functions $f_i(s, t)$ has been designed to return a value close to 1 on parallel s and t by manually inspecting a fair amount of parallel examples in the English-Romanian pair of languages. By negation, we assume that the

⁵ We experimented with different power values for the cohesion score. For $\frac{1}{2}$ (the square root) we had the best results.

⁶ But we acknowledge the fact that the probability of a sentence pair being parallel as computed by the classifier of Munteanu and Marcu is a proper model of parallelism

same feature functions will return a value close to 0 for non-parallel, not-related s and t but this behavior is critically influenced by the quality and completeness of the linguistic computational resources that we use: bilingual translation lexicons, lists of inflectional suffixes used for stemming and lists of stop-words. Thus, generally, a feature function that uses one (or more) of the resources mentioned above can falsely return a value close to 0 for parallel s and t due to the fact that this decision was made in the absence of the relevant entries in that resource. The prototypical example here is that the translation lexicon does not contain the relevant translations for the words in s .

4.1 Features

Before being processed, sentences s and t are tokenized, functional words are identified and content word are stemmed using language-dependent inflectional suffixes. Given these transformations of s and t , all features $f_i(s, t)$ are language-independent. We use 5 features.

$f_1(s, t)$ is the “**content words translation strength**” feature. Given a statistical translation dictionary obtained by e.g. applying GIZA++ on a parallel corpus,⁷ we find the best 1:1 alignment A between content words in s and t such that the translation probability⁸ is maximized. If $\langle cw_i^s, cw_j^t \rangle$ is a word pair from A , $p(\langle cw_i^s, cw_j^t \rangle)$ is the translation probability of the word pair from the dictionary and $|s|$ is the length (in content words) of sentence s , then

$$f_1(s, t) = \frac{\sum_{\langle cw_i^s, cw_j^t \rangle \in A} p(\langle cw_i^s, cw_j^t \rangle)}{|s|} \quad (7)$$

This feature has a maximum value of 1 if all content words from s are translated in t with the maximum probability of 1.

$f_2(s, t)$ is the “**functional words translation strength**” feature. The intuition is that functional words around content words aligned as in $f_1(s, t)$, will also align for parallel s and t because of the fact that, from a dependency-syntactic point of view, functional words (prepositions, determiners, articles, particles, etc.) are

usually governed by or govern nearby content words. Mathematically, if $\langle fw_k^s, fw_l^t \rangle$ is the highest scored pair of aligned functional words near (in a window of ± 3 words) the aligned pair of content words $\langle cw_i^s, cw_j^t \rangle$ from A , $|A|$ is the cardinal of the best alignment as found by $f_1(s, t)$ and $p(\langle fw_k^s, fw_l^t \rangle)$ is the probability of the functional word pair from the dictionary, then

$$f_2(s, t) = \frac{\sum_{\langle cw_i^s, cw_j^t \rangle \in A} p(\langle fw_k^s, fw_l^t \rangle)}{|A|} \quad (8)$$

The maximal value of $f_2(s, t)$ is 1 and it is reached when for each aligned pair of content words from A , there is a pair of functional words that align with the maximum probability of 1.

$f_3(s, t)$ is the “**alignment obliqueness**” feature (Tufiş et al., 2006). Here we have redefined it to be a discounted correlation measure because there are pairs of languages for which the natural word order implies crossing word alignment links. $f_3(s, t)$ also uses the alignment set A of content words described for feature $f_1(s, t)$ from which we derive two source and target vectors x^s and x^t of the same length containing the indices i in the ascending order ($1 \leq i \leq |s|$) and j respectively ($1 \leq j \leq |t|$) of content words cw_i^s and cw_j^t that form an alignment pair in A . Alignment obliqueness is computed as

$$f_3(s, t) = \text{abs}(\rho_{x^s, x^t}) \frac{1}{1 + e^{-10 \frac{|A|}{\min(|s|, |t|)} + 5}} \quad (9)$$

where ρ_{x^s, x^t} is the Pearson correlation coefficient of the x^s and x^t vectors and $\text{abs}(x)$ is the absolute value function. The second term is a modified sigmoid function $f(x) = \frac{1}{1 + e^{-10x + 5}}$ designed to be a discount factor with values between 0 and 1 when x takes on values between 0 and 1. The rather steep variation of $f(x)$ was experimentally modeled in order to heavily discount “rare” alignments for which the Pearson correlation is high. Thus, if A contains only a few alignments relative to $\min(|s|, |t|)$ (the size of A is at most $\min(|s|, |t|)$), then even if ρ_{x^s, x^t} is high, $f_3(s, t)$ should be small because a few alignments usually do not indicate parallelism.

$f_4(s, t)$ is the “**strong translation sentinels**” feature. Intuitively, if sentences s and t are parallel then, frequently (at least in our studied examples), one can find content words that align near the beginning and end of the considered sentences. $f_4(s, t)$ is a binary-valued feature which is 1 if we can find “strong” translation pairs (probability greater than 0.2; set experimentally) between the first 2 content words at the beginning

⁷ To obtain the dictionaries mentioned throughout this section, we have applied GIZA++ on the JRC Acquis corpus (<http://langtech.jrc.it/JRC-Acquis.html>).

⁸ For two source and target words, if the pair is not in the dictionary, we use a 0 to 1 normalized version of the Levenshtein distance in order to assign a “translation probability” based on string similarity alone. If the source and target words are similar above a certain threshold (experimentally set to 0.7), we consider them to be translations.

of s and t and between the last 2 content words at the end of s and t . $f_4(s, t)$ is 0 otherwise.

Finally, $f_5(s, t)$ is the “**end with the same punctuation**” feature. This is also a binary-valued feature which is 1 if both s and t end with the same type of punctuation: period, exclamation mark, etc. It is also 1 if both s and t lack final punctuation. $f_5(s, t)$ is 0 otherwise.

The observant reader has noticed by now that all the features with the exception of $f_5(s, t)$ are not symmetrical because they all depend on the alignment A computed for $f_1(s, t)$ which is not symmetrical and as such, the measure from equation 6 is not symmetrical as well. In order to have evidence from both directions, we will use the arithmetic mean to get the final measure:

$$M(s, t) = M(t, s) = \frac{P(s, t) + P(t, s)}{2} \quad (10)$$

4.2 Learning the optimal weights

The weights θ_2 and θ_3 corresponding to the features “functional words translation strength” and “alignment obliqueness” are language-pair dependent because of the specific word ordering of the source and target languages. At the same time, θ_1 through θ_4 have to be optimized with respect to the translation lexicon in use, since the construction of the word alignments is based on this dictionary. Also, since $P(s, t)$ is not symmetrical, we will have to learn different θ_i weights from source to target and vice versa.

In order to derive a set of optimal weights for each language pair and translation lexicon, we have trained a standard logistic regression classifier. Briefly, the logistic regression classifier learns the θ_i weights that define the hyperplane (whose equation is the same as equation 6) that best separates the positive training examples from the negative ones. In our case, the examples are the multidimensional points whose coordinates are given by the feature functions $f_i(s, t)$.

For each language pair, the training set consists of 9500 parallel sentences⁹ for the positive examples and 9500 of non-parallel sentences (obtained from the parallel pairs by random shuffling) for the negative examples. For the training set in question, we also have 500 additional parallel sentences together with 500 non-parallel sentences (obtained by random shuffling as well) as the test set. An example¹⁰ is obtained by com-

puting all the feature functions $f_i(s, t)$ for the given positive (parallel) or negative (non-parallel) s and t .

Table 1 summarizes the derived optimal weights for 8 language-pairs, in both directions. In every pair, one language is English (en) and the others are: Croatian (hr), Estonian (et), German (de), Greek (el), Lithuanian (lt), Latvian (lv), Romanian (ro) and Slovene (sl).

Lang.	θ_1	θ_2	θ_3	θ_4	θ_5	F1/BL
en-ro	0.31	0.02	0.37	0.21	0.09	0.93/0.88
ro-en	0.31	0.01	0.37	0.20	0.11	0.93/0.91
en-de	0.31	0.02	0.3	0.17	0.2	0.94/0.89
de-en	0.35	0.02	0.28	0.16	0.19	0.96/0.92
en-sl	0.23	0.01	0.38	0.2	0.18	0.96/0.89
sl-en	0.2	0.03	0.38	0.19	0.2	0.94/0.89
en-el	0.61	0.08	0.21	0	0.1	0.99/0.98
el-en	0.47	0.08	0.28	0.07	0.1	0.98/0.98
en-lv	0.27	0.05	0.41	0.16	0.1	0.98/0.96
lv-en	0.49	0.03	0.41	0	0.07	0.99/0.96
en-lt	0.33	0.01	0.41	0.15	0.1	0.96/0.91
lt-en	0.28	0.01	0.41	0.15	0.15	0.94/0.90
en-et	0.28	0.08	0.36	0.17	0.11	0.98/0.96
et-en	0.27	0.07	0.38	0.18	0.1	0.96/0.93
en-hr	0.29	0.01	0.41	0.16	0.13	0.98/0.95
hr-en	0.25	0.02	0.44	0.17	0.12	0.98/0.97

Table 1: Optimal weights for the translation similarity measure

The column named “F1/BL” (see Table 1) indicates the gain in F1 measure when testing the translation similarity measure with the optimal weights on the test set as compared to a baseline (BL) consisting of applying the measure using fixed values of the weights corresponding to our intuition of their importance: $\theta_1 = 0.45$, $\theta_2 = 0.2$, $\theta_3 = 0.15$, $\theta_4 = 0.15$, $\theta_5 = 0.05$. For instance, we imagined that the content words translation strength feature $f_1(s, t)$ is much more important compared to the rest of the features but the training procedure proved us wrong.

5 Experiments and Results

5.1 Experiment Setting

We evaluated our approach on 7 pairs of languages under the framework of the ACCURAT project.¹¹ For each pair, the source language is English (en), while the target languages are: Estonian (et), German (de), Greek (el), Lithuanian (lt), Latvian (lv), Romanian (ro) and Slovene (sl). In order to compute precision and recall when mining for parallel sentences, we have devised

⁹ Mostly from the News domain for all language pairs.

¹⁰ When an example occurs multiple times with both labels, we retain all the occurrences of the example with the most frequent label and remove all the conflicting occurrences.

¹¹ <http://www accurat-project.eu/>

artificial comparable corpora for all mentioned language pairs, with different levels of controlled comparability. Starting from 100 news parallel sentences for all language pairs, the corpora were created by injecting noise (in specific proportions) extracted from the News corpora collected in the ACCURAT project. We experimented with 4 different amounts of noise: 2:1,¹² 5:1, 10:1, 100:1, corresponding to different degrees of comparability, from strongly comparable to weakly comparable. The worst case scenario is by far the one with 100:1 noise and so, most of our experiments were developed under this setting.

We evaluated the efficiency of LEXACC after each of its steps: (i) the extraction of translation pair candidates using the search engine, (ii) candidate pairs filtering and (iii) the usage of the translation similarity measure. Moreover, we evaluated the impact of the extracted data when used for improving SMT translation models.

5.2 Search Engine Efficiency

To measure the efficiency of using the search engine for finding translation candidates in the worst case scenario (100:1 noise ratio), we computed the recall we would obtain if we would have kept the best 100 hits (target sentences) returned by the engine for each source sentence. Instead of brute force analyzing 10,100² sentence pairs, we can now look at only 1 million pairs. This means a search space reduction of about 100 times. Table 2 shows that this approach is effective for most of the language pairs, but poor for en-el and en-ro. One of the reasons might be the quality of the dictionaries we relied on when generating the search engine queries.

Pair	Recall UB	Data Size (pairs / disk size)
en-de	0.98	1,009,500 / 323 Mb
en-el	0.42	1,009,700 / 485 Mb
en-et	0.89	1,008,800 / 345 Mb
en-it	0.93	1,008,200 / 350 Mb
en-lv	0.92	1,008,300 / 366 Mb
en-ro	0.69	1,009,800 / 294 Mb
en-sl	0.80	688,266 / 191 Mb

Table 2: Recall upper boundary (UB) and size (sentence pairs and disk space occupied) for the translation candidates returned by Lucene

5.3 Filtering Efficiency

As already mentioned, filtering is an intermediary step designed to further reduce the search space used for the final analysis. The filtering

module receives high scores for speed and search space reduction for all language pairs. However, in terms of preserving the recall upper boundary, it performs well only for en-lv and en-de and acceptable for en-ro and en-el. It loses about 40% recall for the other 3 language pairs. Table 3 summarizes the results.

Pair	Recall UB	Recall Loss	Size (pairs / disk size)	Search Space Drop
en-de	0.83	15.30%	20,868 / 10 Mb	97.93%
en-el	0.30	28.57%	108,629/69 Mb	89.24%
en-et	0.54	39.32%	34,051 / 22 Mb	96.62%
en-it	0.57	38.70%	35,831 / 21 Mb	96.44%
en-lv	0.83	9.78%	91,305 / 45 Mb	90.94%
en-ro	0.53	23.18%	160,968/67 Mb	84.05%
en-sl	0.44	45%	65,191 / 28 Mb	90.52%

Table 3: Recall upper boundary and size after the filtering step

5.4 Translation Similarity Efficiency

We evaluated the efficiency of the Translation Similarity Measure (TSM) from Section 4 by comparing it with the MaxEnt classifier by Munteanu and Marcu (2005) on English-German (en-de) document pairs with different levels of comparability (2:1 noise ratio, 5:1 and 10:1; see section 5.1). For both TSM and MaxEnt (with the associated confidence score for the “parallel” label), we took into account all possible thresholds with a granularity of 0.01 above which the candidate pairs are considered parallel. We report the results corresponding to the threshold that maximizes F1 for TSM and F1 for MaxEnt (threshold are not the same). We explored 3 possible scenarios. The first one (Table 4) is to compute TSM for all possible sentence pairs.

	2:1		5:1		10:1	
	ME	TSM	ME	TSM	ME	TSM
P	0.800	0.791	0.789	0.760	0.523	0.724
R	0.560	0.760	0.450	0.700	0.450	0.630
F1	0.658	0.775	0.573	0.729	0.483	0.673

Table 4: en-de comparison between the MaxEnt classifier (ME) and the TSM when applied individually onto all possible sentence pairs

The second scenario (Table 5) is to compute TSM only for the candidate pairs proposed by the search engine, without filtering.

	2:1		5:1		10:1	
	ME	LEX	ME	LEX	ME	LEX
P	0.800	0.717	0.789	0.650	0.523	0.618
R	0.560	0.710	0.450	0.650	0.450	0.600
F1	0.658	0.713	0.573	0.650	0.483	0.609

Table 5: en-de comparison between the MaxEnt classifier and LEXACC with no filtering

¹² For each parallel sentence, 2 noise sentences were added

The third scenario is similar to the second one, only this time we use filtering.

	2:1		5:1		10:1	
	ME	LEX	ME	LEX	ME	LEX
P	0.800	0.809	0.789	0.737	0.523	0.742
R	0.560	0.340	0.450	0.450	0.450	0.520
F1	0.658	0.478	0.573	0.559	0.483	0.611

Table 6: en-de comparison between the MaxEnt classifier and LEXACC with filtering

For strongly comparable corpora (with less noise, like the 2:1 corpus) the filtering step is in fact worsening the results. This is something to be expected because the filtering step eliminates a large proportion of the candidate pairs returned by the engine. Thus, filtering should be used only for weakly comparable corpora.

In order to make things more clear, we performed yet another experiment, this time for 100:1 noise ratio which corresponds to a very weakly comparable corpus. In this setting, taking into account all possible sentence pairs as candidate pairs would result in a huge running time and so, we were able to compare only the results obtained by LEXACC with and without filtering.

	LEXACC NO filtering		LEXACC WITH filtering
	Best	Same T ¹³	Best
P	0.327	0.101	0.800
R	0.370	0.710	0.640
F1	0.347	0.177	0.711
<i>Threshold</i>	<i>0.59</i>	<i>0.41</i>	<i>0.41</i>
<i>Running Time</i>	<i>49.72 minutes</i>		<i>5.53 minutes</i>

Table 7: En-De comparison between LEXACC with and without filtering for 100:1 noise

We can see that for weakly comparable corpora, at the same threshold (0.41), filtering gets rid of a lot of noise, keeping the precision high (compare 0.8 with 0.101) at a modest decrease of the recall (compare 0.64 with 0.71).

Table 8 shows the accuracy of LEXACC when running on the 100:1 noise ratio comparable corpora. The running times depend on the sentence lengths and the size of the dictionaries.

Pair	P	R	F1	Thr.	Minutes
en-de	0.800	0.64	0.711	0.41	5.53
en-el	0.550	0.22	0.314	0.35	27.24
en-et	0.284	0.23	0.254	0.34	7.11
en-it	0.398	0.41	0.403	0.39	8.24
en-iv	0.357	0.50	0.416	0.51	11.75
en-ro	0.473	0.27	0.343	0.65	37.33
en-sl	0.219	0.16	0.185	0.34	7.75

Table 8: LEXACC (with filtering) run on the 100:1 noise ratio comparable corpora

¹³ Same T: results obtained without filtering for the threshold yielding the best results with filtering (0.41).

5.5 SMT Experiments

To test the quality of the data extracted by LEXACC, we ran a few experiments with domain-adapted SMT in the automotive industry domain. We manually created a parallel corpus from an English-German comparable corpus of about 3.5 million sentences per language collected from the Web. The results of the experiments with the LEXACC extracted data were compared to the same experiments conducted with the manually extracted parallel data, to examine and compare the influence of the LEXACC extracted data. Table 9 shows the statistics on the sentence pairs and sentence counts in the parallel and LEXACC extracted data.

Data	#pairs	# unique sent. (de/en)
parallel	44,482	42,396 / 44,290
extracted	45,952	12,718 / 13,306

Table 9: Statistics on parallel and extracted data

We compared three systems in our experiments: the “Baseline” system which was trained only on the Europarl (EP, (Koehn, 2005)) and News Commentary corpus (NC),¹⁴ “Automotive.parallel” which added only the parallel data to the baseline and the “Automotive.extracted” which added only the LEXACC extracted data to the baseline. All resulting corpora were aligned using GIZA++ and the MT systems were trained using the Moses SMT Toolkit (Koehn et al., 2007). The languages models were trained using SRILM (Stolcke, 2002).

The Baseline system only uses Europarl, both for the translation and the language model but for the two adapted systems we used an additional language model trained on the domain-specific texts. Tuning via MERT was performed for all systems on a domain-specific development set; testing also used text from the automotive domain. The translations were evaluated using BLEU (Papineni et al., 2001).

System	BLEU
Baseline	18.81%
Automotive.parallel	30.25%
Automotive.extracted	25.44%

Table 10: BLEU scores

As Table 10 shows, it is possible to gain about 6.5 BLEU points over the baseline system with the extracted data. The parallel data outperforms LEXACC, which may be due to the fact that the parallel data includes more unique sentences (see Table 9). But although only approx. 30% of the available unique data was extracted, an increase

¹⁴ <http://www.statmt.org/wmt11/translation-task.html>

of 6.5 BLEU points is recorded -- more than half of the increase achieved with the full parallel data. This means that LEXACC is able to discover salient parallel data that brings significant gains in BLUE score despite its size.

Another area of interest is how the extracted parallel and strongly comparable data compares to clean parallel data. In the extracted data, every German sentence is linked to 3.5 English sentences on average. To examine the effect of this noise, we retrained “Automotive.parallel” with increasing amounts of data. Table 11 shows that the extracted data corresponds to more than 15k of parallel data in terms of BLEU improvement.

System	Training Data	BLEU score
Baseline	EP+NC	18.81%
Automotive.5k	EP+NC+5k Automotive	22.02%
Automotive.10k	EP+NC+10k Automotive	23.36%
Automotive.15k	EP+NC+15k Automotive	24.98%
Automotive.20k	EP+NC+20k Automotive	26.48%
Automotive.45k	EP+NC+full Automotive	30.25%

Table 11: Experiments with adding data

The data LEXACC extracts is of high enough quality to be useful for SMT purposes, as the noise is filtered out during the training phase.

6 Conclusions

Parallel sentence mining from comparable corpora is a well-studied problem with several reliable solutions already discussed in the literature. We present yet another original hybrid approach (LEXACC) based on CLIR combined with a complex, trainable translation similarity measure but with a strong emphasis on practical issues such as the reduction of the search space and the behaviour of the translation similarity measure as a function of the comparability level of the corpus (an aspect that is not well studied).

LEXACC is currently used in the ACCURAT project for parallel data mining from comparable corpora and we have presented evidence that it is able to extract good quality parallel sentences that improve SMT systems.

7 Acknowledgements

This work has been supported by the ACCURAT project (<http://www accurat-project.eu/>) funded by the European Community’s Seventh Framework Program (FP7/2007-2013) under the Grant Agreement no. 248347.

References

- Fung, Pascale and Percy Cheung. 2004. *Mining very-non-parallel corpora: parallel sentence and lexicon extraction via bootstrapping and EM*. In: Proceedings of the EMNLP-2004, Barcelona, Spain, pp. 57–63.
- Gale, William A. and Kenneth W. Church. 1993. *A Program for Aligning Sentences in Bilingual Corpora*. Computational Linguistics 19 (1): 75–102.
- Ion, Radu, Alexandru Ceașu and Elena Irimia. 2011. *An Expectation Maximization Algorithm for Textual Unit Alignment*. In: Proceedings of BUCC-2011, Portland, Oregon, USA, pp. 128–135.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin and Evan Herbst. 2007. *Moses: Open Source Toolkit for Statistical Machine Translation*. In: Proceedings of the 45th Annual Meeting of the ACL Companion Volume Proceedings of the Demo and Poster Sessions, Prague, pp. 177–180.
- Koehn, Philipp. 2005. *Europarl: A Parallel Corpus for Statistical Machine Translation*. In: Proceedings of MT Summit 2005.
- Munteanu, Dragos and Daniel Marcu. 2005. *Improving Machine Translation Performance by Exploiting Comparable Corpora*. Computational Linguistics, 31(4): 477–504.
- Och, Franz Josef and Hermann Ney. 2000. *Improved Statistical Alignment Models*. In: Proceedings of the ACL 2000, Hong Kong, China, pp. 440–447.
- Papineni, Kishore, Salim Roukos, Todd Ward and Wei-Jing Zhu. 2001. *Bleu: a Method for Automatic Evaluation of Machine Translation*. IBM Report.
- Quirk, Chris, Raghavendra Udapa U. and Arul Menezes. 2007. *Generative Models of Noisy Translations with Applications to Parallel Fragment Extraction*. In: Proceedings of the MT Summit XI, European Association for Machine Translation.
- Rauf, Sadaf and Holger Schwenk. 2011. *Parallel sentence generation from comparable corpora for improved SMT*. Machine Translation, 25(4): 341–375.
- Stolcke, Andreas. 2002. *SRILM - An Extensible Language Modeling Toolkit*. In: Proceedings of ICSLP, Vol. 2, pp. 901–904.
- Tillmann, Christoph. 2009. *A Beam-Search Extraction Algorithm for Comparable Data*. Proceedings of the ACL-IJCNLP 2009 Conference Short Papers.
- Tufiş, Dan, Radu Ion, Alexandru Ceașu and Dan Ștefănescu. 2006. *Improved Lexical Alignment by Combining Multiple Reified Alignments*. Proceedings of EACL 2006, Trento, Italy, pp. 153–160.

Domain Adaptation of Statistical Machine Translation using Web-Crawled Resources: A Case Study

Pavel Pecina¹, Antonio Toral², Vassilis Papavassiliou³, Prokopis Prokopidis³, Josef van Genabith²

¹Faculty of Mathematics and Physics

²School of Computing

³Institute for Language and

Charles University in Prague

Dublin City University

Speech Processing, Athena RIC

Czech Republic

Dublin 9, Ireland

Athens, Greece

pecina@ufal.mff.cuni.cz {atoral,josef}@computing.dcu.ie {vpapa,prokopis}@ilsp.gr

Abstract

We tackle the problem of domain adaptation of Statistical Machine Translation by exploiting domain-specific data acquired by domain-focused web-crawling. We design and evaluate a procedure for automatic acquisition of monolingual and parallel data and their exploitation for training, tuning, and testing in a phrase-based Statistical Machine Translation system. We present a strategy for using such resources depending on their availability and quantity supported by results of a large-scale evaluation on the domains of Natural Environment and Labour Legislation and two language pairs: English–French, English–Greek. The average observed increase of BLEU is substantial at 49.5% relative.

1 Introduction

Recent advances of Statistical Machine Translation (SMT) have improved Machine Translation (MT) quality to such an extent that it can be successfully used in industrial processes (Flournoy and Duran, 2009). However, this mostly happens in very specific domains for which ample training data is available (Wu et al., 2008). Using in-domain¹ data for training has a substantial effect on the final translation quality: SMT, as any other machine-learning application, is not guaranteed to perform optimally if the data for training and testing are not identically (and independently) distributed, which is often the case in practice. The main problem is usually vocabulary coverage: specific domain texts typically contain vocabulary that is not likely to be found in texts from other domains (Banerjee et al., 2010). Other problems can be caused by divergence in style or genre where the difference is not only in lexis but also in grammar.

© 2012 European Association for Machine Translation.

¹In this work, in-domain always refers to the domain of test data.

In order to achieve optimal performance, an SMT system should be trained on data from the same domain, genre, and style as it is applied to. For many domains, though, in-domain data of a size sufficient to train a full system is hard to find. Recent experiments have shown that even small amounts of such data can be used to adapt a system to the domain of interest (Koehn et al., 2007).

In this work, we present a strategy for automatic web-crawling and cleaning of domain-specific data. Further, our exhaustive experiments, carried out for the Natural Environment (*env*) and Labour Legislation (*lab*) domains and English–French (*EN–FR*) and English–Greek (*EN–EL*) language pairs (in both directions), demonstrate how the crawled data improves SMT quality.

After an overview of related work, we discuss the possibility of adapting a general-domain SMT system by using various types of in-domain data. Then, we present our web-crawling procedure followed by a description of a series of experiments exploiting the data we acquired. Finally, we report on the results and conclude with recommendations for similar attempts to domain adaptation in SMT.

2 Related work and state of the art

2.1 Domain-focused web crawling

A key challenge for a focused crawler that aspires to build domain-specific web collections is the prioritisation of the links to follow. Several algorithms have been exploited for selecting the most promising links. The Best-First algorithm (Cho et al., 1998) sorts the links with respect to their relevance scores and selects a predefined amount of them as the seeds for the next crawling cycle. Menczer and Belew (2000) proposed an adaptive population of agents, called InfoSpiders, and searched for pages relevant to a domain using evolving query vectors and Neural Networks to decide which links to follow. Hybrid models and modifications of these crawling strategies have

<i>language pair (L1–L2)</i>	<i>dom</i>	<i>set</i>	<i>source</i>	<i>sentence pairs</i>	<i>L1 tokens / vocabulary</i>		<i>L2 tokens / vocabulary</i>	
English–French	<i>gen</i>	train	Europarl 5	1,725,096	47,956,886	73,645	53,262,628	103,436
		dev	WPT 2005	2,000	58,655	5,734	67,295	6,913
		test	WPT 2005	2,000	57,951	5,649	66,200	6,876
English–Greek	<i>gen</i>	train	Europarl 5	964,242	27,446,726	61,497	27,537,853	173,435
		dev	WPT 2005	2,000	58,655	5,734	63,349	9,191
		test	WPT 2005	2,000	57,951	5,649	62,332	9,037

Table 1: Detailed statistics of the general-domain data sets obtained from the Europarl corpus and the WPT 2005 workshop.

also been proposed (Gao et al., 2010) with the aim of reaching relevant pages rapidly.

Apart from the crawling algorithm, classification of web content as relevant to a domain or not also affects the acquisition of domain-specific resources, on the assumption that relevant pages are more likely to contain links to more pages in the same domain. Qi and Davison (2009) review features and algorithms used in web page classification. In most of the algorithms reviewed, on-page features (i.e. textual content and HTML tags) are used to construct a corresponding feature vector and then, several machine-learning approaches, such as SVMs, Decision Trees, and Neural Networks, are employed (Yu et al., 2004).

Considering the Web as a parallel corpus, Resnik and Smith (2003) proposed the STRAND system, in which they used Altavista to search for multilingual websites and examined the similarity of the HTML structures of the fetched web pages in order to identify pairs of potentially parallel pages. Similarly, Esplà-Gomis and Forcada (2010) proposed Bitextor, a system that exploits shallow features (file size, text length, tag structure, and list of numbers in a web page) to mine parallel documents from multilingual web sites. Besides structure similarity, other systems either filter fetched web pages by keeping only those containing language markers in their URLs (Désilets et al., 2008), or employ a predefined bilingual wordlist (Chen et al., 2004), or a naive aligner (Zhang et al., 2006) in order to estimate the content similarity of candidate parallel web pages.

2.2 Domain adaptation in SMT

The first attempt towards domain adaptation in SMT was made by Langlais (2002) who integrated in-domain lexicons into the translation model. Eck et al. (2004) presented a language model adaptation technique applying an information retrieval approach based on selecting similar sentences from available training data. Hildebrand et al. (2005) applied the same approach on the translation model. Wu et al. (2005) proposed an align-

ment adaptation approach to improve domain-specific word alignment. Munteanu and Marcu (2005) automatically extracted in-domain bilingual sentence pairs from large comparable (non-parallel) corpora to enlarge the in-domain bilingual corpus. Koehn and Schroeder (2007) integrated in-domain and out-of-domain language models as log-linear features in the Moses (Koehn et al., 2007) phrase-based SMT system with multiple decoding paths for combining multiple domain translation tables. Nakov (2008) combined in-domain translation and reordering models with out-of-domain models into Moses. Finch and Sumita (2008) employed a probabilistic mixture model combining two models for questions and declarative sentences with a general model. They used a probabilistic classifier to determine a vector of probability representing class membership.

In general, all approaches to domain adaptation of SMT depend on the availability of domain-specific data. If the data is available, it can be directly used to improve components of the MT system. Otherwise, it can be extracted from a pool of texts from different domains or even from the web, which is also the case in our work.

3 Resources and their acquisition

In this section, we review the existing resources we used for training the general-domain systems and present the acquisition procedures of in-domain data used for domain adaptation of these systems.

3.1 Existing general domain data

For the baseline, a general-domain system, we exploited the widely used data provided for the SMT workshops (WPT 2005 – WMT 2010): the Europarl parallel corpus (Koehn, 2005) as training data for translation and language models, and WPT 2005 development and test sets as development and test data for general-domain parameter optimization and testing, respectively (Table 1). Europarl is extracted from the European Parliament proceedings and for practical reasons we consider this corpus to contain general-domain texts.

language	dom	initial phase				main phase						
		sites	pages stored /	samples /	acc (%)	sites	pages visited	/ stored	($\Delta\%$)	/ dedup	($\Delta\%$)	t(h)
English	env	146	505	224	92.9	3,181	90,240	34,572	38.3	28,071	18.8	47
	lab	150	461	215	91.6	1,614	121,895	22,281	18.3	15,197	31.8	50
French	env	106	543	232	95.7	2,016	160,059	35,488	22.2	23,514	33.7	67
	lab	64	839	268	98.1	1,404	186,748	45,660	27.2	26,675	41.6	72
Greek	env	112	524	227	97.4	1,104	113,737	31,524	27.7	16,073	49.0	48
	lab	117	481	219	88.1	660	97,847	19,474	19.9	7,124	63.4	38
Average					94.0			25.6		39.7		

Table 2: Statistics from the initial (focused on domain-classification accuracy estimation) and main phases of crawling monolingual data: *stored* refers to the *visited* pages classified as in-domain, *dedup* refers to pages after near-duplicate removal, *time* is the total duration (in hours), *acc* is accuracy estimated on the *sampled* pages, Δ refers to reduction w.r.t. *pages visited*.

language	dom	paragraphs all	/ clean	($\Delta\%$)	/ unique	($\Delta\%$)	sentences	tokens	vocabulary
English	env	5,841,059	1,088,660	18.6	693,971	11.9	1,700,436	44,853,229	225,650
	lab	3,447,451	896,369	26.0	609,696	17.7	1,407,448	43,726,781	136,678
French	env	4,440,033	1,069,889	24.1	666,553	15.0	1,235,107	42,780,009	246,177
	lab	5,623,427	1,382,420	24.6	822,201	14.6	1,232,707	46,992,912	180,628
Greek	env	3,023,295	672,763	22.3	352,017	11.6	655,353	20,253,160	324,544
	lab	2,176,571	521,109	23.9	284,872	13.1	521,358	15,583,737	273,602
Average					23.3	14.0			

Table 3: Statistics from the cleaning stage of the monolingual data acquisition procedure and of the final data set: *clean* refers to paragraphs classified as non-boilerplate, *unique* to those kept after duplicate removal, Δ to reduction w.r.t. *paragraphs all*.

3.2 Web-crawling for monolingual data

To acquire monolingual in-domain corpora used in improving language models, we enhanced a workflow described in Pecina et al. (2011). Considering the small size of crawled data in that work (repeated here as col. 3–6 in Table 2), we implemented a focused monolingual crawler that adopts a distributed computing architecture based on Bixo (2011), an open source web mining toolkit. Moreover, an out-link relevance score l was calculated as: $l = p/N + \sum_{i=1}^M n_i \cdot w_i$, where p is the relevance score of its source page as in Pecina et al. (2011), N is the amount of links originating from the source page, M is the number of entries in a domain definition consisting of relevant terms extracted from Eurovoc², n_i denotes the number of occurrences of the i -th term in the surrounding text and w_i is the weight of the i -th term. Further processing steps include boilerplate detection and language identification at paragraph level. These enhancements resulted in acquiring much more in-domain data (col. 8 in Table 2). In addition, the evolutions of the crawls were satisfactory since the ratio of pages classified as in-domain with the visited ones is 25.6% on average (col. 9 in Table 2).

Then, near-duplicates were removed by employing the deduplication strategy included in the Nutch framework³. The relatively high percentages of documents removed (col. 13 in Table 2) are

in accordance with Baroni et al.’s (2009) observation that during building of the Wacky corpora the amount of documents was reduced by more than 50% after deduplication. Another observation is that the percentages of duplicates for the *lab* domain are much higher than the ones for *env*. This can be explained by the fact that *lab* web pages are mainly legal documents or press releases replicated on many websites.

Final processing of the monolingual data (see Table 3) concerned the exclusion of paragraphs annotated as not in the targeted language or as boilerplate, which reduced their total amount to 23.3% on average (col. 5). Removal of duplicate paragraphs then reduced their total number to 14.0% on average (col. 7). However, most of the removed paragraphs were very short chunks of text (such as navigation links). In terms of tokens, the reduction is only to 50.6%. The last three columns in Table 3 refer to the final monolingual data sets used for training language models. For *EN* and *FR*, we acquired about 45 million tokens for each domain; for *EL*, which is less frequent on the web, we obtained only about 15–20 million tokens.

3.3 Web-crawling for parallel data

Some steps involved in parallel data acquisition (including language identification and cleaning) were discussed in the previous subsection as a part of the monolingual data acquisition. To guide the focused bilingual crawler we used sets of bilin-

²<http://eurovoc.europa.eu/>

³<http://nutch.apache.org>

<i>language pair</i>	<i>dom</i>	<i>sites</i>	<i>docs</i>	<i>sentences all</i>	<i>/ paired</i>	$(\Delta\%)$	<i>/ good</i>	$(\Delta\%)$	<i>/ unique</i>	$(\Delta\%)$	<i>/ sampled</i>	<i>/ corrected</i>
English–French	<i>env</i>	6	559	19,042	14,881	78.1	14,079	73.9	13,840	72.7	3,600	3,392
	<i>lab</i>	4	900	35,870	31,541	87.9	27,601	76.9	23,861	66.5	3,600	3,411
English–Greek	<i>env</i>	14	288	17,033	14,846	87.2	14,028	82.4	13,253	77.8	3,600	3,000
	<i>lab</i>	7	203	13,169	11,006	83.6	9,904	75.2	9,764	74.1	2,700	2,506
<i>Average</i>							84.2	77.1	72.8			

Table 4: Statistics from the parallel data acquisition: document pairs (*docs*), source sentences (*sentences all*), aligned sentence pairs (*paired*), those of sufficient translation quality (*good*); after duplicate removal (*unique*); sentences randomly selected for manual correction (*sampled*) and those really corrected (*corrected*). Δ always refers to percentages w.r.t. the previous step.

gual topic definitions. In order to construct the list of seed URLs we selected web pages that were collected during the monolingual crawls and originated from in-domain multilingual web sites. Since it is likely that these multilingual sites contain parallel documents, we initialize the crawler with these seed URLs and force the crawler to follow only links internal to these sites. After downloading in-domain pages from the selected web sites, we employed Bitextor to identify pairs of documents that could be considered parallel.

3.4 Parallel sentence extraction

After identification of parallel documents, the next steps aimed at extraction of parallel sentences. For each document pair free of boilerplate paragraphs, we applied these steps: sentence splitting and tokenization by the Europarl tools, and sentence alignment by Hunalign (Varga et al., 2005). Hunalign implements a heuristic, language-independent method for identification of parallel sentences in parallel texts which can be improved by providing an external bilingual dictionary of word forms. Without having such dictionaries for *EN–FR* and *EN–EL* at hand, we realign data in these languages from Europarl by Hunalign and used the dictionaries produced by this tool.

For each sentence pair identified as parallel, Hunalign provides a confidence score which reflects the level of parallelness. We manually investigated a sample of sentence pairs extracted by Hunalign from the pool data (about 50 sentence pairs for each language pair and domain), by relying on the judgement of native speakers, and estimated that sentence pairs with a score above 0.4 are of a good translation quality. We kept sentence pairs with 1:1 alignment only (one sentence on each side) and removed those with scores below this threshold. Finally, we also removed duplicate sentence pairs.

The statistics from the parallel data acquisition procedure are given in Table 4. On average, 84.2% of the source sentences extracted from the parallel documents were aligned in the 1:1 fashion (col. 7),

10% of them were removed due to low translation quality, and after discarding duplicate sentences pairs we acquired 72.8% of the original source sentences aligned to their target sides (col. 11).

The translation quality of the parallel sentences obtained by the procedure described above is not guaranteed in any sense. Tuning the procedure and focusing on high-quality translations is possible but leads to a trade-off between quality and quantity. For translation model training, high translation quality of the data is not as essential as for testing. Bad phrase pairs can be removed from the translation tables based on their low translation probabilities. However, a development set containing sentence pairs which are not good translations of each other might lead to sub-optimal values of model weights which would harm system performance. If such sentence pairs are used in the test set, the evaluation would clearly be unreliable.

In order to create reliable test and development sets for each language pair and domain, we performed the following low-cost procedure. From the data obtained by the steps described in the previous section, we selected a random sample of 3,600 sentence pairs (2,700 for *EN–EL* in the *lab* domain, for which less data was available) and asked native speakers to check and correct them. The task consisted of checking that the sentence pairs belonged to the right domain, the sentences within a sentence pair were equivalent in terms of content, and the translation quality was adequate and (if needed) correcting it. The goal was to obtain at least 3,000 correct sentence pairs for each domain and language pair; thus the correctors did not have to correct every sentence pair. They were allowed to skip (remove) misaligned sentence pairs and asked to remove those sentence pairs that were obviously from a very different domain (despite being correct translations). The number of corrected sentences is in the last column of Table 4.

According to the human judgements (see Table 5), 53–72% of sentence pairs were accurate translations, 22–34% needed only minor corrections, 1–

category	EN-EL / env	EN-FR / lab
1. perfect translation	53.49	72.23
2. minor corrections done	34.15	21.99
3. major corrections needed	3.00	0.33
4. misaligned sentence pair	5.09	1.58
5. wrong domain	4.28	3.86

Table 5: Results (%) of the manual correction of parallel data.

3% would require major corrections (which was not necessary, as the accurate sentence pairs together with those requiring minor corrections were enough to reach our goal of at least 3,000 sentence pairs in most cases), 2–5% of sentence pairs were misaligned and would have had to be translated completely, and about 4% were from a different domain (despite being correct translations).

Further, we selected 2,000 pairs from the corrected sentences for the test set and left the remaining part for the development set. The parallel sentences which were not selected for corrections were used as training sets. See further statistics in Table 6. The correctors confirmed that the manual corrections were about 5–10 times faster than translating the sentences from scratch, so this can be viewed as low-cost method for acquiring in-domain test and development sets for SMT.

4 Domain adaptation experiments

In this section, we present experiments that exploit all the acquired in-domain data in eight different evaluation scenarios involving two domains (*env*, *lab*) and two language pairs (*EN-FR*, *EN-EL*) in both directions. Our primary evaluation measure is BLEU (Papineni et al., 2002). For detailed analysis we also present NIST (Doddington, 2002) and METEOR (Banerjee and Lavie, 2005) in Table 8.

4.1 System description

Our MT system is based on Moses (Koehn et al., 2007). For training the baseline system, training data is tokenized and lowercased using the Europarl tools. The original (non-lowercased) target sides of the parallel data are kept for training the Moses recaser. The lowercased versions of the target sides are used for training an interpolated 5-gram language model with Kneser-Ney discounting using the SRILM toolkit (Stolcke, 2002). Translation models are trained on the relevant parts of the Europarl corpus, lowercased and filtered on sentence level; we kept all sentence pairs having less than 100 words on each side and with length ratio within the interval (0.11,9.0). The maximum

pair	dom	set	sents	L1 tokens / voc	L2 tokens / voc
English-French	<i>env</i>	train	10,240	300,760 / 10,963	362,899 / 14,209
		dev	1,392	41,382 / 4,660	49,657 / 5,542
		test	2,000	58,865 / 5,483	70,740 / 6,617
	<i>lab</i>	train	20,261	709,893 / 12,746	836,634 / 17,139
		dev	1,411	52,156 / 4,478	61,191 / 5,535
		test	2,000	71,688 / 5,277	84,397 / 6,630
English-Greek	<i>env</i>	train	9,653	240,822 / 10,932	267,742 / 20,185
		dev	1,000	27,865 / 3,586	30,510 / 5,467
		test	2,000	58,073 / 4,893	63,551 / 8,229
	<i>lab</i>	train	7,064	233,145 / 7,136	244,396 / 14,456
		dev	506	15,129 / 2,227	16,089 / 3,333
		test	2,000	62,953 / 4,022	66,770 / 7,056

Table 6: Details of the in-domain parallel data sets obtained by web-crawling and manual correction: sentence pairs (*sents*), source (*L1*) and target (*L2*) tokens and vocabulary size (*voc*).

length of aligned phrases is set to 7 and the re-ordering models are generated using parameters: *distance*, *orientation-bidirectional-fe*. The model parameters are optimized by Minimum Error Rate Training (Och, 2003, MERT) on development sets.

For decoding, test sentences are tokenized, lowercased, and translated by the tuned system. Letter casing is then reconstructed by the recaser and extra blank spaces in the tokenized text are removed in order to produce human-readable text.

4.2 Using out-of-domain test data

A number of previous experiments (Wu et al., 2008; Banerjee et al., 2010, e.g.) showed significant degradation of translation quality if an SMT system was applied to out-of-domain data. In order to verify this observation we trained and tuned our system on general-domain data and compared its performance on test sets from general (*gen*) and specific (*env*, *lab*) domains (the results are referred to as vX and $v0$ in Table 7, respectively). The average decrease in BLEU is 44.3%: while on general-domain test sets we observe scores in the interval 42.24–57.00, the scores on the specific-domain test sets are in the range 20.20–31.79. This is presumably caused by the divergence of training and test data: the out-of-vocabulary (OOV) rate increased from 0.25% to 0.90% (see col. 4 and 16 in Table 7).

4.3 Using in-domain development data

Optimization of parameters of the SMT log-linear models is known to have a big influence on the performance. The first step towards domain adaptation of a general-domain system it to use in-domain development data. Such data usually comprises of a small set of parallel sentences which are repeatedly translated while the model parameters are adjusted towards their optimal val-

<i>direction</i>	<i>dom</i>	<i>vX / OOV</i>		<i>dom</i>	<i>v0 / OOV</i>		<i>v1 / Δ%</i>		<i>v2 / Δ%</i>		<i>v3 / Δ%</i>		<i>v4 / Δ% / OOV</i>		
English–Fench	<i>gen</i>	49.12	0.11	<i>env</i>	28.03	0.98	35.81	27.8	39.23	40.0	40.53	44.6	40.72	45.3	0.65
				<i>lab</i>	22.26	0.85	30.84	35.6	34.00	52.7	39.55	77.7	39.35	76.8	0.48
Fench–English	<i>gen</i>	57.00	0.11	<i>env</i>	31.79	0.81	39.04	22.5	40.57	27.6	42.23	32.8	42.17	32.7	0.54
				<i>lab</i>	27.00	0.68	33.52	23.7	38.07	41.0	44.14	63.5	43.85	62.4	0.38
English–Greek	<i>gen</i>	42.24	0.22	<i>env</i>	20.20	1.15	26.18	29.1	32.06	58.7	33.83	67.5	34.50	70.8	0.82
				<i>lab</i>	22.92	0.47	28.79	25.7	33.59	46.6	33.54	46.3	33.71	47.1	0.40
Greek–English	<i>gen</i>	44.15	0.56	<i>env</i>	29.23	1.53	34.15	16.8	36.93	26.3	39.13	33.9	39.18	34.0	1.20
				<i>lab</i>	31.71	0.69	37.55	18.4	40.17	26.7	40.44	27.5	40.33	27.2	0.62
Average		0.25			0.90		25.5		40.0		49.2		49.5	0.64	

Table 7: BLEU scores from domain adaptation of the baseline general-domain systems (*v0*) by exploiting: corrected devel. data (*v1*), monolingual training data (*v2*), parallel training data (*v3*), both monolingual and parallel training data (*v4*); *vX* refers to the baseline systems applied to general-domain test sets, *OOV* to out-of-vocabulary rates, Δ to relative improvement over *v0*.

ues. The minimum number of development sentences is not strictly given. The only requirement is that the optimization procedure (MERT in our case) must converge, which might not happen if the set is too small. By using the parallel data acquisition procedure (see Section 3.2), we acquired development sets (506–1,411 sentence pairs in each) which proved to be very beneficial: compared to the baseline systems trained and tuned on general-domain data only (*v0*), systems trained on general-domain data and tuned on in-domain data (*v1*) improved BLEU scores by 25.5% on average. Taking into account that the development sets contain only several hundreds of parallel sentences each, such improvement is remarkable (compare columns *v0* and *v1* in Table 7).

4.4 Adding in-domain monolingual data

Improving an SMT system by adding in-domain monolingual training data cannot reduce the relatively high OOV rate observed when general-domain systems were applied on test sets from specific domains. However, such data can improve the language models and contribute to better estimations of probabilities of n-grams consisting of known words. To verify this hypothesis, we trained systems (*v2*) on general-domain parallel training data, in-domain development data, and a concatenation of general-domain and in-domain monolingual data described in Section 3.2.1 (comprising 15–45 million words). Compared to the systems *v1*, the BLEU scores were improved by additional 14.5% absolute on average. In comparison with the baseline systems *v0*, the total increase of BLEU is 40.0% on average. The most substantial improvement over the system *v1* is achieved for translations to Greek (23.0% for *env*, 16.2% for *lab*) despite the smallest size of the monolingual data acquired for this language (Table 3) which is probably due to the complex Greek morphology.

4.5 Adding in-domain parallel training data

Parallel data is essential for building translation models of SMT systems. While a good language model can improve an SMT system by preferring better translation options in given contexts, it has no effect if the translation model offers no translation at all, which is the case for OOV words. In the next experiment, we use in-domain parallel training data acquired as described in Section 3.2.3 (7–20 thousand sentence pairs). First, we trained systems (*v3*) on a concatenation of general-domain and in-domain parallel training data, in-domain development data, and a general-domain monolingual data only which outperformed the previous systems (*v2*) by additional 9.2% absolute on average (49.2% over the baseline). In some scenarios, the overall improvement was above 70%.

To provide a complete picture we also trained fully adapted systems (*v4*) using both general-domain and in-domain sets of parallel and monolingual data and tuned on the corrected in-domain development sets. In most scenarios the difference of results of these systems compared to systems *v3* are not statistically significant ($p=0.05$). The average relative improvement over the baseline (*v0*) is 49.5%, which is almost identical to 49.2% from the previous experiment (*v3*). In practice, this means that using additional monolingual in-domain data on top of the in-domain parallel data has no effect on the translation quality. Although additional experiments would verify whether larger monolingual data could bring any additional improvement or not, it seems that parallel data is more important.

5 Conclusions

We presented two methods for the acquisition of domain-specific monolingual and parallel data from the web. They employ existing open-source tools for normalization, language identification,

		Natural Environment								Labour Legislation							
sys	BLEU/ $\Delta\%$	NIST/ $\Delta\%$	MET / $\Delta\%$	WER / $\Delta\%$	BLEU/ $\Delta\%$	NIST/ $\Delta\%$	MET / $\Delta\%$	WER / $\Delta\%$	BLEU/ $\Delta\%$	NIST/ $\Delta\%$	MET / $\Delta\%$	WER / $\Delta\%$	BLEU/ $\Delta\%$	NIST/ $\Delta\%$	MET / $\Delta\%$	WER / $\Delta\%$	
English-French	v0	28.03	0.0	7.03	0.0	63.32	0.0	63.70	0.0	22.26	0.0	6.27	0.0	56.73	0.0	69.93	0.0
	v1	35.81	27.7	8.10	15.2	68.44	8.0	53.78	-15.5	30.84	38.5	7.42	18.3	62.94	10.9	57.99	-17.0
	v2	39.23	39.9	8.43	19.9	70.35	11.1	51.34	-19.4	34.00	52.7	7.68	22.4	65.56	15.5	57.06	-18.4
	v3	40.53	44.6	8.61	22.4	71.10	12.2	50.04	-21.4	39.55	77.6	8.37	33.4	69.82	23.0	52.04	-25.5
	v4	40.72	45.2	8.63	22.7	71.23	12.4	49.92	-21.6	39.35	76.7	8.34	33.0	69.79	23.0	52.29	-25.2
French-English	v0	31.79	0.0	7.77	0.0	66.25	0.0	57.09	0.0	27.00	0.0	7.07	0.0	59.90	0.0	61.57	0.0
	v1	39.04	22.8	8.75	12.6	69.17	4.4	48.26	-15.4	33.52	24.1	7.98	12.8	63.70	6.3	53.39	-13.2
	v2	40.57	27.6	8.90	14.5	70.23	6.0	47.19	-17.3	38.07	41.0	8.47	19.8	66.88	11.6	50.35	-18.2
	v3	42.23	32.8	9.09	16.9	71.40	7.7	46.07	-19.3	44.14	63.4	9.22	30.4	71.24	18.9	45.49	-26.1
	v4	42.17	32.6	9.09	16.9	71.32	7.6	46.05	-19.3	43.85	62.4	9.17	29.7	71.07	18.6	45.81	-25.6
English-Greek	v0	20.20	0.0	5.73	0.0	82.81	0.0	67.83	0.0	22.92	0.0	5.93	0.0	87.27	0.0	65.88	0.0
	v1	26.18	29.6	6.57	14.6	84.19	1.6	60.80	-10.3	28.79	25.6	6.80	14.6	87.91	0.7	58.20	-11.6
	v2	32.06	58.7	7.24	26.3	84.52	2.0	56.68	-16.4	33.59	46.5	7.36	24.1	88.34	1.2	54.71	-16.9
	v3	33.83	67.4	7.63	33.1	86.10	3.9	53.47	-21.1	33.54	46.3	7.34	23.7	89.55	2.6	54.68	-17.0
	v4	34.50	70.7	7.57	32.1	85.91	3.7	54.16	-20.1	33.71	47.0	7.34	23.7	89.42	2.4	54.71	-16.9
Greek-English	v0	29.23	0.0	7.50	0.0	60.57	0.0	54.69	0.0	31.71	0.0	7.76	0.0	62.42	0.0	52.34	0.0
	v1	34.16	16.8	8.01	6.8	64.98	7.2	51.15	-6.4	37.55	18.4	8.28	6.7	67.36	7.9	49.02	-6.3
	v2	36.93	26.3	8.27	10.2	66.60	9.9	49.40	-9.6	40.17	26.6	8.58	10.5	68.67	10.0	47.03	-10.1
	v3	39.13	33.8	8.55	14.0	68.24	12.6	47.94	-12.3	40.44	27.5	8.61	10.9	68.91	10.4	46.78	-10.6
	v4	39.18	34.0	8.54	13.8	68.19	12.5	47.94	-12.3	40.33	27.1	8.60	10.8	68.83	10.2	47.00	-10.2

Table 8: Complete results of the domain adaptation experiments. With the exception of NIST, all scores are percentages; MET denotes METEOR, system identifiers refer to those in Table 7, and Δ to relative improvement over the baseline systems v0.

cleaning, deduplication, and parallel sentence extraction. These methods were applied to acquire monolingual and parallel data for two language pairs and two domains with only minimal manual intervention (domain definitions and seed URLs).

The acquired resources were then successfully used to adapt general-domain SMT systems to the new domains. The average relative improvement of BLEU achieved in eight scenarios was a substantial 49.5%. Based on our experiments we made the following observations: even small amounts of in-domain parallel data is more important for translation quality than large amounts of in-domain monolingual data. As few as 500–1,000 sentence pairs can be used as development data with expected 25% relative improvement of BLEU. Additional parallel data can be used to improve translation models: 7,000–20,000 sentences pairs in our experiments increased BLEU by other 25% relative on average. If such data is not available, a general-domain system can benefit from using additional in-domain monolingual data, however quite large amounts (tens of million words) are necessary to obtain a moderate improvement.

Acknowledgments

This research was supported by the EU FP7 project PANACEA (contract no. 7FP-ITC-248064) and by the Czech Science Foundation (grant no.

P103/12/G084). We thank Victoria Arranz, Olivier Hamon, and Khalid Choukri for their help with manual correction of the *EN-FR* data; Maria Giagkou and Voula Giouli for construction of the domain definitions and correction of the *EN-EL* data.

References

- Banerjee, S. and A. Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proc. of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp 65–72, Ann Arbor, Michigan.
- Banerjee, P., J. Du, B. Li, S. Naskar, A. Way, and J. van Genabith. 2010. Combining Multi-Domain Statistical Machine Translation Models using Automatic Classifiers. In *The Ninth Conference of the Association for MT in the Americas*, pp 141–150.
- Baroni, M., S. Bernardini, A. Ferraresi, and E. Zanchetta. 2009. The WaCky Wide Web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.
- Bixo. 2011. Web mining toolkit. <http://openbixo.org/>.
- Chen, J., R. Chau, and C.-H. Yeh. 2004. Discovering parallel text from the World Wide Web. In *Proc. of the 2nd workshop on Australasian information security, Data Mining and Web Intelligence, and Software Internationalisation*, volume 32, pp 157–161, Darlinghurst, Australia.
- Cho, J., H. Garcia-Molina, and L. Page. 1998. Efficient crawling through URL ordering. *Comput. Netw. ISDN Syst.*, 30:161–172.

- Désilets, A., B. Farley, M. Stojanovic, and G. Pate-naude. 2008. WeBiText: Building Large Heterogeneous Translation Memories from Parallel Web Content. In *Proc. of Translating and the Computer (30)*, London, UK.
- Doddington, G. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proc. of the second international conference on Human Language Technology Research*, pp 138–145, San Diego, California.
- Eck, M., S. Vogel, and A. Waibel. 2004. Language Model Adaptation for Statistical Machine Translation based on Information Retrieval. In *International Conference on Language Resources and Evaluation*, Lisbon, Portugal.
- Esplà-Gomis, M. and M. L. Forcada. 2010. Combining Content-Based and URL-Based Heuristics to Harvest Aligned Bitexts from Multilingual Sites with Bitextor. *The Prague Bulletin of Mathematical Linguistics*, 93:77–86.
- Finch, A. and E. Sumita. 2008. Dynamic model interpolation for statistical machine translation. In *Proc. of the Third Workshop on Statistical Machine Translation*, pp 208–215, Columbus, Ohio, USA.
- Flournoy, R. and C. Duran. 2009. Machine translation and document localization at Adobe: from pilot to production. In *MT Summit XII: proc. of the twelfth Machine Translation Summit*, pp 425–428.
- Gao, Z., Y. Du, L. Yi, Y. Yang, and Q. Peng. 2010. Focused Web Crawling Based on Incremental Learning. *Journal of Comp. Information Systems*, 6:9–16.
- Hildebrand, A. S., M. Eck, S. Vogel, and A. Waibel. 2005. Adaptation of the Translation Model for Statistical Machine Translation based on Information Retrieval. In *Proc. of the 10th Annual Conference of the European Association for Machine Translation*, pp 133–142, Budapest, Hungary.
- Hua, W., W. Haifeng, and L. Zhanyi. 2005. Alignment model adaptation for domain-specific word alignment. In *43rd Annual Meeting on Association for Computational Linguistics*, pp 467–474, Ann Arbor, Michigan, USA.
- Koehn, P. and J. Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proc. of the Second Workshop on Statistical Machine Translation*, pp 224–227, Prague, Czech Rep.
- Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proc. of the 45th Annual Meeting of the ACL on Interactive Poster and Demo Sessions*, pp 177–180, Prague, Czech Rep.
- Koehn, P. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proc.: the tenth Machine Translation Summit*, pp 79–86, Phuket, Thailand.
- Kohlschütter, C., P. Fankhauser, and W. Nejdl. 2010. Boilerplate detection using shallow text features. In *Proc. of the 3rd ACM International Conference on Web Search and Data Mining*, pp 441–450, NY.
- Langlais, P. 2002. Improving a general-purpose Statistical Translation Engine by terminological lexicons. In *COLING-02 on COMPUTERM 2002: second international workshop on computational terminology - Volume 14*, pp 1–7, Taipei, Taiwan.
- Menczer, F. and R. K. Belew. 2000. Adaptive Retrieval Agents: Internalizing Local Context and Scaling up to the Web. *Machine Learning*, 39:203–242.
- Munteanu, D. S. and D. Marcu. 2005. Improving Machine Translation Performance by Exploiting Non-Parallel Corpora. *Comput. Linguist.*, 31:477–504.
- Nakov, P. 2008. Improving English-Spanish statistical machine translation: experiments in domain adaptation, sentence paraphrasing, tokenization, and recasing. In *Proc. of the Third Workshop on Statistical Machine Translation*, pp 147–150, Columbus, USA.
- Och, F. J. 2003. Minimum error rate training in statistical machine translation. In *41st Annual Meeting on Association for Computational Linguistics*, pp 160–167, Sapporo, Japan.
- Papineni, K., S. Roukos, T. Ward, and W.-J. Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *40th Annual Meeting on Association for Computational Linguistics*, pp 311–318, Philadelphia, USA.
- Pecina, P., A. Toral, A. Way, V. Papavassiliou, P. Prokopidis, and M. Giagkou. 2011. Towards Using Web-Crawled Data for Domain Adaptation in Statistical Machine Translation. In *Proc. of the 15th Annual Conference of the European Association for Machine Translation*, pp 297–304, Leuven, Belgium.
- Qi, X. and B. D. Davison. 2009. Web page classification: Features and algorithms. *ACM Computing Surveys*, 41:12:1–12:31.
- Resnik, P. and N. A. Smith. 2003. The Web as a parallel corpus. *Computational Linguistics*, 29:349–380.
- Stolcke, A. 2002. SRILM—an extensible language modeling toolkit. In *Proc. of International Conference on Spoken Language Processing*, pp 257–286, Denver, Colorado, USA.
- Varga, D., L. Németh, P. Halácsy, A. Kornai, V. Trón, and V. Nagy. 2005. Parallel corpora for medium density languages. In *Recent Advances in Natural Language Processing*, pp 590–596.
- Wu, H., H. Wang, and C. Zong. 2008. Domain adaptation for statistical machine translation with domain dictionary and monolingual corpora. In *Proc. of the 22nd International Conference on Computational Linguistics - Volume 1*, pp 993–1000.
- Yu, H., J. Han, and K. C.-C. Chang. 2004. PEBL: Web Page Classification without Negative Examples. *IEEE Transactions on Knowledge and Data Engineering*, 16(1):70–81.
- Zhang, Y., K. Wu, J. Gao, and P. Vines. 2006. Automatic Acquisition of Chinese-English Parallel Corpus from the Web. In *Proc. of the 28th European Conference on Information Retrieval*, pp 420–431.

Relevance Ranking for Translated Texts

Marco Turchi, Josef Steinberger

European Commission JRC

IPSC - GlobeSec

Via Fermi 2749,

21020 Ispra (VA), Italy

name.surname@jrc.ec.europa.eu

Lucia Specia

Department of Computer Science

University of Sheffield

Regent Court, 211 Portobello

Sheffield, S1 4DP, UK

L.Specia@dcs.shef.ac.uk

Abstract

The usefulness of a translated text for gisting purposes strongly depends on the overall translation quality of the text, but especially on the translation quality of the most informative portions of the text. In this paper we address the problems of ranking translated sentences within a document and ranking translated documents within a set of documents on the same topic according to their informativeness and translation quality. An approach combining quality estimation and sentence ranking methods is used. Experiments with French-English translation using four sets of news commentary documents show promising results for both sentence and document ranking. We believe that this approach can be useful in several practical scenarios where translation is aimed at gisting, such as multilingual media monitoring and news analysis applications.

1 Introduction

Reading and understanding the main ideas behind documents written in different languages can be necessary or desirable in a number of scenarios. Existing online translation systems such as *Google Translate* and *Bing Translator*¹ serve to this purpose, mitigating the language barrier effects. Despite the large improvements in translation quality in recent years, translated documents are still affected by the presence of sentences which are not correctly translated and in the extreme case,

whose original meaning has been lost. These sentences can compromise the readability and reliability of translated documents, especially if they are the ones that should convey the most important information in the document.

Quality estimation methods can flag incorrect translations without access to reference sentences, however the informativeness of these sentences is not taken into account. On the other hand, sentence ranking methods are able to identify the most relevant sentences in a given language for tasks such as document summarisation. However, the performance of sentence ranking algorithms for machine translated texts can be significantly degraded due to the introduction of errors by the translation process, as it has been shown for other language processing tasks, e.g. in information retrieval (Savoy and Dolamic, 2009). Moreover, particularly in the case of supervised ranking methods, these may only be available for the source language.

In this paper we propose combining quality estimation and relevance sentence ranking methods in order to identify the most relevant translated texts. We experiment with two ranking tasks:

- The ranking of translated sentences within a document; and
- The ranking of documents within a set of documents on the same topic.

An evaluation with French-English translations in groups of news commentary documents in different domains has shown promising results for both sentence and document ranking.

2 Related work

A considerable amount of work has been dedicated in recent years to estimating the quality of ma-

©2012 European Association for Machine Translation.

¹translate.google.com/ and www.microsofttranslator.com/

chine translated texts, i.e., the problem of predicting the quality of translated text without access to reference translations. Most related work focus on predicting different types of sentence-level quality scores, including automatic and semi-automatic MT evaluation metrics such as TER (He et al., 2010), HTER (Specia and Farzindar, 2010; Bach et al., 2011), post-editing effort scores and post-editing time (Specia, 2011). At document level, similar to this paper, Soricut and Echihabi (2010) focus on the ranking translated documents according to their estimated quality so that the top n documents can be selected for publishing. A range of indicators from the MT system, source and translation texts have been used in previous work. However, none of these include the notion of informativeness of the texts.

The sentence ranking problem has been widely studied in particular for document summarization, where different approaches have been proposed to quantify the amount of information contained in each sentence. In (Goldstein et al., 1999), a technique called Maximal Marginal Relevance (MMR) was introduced to measure the relevance of each sentence in a document according to a user provided query. Other approaches represent a document as a set of trees and take the position of a sentence in a tree is indicative of its importance (Carlson et al., 2001). Graph theory has been extensively used to rank sentences (Yeh et al., 2008) or keywords (Mihalcea, 2004), with their importance determined using graph connectivity measures such as in-degree or PageRank. A sentence extraction method based on Singular Value Decomposition over term-by-sentence matrices was introduced in (Gong and Xin, 2002).

The combination of relevance and translation quality scores has been recently proposed in the context of cross-language document summarization. In (Wan et al., 2010), sentences in a document were ranked using the product of quality estimation and relevance scores, both computed using the source text only. The best five sentences were added to a summary, and then translated to the target language. (Boudin et al., 2010) used both source and target language features for quality estimation and targeted multi-document summarization, selecting sentences from different translated documents to generate a summary.

This paper extends previous work in the attempt to rank translated sentences within documents, but

with a different objective: instead of selecting a pre-defined number of sentences to compose a summary, we aim at obtaining a global ranking of sentences within a document according to their informativeness and translation quality and use this ranking to assign a global score to each document for the ranking of groups of documents. This requires different evaluation strategies from those used in the text summarization field, as we will discuss in Section 5.2.

3 Quality estimation method

The quality estimation method used in this paper is that proposed in (Specia, 2011). A sentence-level model is built using a Support Vector Machines regression algorithm with radial basis function kernel from the LIBSVM package (Chang and Lin, 2011) and a number of shallow and MT system-independent features. These features are extracted from the source sentences and their corresponding translations, and from monolingual and parallel corpora. They include source & translation sentence lengths, source & translation sentence language model probabilities, average number of translations per source word, as given by probabilistic dictionaries, percentages of numbers, content-/non-content words in the source & translation sentences, among others. The regression algorithm is trained on examples of translations and their respective human judgments for translation quality (Section 5.1).

4 Sentence ranking methods

4.1 Co-occurrence-based ranking

Originally proposed by (Gong and Xin, 2002) and later improved by (Steinberger and Ježek, 2004), this is an unsupervised method based on the application of Singular Value Decomposition (SVD) to individual documents or sets of documents on the same topic. It has been reported to have the best performance in the multilingual multi-document summarization task at TAC 2011. The method first builds a term-by-sentence matrix from the text, then applies SVD and uses the resulting matrices to identify and extract the most salient sentences. SVD is aimed at finding the latent (orthogonal) dimensions, which would correspond to the different topics discussed in the set of documents.

More formally, we first build a matrix \mathbf{A} where each column represents the weighted term-frequency vector of a sentence j in a given docu-

ment or set of documents. The weighting schemes found to work best in (Steinberger and Ježek, 2009) are a binary local weight and an entropy-based global weight.

After that step, SVD is applied to the matrix as $\mathbf{A} = \mathbf{USV}^T$, and subsequently a matrix $\mathbf{F} = \mathbf{S} \cdot \mathbf{V}^T$ reduced to r dimensions² is derived.

Sentence selection starts with measuring the length of the sentence vectors in \mathbf{F} . This length can be viewed as a measure of the importance of that sentence within the top topics (the most important dimensions). In other words, the length corresponds to the combined weight across the most important topics. We call it *co-occurrence sentence score*. The sentence with the largest score is selected as the most informative (its corresponding vector in \mathbf{F} is denoted by \mathbf{f}_{best}). To prevent selecting a sentence with similar content in the next step, the topic/sentence distribution in matrix \mathbf{F} is changed by subtracting the information contained in the selected sentence:

$$\mathbf{F}^{(it+1)} = \mathbf{F}^{(it)} - \frac{\mathbf{f}_{best} \cdot \mathbf{f}_{best}^T}{|\mathbf{f}_{best}|^2} \cdot \mathbf{F}^{(it)}$$

The vector lengths of similar sentences are thus decreased, which avoids selecting the same/similar sentences. We call this a *redundancy filter*. After this subtraction, the process continues with the sentence which has the largest co-occurrence sentence score computed on the updated matrix \mathbf{F}^1 (the first update of the original matrix \mathbf{F}^0). The process is repeated until all the sentences of the document(s) are annotated with their co-occurrence sentence score.

Since it is unsupervised, in our work this method was applied to both the source language texts and the translated texts.

4.2 Profile-based ranking

The supervised profile-based ranking algorithm by (Pouliquen et al., 2003) was proposed for addressing the multi-label categorization problem using the Eurovoc thesaurus³. Models for thousands of categories were trained using only positive samples for each category. The training process consisted in identifying a list of representative words and associating to each of them a log-likelihood

²The degree of importance of each ‘latent’ topic is given by the singular values and the optimal number of latent topics (i.e., dimensions) r can be tuned on some development data.

³Eurovoc.europa.eu/

weight, using the training set as the reference corpus. A new document was represented as a vector of words with their frequency in the document. The most appropriate categories for the new document were found by ranking the category vector representations (the *profiles*) according to their cosine similarity to the vector representation of the new document.

In this paper we are primarily interested in the ranking of sentences, as opposed to the ranking of categories. Since we know beforehand which category (a topic of interest) a document belongs to, a profile vector is created for that category using human labeled data. The cosine similarity for each sentence in the document and the category vector is computed and all the sentences are ranked according to their cosine value.

In our work this method was applied to the source language sentences only.

5 Experimental settings

5.1 Corpora

Relevance ranking training The profile-based method (Section 4.2) is trained using 1,000 French news documents for each of our four topics of interest. These documents were selected using an in-house news categorization system (Steinberger et al., 2009), where category definitions are created by humans. Articles are said to fall into a given category if they satisfy the category definition, which consists of Boolean operators with optional vicinity operators and wild cards. Alternative classifiers can also be trained using the Eurovoc human labeled multi-lingual resource.

Quality estimation training To train the regression algorithm for the quality estimation model we use the French-English corpus created in (Specia, 2011), which is freely available⁴. This corpus contains 2,525 French news sentences from the WMT *news-test2009* dataset and their translations into English using a statistical machine translation system built from the Moses toolkit⁵. These sentences were scored by a human translator according to the effort necessary to correct them: 1 = requires complete retranslation; 2 = requires some retranslation; 3 = very little

⁴www.dcs.shef.ac.uk/~lucia/resources.html

⁵www.statmt.org/wmt10/

post editing needed; 4 = fit for purpose. An average human score of 2.83 was reported.

Evaluation corpus To evaluate the performance of our approach we use the multilingual summary evaluation dataset created by Turchi et al. (2010)⁶. It contains four sets of documents covering four topics: *Israeli-Palestinian conflict (IPC)*, *Malaria (M)*, *Genetics (G)* and *Science and Society (SS)*. Each set contains five documents, here in French. All sentences (amounting to 789) in these documents were annotated by four human annotators with binary labels indicating whether or not it is informative to that topic. Therefore, the final score for each sentence is a discrete number ranging from 0 (uninformative) to 4 (very informative). These French sentences were then translated using the same Moses system as in the training set for quality estimation and annotated for quality using the 1-4 scoring scheme. The average human quality scores are shown in Table 2.

5.2 Evaluation metrics for ranking

Our goal is to find the best possible ranking of translated sentences and documents according to their relevance and translation quality. While the ranked sentences/documents could be used for many applications, including cross-lingual summarization, we are interested in a more general ranking approach, and therefore our evaluation is task-independent. We use the following metrics:

Sentence ranking Sentences in the system output and gold standard documents are first ordered according to their combined score for relevance and translation quality (or relevance score only, for the monolingual ranking evaluation, Table 1). We then compute the Spearman’s rank correlation coefficient (ρ) between the two rankings. Additionally, inspired by the vBLEU Δ metric (Soricut and Echiabi, 2010), we compute $Avg\Delta$, a metric that measures the relative gain (or loss) in performance obtained from selecting the top $k\%$ sentences ranked according to the predicted scores, as compared to the performance obtained from randomly selecting $k\%$ sentences:

$$Avg\Delta = (Avg_{sys} - Avg_{gold})$$

⁶langtech.jrc.it/JRC_Resources.html

where Avg_{gold} is the average *gold-standard* score for *all* sentences in the test set (i.e., the approximate score if sentences are randomly taken) and Avg_{sys} is the average *gold-standard* score for the top $k\%$ sentences from the test set ranked according to the predicted (system) scores.

Intuitively, the smaller the k , the higher the upper bound $Avg\Delta$, but the harder the ranking task becomes. Larger values of k should result in smaller values for $Avg\Delta$. For $k = 100$, $Avg\Delta = 0$. In this paper we compute $Avg\Delta$ over different values of k : 10, 25 and 50, and consider the arithmetic mean over these values of k as our final metric, $Avg\Delta_{all}$.

Document ranking Likewise in sentence ranking, both gold-standard and system rankings for the documents are compared. Since there are only five documents within each set of documents, Spearman’s rank correlation coefficient would not be reliable. We instead evaluate the pairwise rankings of documents using Cohen’s Kappa coefficient (κ) (Cohen, 1960), defined as: $\kappa = \frac{P(A)-P(E)}{1-P(E)}$, where $P(A)$ is the proportion of times the gold-standard and system ranking agree on the ranking of a pair of documents and $P(E)$ is the proportion of times they could agree by chance. This probability is empirically computed by observing the frequency of ties, as in (Callison-Burch et al., 2011).

6 Experiments and results

In what follows we show the results of the quality estimation and relevance ranking methods on their own and then we present the results obtained with the combination of these two methods.

6.1 Quality estimation

The performance of the quality estimation method is shown in Table 2. The average regression error is measured using Root Mean Squared Error, $RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}$, where N is the number of test sentences, \hat{y} is the predicted score and y is the actual score for that test sentence. The performance is generally lower than what has been reported in (Specia, 2011) for French-English and similar settings ($RMSE = 0.662$). The decrease in performance is most likely due to the difference in the text domain of the training and test

	G		IPC		M		SS		Macro Av.	
	$Avg\Delta_{all}$	ρ	$Avg\Delta_{all}$	ρ	$Avg\Delta_{all}$	ρ	$Avg\Delta_{all}$	ρ	$Avg\Delta_{all}$	ρ
<i>InvPos</i>	-0.254	-0.088	-0.08	0.006	-0.22	0.012	0.132	0.015	-0.105	-0.013
<i>Length</i>	0.287	0.328	0.322	0.278	0.75	0.541	0.156	0.113	0.378	0.315
PB 1000	0.312	0.285	0.358	0.321	0.329	0.286	0.227	0.072	0.307	0.242
PB 2000	0.568	0.401	0.568	0.338	0.385	0.303	0.154	0.141	0.419	0.296
PB 5000	0.478	0.249	0.503	0.31	0.607	0.451	0.046	0.095	0.409	0.271
Co_R_S 25	0.293	0.364	0.469	0.301	0.544	0.428	0.203	0.244	0.377	0.335
Co_NR_S 2	0.267	0.269	0.388	0.236	0.28	0.389	0.607	0.367	0.386	0.316
Co_NR_S 5	0.12	0.224	0.605	0.3	0.394	0.389	0.412	0.365	0.382	0.32
Co_R_D 25	0.292	0.295	0.53	0.362	0.589	0.461	0.18	0.208	0.398	0.332
Co_R_D 5	0.271	0.263	0.446	0.335	0.546	0.41	0.183	0.296	0.362	0.326
<i>Oracle</i>	1.559	1	1.623	1	1.453	1	1.5	1		
<i>Lower bound</i>	-0.94	-1	-0.898	-1	-0.726	-1	-0.9	-1		

Table 1: Performance of the sentence ranking methods on monolingual data. PB: profile-based ranker; Co: co-occurrence-based ranker; R/NR: Redundancy reduction enabled/disabled; D/S: ranking based on individual documents or sets of documents on the same topic of interest. The *Oracle* values are obtained using the gold-standard ranking, while the *Lower bound* values consider the inverted gold-standard ranking.

Topic	Avg. human score	RMSE
<i>IPC</i>	3.29	0.696
<i>G</i>	3.00	0.755
<i>M</i>	3.14	0.734
<i>SS</i>	2.89	0.712

Table 2: Average human score and regression error of the quality estimation approach.

datasets. The training dataset covers main news stories from September to October 2008, while the test set covers news commentaries on specific topics from 2005 to 2009.

6.2 Monolingual relevance ranking

The performance of the relevance ranking methods on the original, source-language texts is shown in Table 1. For the unsupervised co-occurrence ranking (Co), we run a number of experiments with different settings. We perform a greedy search on the number of dimensions to be used: 1, and 2%, 5%, 10%, 25% or 40% of the total. We run several experiments enabling (R) and disabling (NR) the sentence redundancy filter and on the full set of documents (S) and on a single document (D). We report here the settings that work the best across different topics. For the profile-based ranking (PB), based on our previous experience with this method, we chose to use the following numbers of words defining the profile vector: 1, 000, 2, 000 and 5, 000.

To define the gold-standard scores for the evaluation at sentence level, we use the number of annotators who selected the sentence as relevant (0-4). The results in Table 1 are the average performance

for all documents within a set of documents for each topic. They are compared against baselines proposed in (Kennedy and Szpakowicz, 2011):

- Inverse position (*InvPos*): each sentence is associated with the inverse of its position in the document. The ranking of the sentences thus corresponds to their position in the document and the inverse position is used as their relevance score.
- Sentence length (*Length*): each sentence is associated with the number of words that it contains. Longer sentences are deemed more informative.

The proposed baselines are highly competitive, in particular *Length*. This reflects the fact that longer sentences are naturally better candidates to be more informative, simply because they contain more words. Both methods in all settings outperform the *InvPos* ranker. Except for the **M** topic, most settings of the co-occurrence method and at least one setting of the profile-based method outperform *Length* according to $Avg\Delta_{all}$.

The last column of the Table shows that on average (all topics), the profile-based method seems to be slightly better suited for ranking the top 50% documents, with better $Avg\Delta_{all}$, while the co-occurrence-based method seems to be better for producing a global ranking of all sentences in the dataset, with better ρ coefficient. While the performances of the variations of the co-occurrence-based method seem to be highly dependent on the topic of the documents, it can be observed that on

	G		IPC		M		SS		Macro Av.	
	$Avg\Delta_{all}$	ρ	$Avg\Delta_{all}$	ρ	$Avg\Delta_{all}$	ρ	$Avg\Delta_{all}$	ρ	$Avg\Delta_{all}$	ρ
<i>Length</i>	0.593	0.272	0.886	0.259	2.075	0.512	0.365	0.089	0.981	0.283
<i>Length_QE</i>	0.853	0.28	1.02	0.258	2.156	0.518	0.5	0.096	1.132	0.288
Co-Tr_R_S 25	0.374	0.177	1.527	0.31	1.843	0.398	0.607	0.197	1.087	0.27
Co-Tr_NR_S 5	0.574	0.276	1.284	0.302	0.832	0.341	1.196	0.344	0.971	0.315
Co-Tr_NR_S 2	0.945	0.282	1.518	0.242	1.393	0.377	1.174	0.313	1.257	0.303
Co-Tr_R_D 25	0.834	0.217	1.577	0.323	1.668	0.44	0.99	0.246	1.267	0.306
Co-Tr_R_D 5	0.752	0.238	1.598	0.289	1.536	0.341	1.101	0.274	1.246	0.285
PB 1000	0.853	0.262	1.018	0.304	0.726	0.268	0.657	0.06	0.814	0.224
PB 2000	1.78	0.386	1.375	0.318	1.19	0.318	0.642	0.12	1.247	0.286
PB 5000	1.455	0.239	1.589	0.279	1.926	0.41	0.06	0.062	1.258	0.248
Co_R_S 25	0.728	0.327	1.521	0.299	1.768	0.405	0.665	0.222	1.171	0.314
Co_NR_S 5	0.443	0.198	1.494	0.275	1.262	0.361	0.947	0.349	1.037	0.296
Co_NR_S 2	0.981	0.241	1.121	0.23	0.944	0.369	1.383	0.34	1.108	0.295
Co_R_D 25	0.729	0.262	2.163	0.341	1.481	0.402	0.68	0.172	1.264	0.294
Co_R_D 5	0.77	0.21	1.326	0.317	1.344	0.384	0.534	0.23	0.994	0.286
<i>Oracle</i>	5.249	1	4.109	1	3.854	1	3.707	1		
<i>Lower bound</i>	-2.859	-1	-2.335	-1	-1.844	-1	-2.097	-1		

Table 3: Performance of the approaches combining informativeness and quality estimation for sentence ranking. Co-Tr: co-occurrence-based ranker applied directly to translated sentences; PB: profile-based ranker combined with quality estimates, Co: co-occurrence-based ranker applied to source texts and combined with quality estimates. R/NR and D/S as in Table 1.

average across different topics all these variations perform similarly.

We used the same methods - except the *InvPos* baseline, which clearly performs very poorly - and settings to assess the ranking of translated documents.

6.3 Relevance ranking for translated texts

We combine the translation quality and sentence ranking scores for each translated sentence t_i by taking their product:

$$score(t_i) = relevance(s_i) \times quality(t_i)$$

where $relevance(s_i)$ is given by either the co-occurrence (Co) or profile-based (PB) methods applied to the source language sentence s_i , and $quality(t_i)$ is given by the quality estimation method applied to the translation of s_i .

This is done for both the gold-standard annotation and the systems' predictions. The ranges of these two values are different, but this difference is not relevant, since we are only interested in the ranking of the sentences, as opposed to their absolute scores.

Using the product for combining scores is however not ideal: a translation with very low quality but high relevance can receive comparable scores as translations with high quality but low relevance. We have also experimented with using quality estimates as a filter for the relevance rankings. In other words, setting a threshold on the translation

quality scores below which a translated sentence is ranked at the bottom of the list even if its corresponding source is highly relevant. This strategy however was strongly affected by the choice of the threshold and resulted in generally poorer performance. Due to space constraints, we only present the results using the product of the two scores.

In the first set of experiments we evaluate the ability of our approach to rank translated sentences within a document. We combine the quality and the relevance scores at sentence level as explained above. As an alternative approach, we apply the unsupervised co-occurrence-based method (Co-Tr) to directly estimate the relevance of the translated text without any quality filtering. In this case, $score(t_i) = relevance(t_i)$. This approach does not explicitly address translation performance. Nevertheless, it can account for some translation problems implicitly, particularly words left untranslated or translated incorrectly. In all cases, the evaluation is performed comparing the system outputs against the combined (product) gold-standard. Results are shown in Table 3. The *Length* baseline is the same as in the monolingual setting and does not include the quality estimation filter. It is also compared against the combined gold-standard.

It is interesting to note that the quality estimation has a positive impact even for the baseline *Length_QE*, confirming that long sentences are often badly translated. The performance of most set-

tings of the co-occurrence and profile-based methods outperform the baselines, except for the **M** topic, as in the monolingual experiments. On average, the co-occurrence method on translated and source data provides better performance than the profile-based method in terms of ρ , while all methods are comparable according to $Avg\Delta_{all}$. This seems to indicate that the profile-based is good at ranking good quality informative sentences, but fails at ranking informative but poorly translated sentences. A possible reason is that it scores each sentence independently from the others and relies on the quality of the training data.

The best settings of the co-occurrence-based method applied to the source language texts outperform the best settings of the same method applied to translated texts. This is more evident in terms of $Avg\Delta$, as opposed to ρ . This seems to indicate that the combination strategy based on the product of the translation quality and relevance scores may not be the most appropriate for fine-grained ranking. Although the monolingual (Table 1) and cross-lingual (Table 3) results are not directly comparable because of their different upper and lower bounds (due to the different gold-standard values in each of these experiments), we can note similar trends with respect to the two ranking methods, Co and PB.

In the second set of experiments we assess the task of ranking documents within a set of documents on the same topic. To produce a unique score for each document, the sentence scores are scaled into $[0, 1]$ and averaged. Documents are then ranked according their average values within their respective groups. The same process is performed using the gold-standard scores and the κ is computed, as shown in Table 4.

The best scores of the proposed approaches vary from moderate to substantial. For the **G**, **IPC** and **M** topics, the best settings of the co-occurrence-based method on the source language outperform the baselines and is superior or equal the other methods. For the **SS** topic, the *Length* baseline is the best method. The co-occurrence method applied directly on the translated sentences is often as good as the two proposed methods that use the source language data. The co-occurrence methods on translated text can in fact be better for heterogeneous sets of documents such as **M**, but in general the usage of source language text can be beneficial.

Overall, the experiments in this paper show

	G	IPC	M	SS
<i>Length</i>	0.4	0.4	0.2	0.8
<i>Length_QE</i>	0.4	0.6	0.2	0.6
Co-Tr_R_S 25	0.6	0.4	0.4	0.6
Co-Tr_NR_S 5	0.8	0.0	0.4	0.4
Co-Tr_NR_S 2	0.4	0.0	0.4	0.2
Co-Tr_R_D 25	0.6	0.0	0.4	0.2
Co-Tr_R_D 5	1.0	0.4	0.0	0.4
PB 1000	0.8	0.4	0.2	0.0
PB 2000	0.8	0.6	0.0	0.0
PB 5000	0.8	0.6	0.2	0.2
Co_R_S 25	0.8	0.0	0.2	0.4
Co_NR_S 5	0.6	0.4	0.2	0.0
Co_NR_S 2	0.6	0.2	0.2	0.4
Co_R_D 25	0.2	0.0	0.4	0.2
Co_R_D 5	1.0	0.0	0.0	0.0

Table 4: Kappa coefficient of the various approaches combining informativeness and quality estimation for document ranking.

significant variations in performance for different methods and settings of the same method over different topics. We believe this is mostly due to the differences in the level of homogeneity of the documents within each topic. Nevertheless, if we consider only the average results over the four topics, we find that most methods/settings perform similarly. This average result however hides significant differences between the methods/settings and opens the way for future research into a better understanding of how to select the best methods and settings for different types of corpora.

7 Conclusions and future work

We have proposed combining source relevance information and translation quality estimates to rank translated sentences and documents within groups of texts on the same topic. The approach has shown promising results and it is potentially useful in different scenarios. These include applications where large numbers of documents with redundant information are clustered together according to certain criteria, for example, news on a given topic in media monitoring and news analysis applications, or reviews on a given product/service, and then machine translated to be published in other languages. In this scenario, it would be wise to select for publication only a subset of those documents whose translations are both relevant and of good quality. Additionally, the identification of relevant and high-quality sentences in documents can be used to highlight portions of a document that can be relied upon for gisting purposes, especially in cases where the reader does not have

access to the source document.

In future work, we plan to investigate better ways of combining the translation quality and relevance scores, as well as further investigate the effects of methods and settings on different topics.

References

- Bach, N., F. Huang, and Y. Al-Onaizan. 2011. Goodness: A Method for Measuring Machine Translation Confidence. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 211–219, Portland.
- Boudin, F., S. Huet, and J.M. Torres-Moreno. 2010. A graph-based approach to cross-language multi-document summarization. *Research journal on Computer science and computer engineering with applications (Polibits)*, 1:21–24.
- Callison-Burch, C., P. Koehn, C. Monz, and O. Zaidan. 2011. Findings of the 2011 workshop on statistical machine translation. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 22–64, Edinburgh, Scotland.
- Carlson, L., J.M. Conroy, D. Marcu, D.P. O’Leary, M.E. Okurowski, A. Taylor, and W. Wong. 2001. An empirical study of the relation between abstracts, extracts, and the discourse structure of texts. In *Proceedings of Document Understanding Conference*.
- Chang, C. and C. Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27–27. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- Cohen, J. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, April.
- Goldstein, J., M. Kantrowitz, V. Mittal, and J.G. Carbonell. 1999. Summarizing text documents: sentence selection and evaluation metrics. *Computer Science Department*, page 347.
- Gong, Y. and L. Xin. 2002. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of ACM SIGIR*, New Orleans, US.
- He, Y., Y. Ma, J. van Genabith, and A. Way. 2010. Bridging smt and tm with translation recommendation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 622–630, Uppsala, Sweden, July.
- Kennedy, A. and S. Szpakowicz. 2011. Evaluation of a sentence ranker for text summarization based on rogets thesaurus. In *Text, Speech and Dialogue*, pages 101–108. Springer.
- Mihalcea, R. 2004. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, page 20.
- Pouliquen, B., R. Steinberger, and C. Ignat. 2003. Automatic annotation of multilingual text collections with a conceptual thesaurus. In *Proceedings of the workshop Ontologies and Information Extraction at the EUROLAN’2003, Bucharest, Romania*.
- Savoy, J. and L. Dolamic. 2009. How effective is google’s translation service in search? *Communications of the ACM*, 52(10):139–143.
- Soricut, R. and A. Echiabi. 2010. Trustrank: Inducing trust in automatic translations via ranking. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 612–621, Uppsala, Sweden, July.
- Specia, L. and A. Farzindar. 2010. Estimating machine translation post-editing effort with hter. In *Proceedings of the AMTA-2010 Workshop Bringing MT to the User: MT Research and the Translation Industry*, Denver, Colorado.
- Specia, L. 2011. Exploiting objective annotations for measuring translation post-editing effort. In *15th Conference of the European Association for Machine Translation*, pages 73–80, Leuven, Belgium.
- Steinberger, J. and K. Ježek. 2004. Text summarization and singular value decomposition. In *Proceedings of the 3rd ADVIS conference*, Izmir, Turkey.
- Steinberger, J. and K. Ježek. 2009. Update summarization based on novel topic distribution. In *Proceedings of the 9th ACM Symposium on Document Engineering, Munich, Germany*.
- Steinberger, R., B. Pouliquen, and E. Van der Goot. 2009. An introduction to the europe media monitor family of applications. In *Information Access in a Multilingual World-Proceedings of the SIGIR 2009 Workshop (SIGIR-CLIR 2009)*, pages 1–8.
- Turchi, M., J. Steinberger, M. Kabadjov, and R. Steinberger. 2010. Using parallel corpora for multilingual (multi-document) summarisation evaluation. *Multilingual and Multimodal Information Access Evaluation*, pages 52–63.
- Wan, X., H. Li, and J. Xiao. 2010. Cross-language document summarization based on machine translation quality prediction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 917–926.
- Yeh, J.Y., H.R. Ke, and W.P. Yang. 2008. iSpreadRank: Ranking sentences for extraction-based summarization using feature weight propagation in the sentence similarity network. *Expert Systems with Applications*, 35(3):1451–1462.

Automatic Tune Set Generation for Machine Translation with Limited In-domain Data

Jinying Chen, Jacob Devlin, Huaigu Cao, Rohit Prasad,
and Premkumar Natarajan

Raytheon BBN Technologies

10 Moulton Street, Cambridge 02138, USA

{jchen, jdevlin, hcao, rprasad, pnatarajan}@bbn.com

Abstract

Many effective adaptation techniques for statistical machine translation crucially rely on in-domain development sets to learn model parameters. In this paper we present a novel method that automatically generates the matching tune set for Arabic-to-English MT with limited in-domain data¹. This technique improves our MT system over two baselines (tuned on data from the same domain but different genres) by 1.2 and 3.5 BLEU points using significantly less tuning data (1/6 and 1/2 of the baselines). Lexical and morphological features contribute to the success of our method in different ways. Generating tune sets using length distribution also improves the system significantly. Finally, our method obtains competitive results in experiments where in-genre tune sets are available.

1 Introduction

Adapting statistical machine translation (SMT) systems to different domains is a well-known and challenging problem. Many effective techniques developed for SMT adaptation crucially rely on in-domain development sets to learn model parameters or interpolation weights. For example, Koehn and Schroeder (2007); Ueffing *et al.*, (2007); Matsoukas *et al.* (2009); Foster *et al.*, (2010), to name a few. However, in some situations, in-domain data can be so limited and in a

few cases, no matching tune sets are available. Our problem falls into this category.

Most existing work in domain adaptation for SMT focused on language models, translation models, lexicons and parallel training data (Koehn and Schroeder, 2007; Lü *et al.*, 2007; Wu *et al.*, 2008; Matsoukas *et al.*, 2009; Foster *et al.*, 2010). Tune set adaptation shares a belief with other adaptation techniques that using training data similar to the test set (in domain, topic, and style) plays a critical role in SMT performance. A unique feature of this problem, however, is the high demands of matching quality. Many parameters in the SMT system are estimated using the tune set, so negative effects caused by noise (e.g., mismatch in topic and translation style) can be propagated easily. In fact, a large number of SMT domain adaptation techniques also adopted a general framework that requires a tune set to learn model parameters (Ueffing *et al.*, 2007) or interpolation weights for data from different domains (Koehn and Schroeder, 2007; Matsoukas *et al.*, 2009; Foster *et al.*, 2010).

In this paper we present a highly effective method that automatically generates matching tune sets for an Arabic-to-English MT task with considerably limited in-domain data. Our method is based on the nearest neighbor approach and a novel n -gram based similarity metric. It generates the tune set by extracting the nearest neighbors from a data set of mixed, different genres for each test segment. This method can be applied to any new test set because it only uses the source side of the test segments to find neighbors. Word based and morphological tag based features were used to capture different similarity patterns between neighbors. Compared with two baseline systems, which were tuned on the full data set and one of its subsets, the MT system tuned on the automatically generated tune set increased the BLEU scores by 1.2 and 3.5 points (29.66 vs. 28.43 and 29.66 vs. 26.11), respective-

© 2012 European Association for Machine Translation.

¹ This paper is based upon work supported by the DARPA MADCAT Program (Approved for Public Release, Distribution Unlimited). The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

ly. Furthermore, the tune sets generated by our method are much compact at only 1/6 and 1/2 the size of the baseline tune sets, respectively.

Further experiments suggested that both lexical and morphological features contributed to the effectiveness of this method. Length distribution is another important factor that affected performance. By using a length penalty score, our method naturally captured length distribution of the test set. Two comparative experiments with matching in-domain tune sets also obtained competitive results, which confirmed the robustness of our method. Another contribution of our work is to provide empirical evidence for various factors that impact tune set quality.

The rest of this paper is organized as follows. We reviewed related work in Section 2. In Section 3, we introduce our translation problem and the specific difficulty we faced. The similarity measure and features used by our tune set generation method are discussed in Section 4. In Section 5, we introduce the general techniques we used to adapt our MT system to the new domain. Experimental setup is described in Section 6 and experimental results and discussions are provided in Section 7. We conclude our paper in Section 8.

2 Related Work

Utiyama *et al.*, (2009) used a nearest neighbor-based approach to find optimal tune sets from a relatively large amount of in-domain parallel training corpora. Their method used the average of BLEU-1 to BLEU-4 scores to measure segment-level similarity. It outperformed a random sampling-based baseline by over 2 points in BLEU. Unlike their work, we developed a new similarity metric by observing the “*bias nature*” of BLEU in measuring segment-level similarity. Our experiments showed that our method was more effective in finding the matching tune set.

Hui *et al.* (2010) described the strategy for choosing the best tune set from a list of available in-domain tune sets based on their similarities to the test set (measured by a modified BLEU score). Unlike their work, we constructed the tune set from scratch by using a segment-level similarity measure.

Apart from tune set sampling and selection techniques mentioned above, some attempts have been made in sampling parallel training data. Lü *et al.*, (2007) used the nearest neighbor-based method to generate a compact parallel training corpus that matched the test and tune sets. They used the standard TF-IDF weighting scheme to

measure segment-level similarity. They observed that, over a threshold (1000 in their case) of the number of neighbors used, the MT performance would drop due to noisy data included. We observed similar phenomena in our experiments (as discussed in Section 7.3) but the threshold was much lower ($=2$ in our case). This suggests that accurate matching is more demanding in tune set generation than in training set generation

3 Problem Setting

Our task is Arabic-to-English translation on image text from the field (legal filings, *etc.*), which we will refer to as the *Field Document* domain. This task has limited in-domain data, with 0.4M translated words in total. We have a state-of-the-art MT system trained on a large amount of out-of-domain data, including 50M words of news-wire and web bilingual data and 9 billion words of English text (to train the language model). This is a typical domain adaptation problem.

A specific difficulty we faced in this task, however, is that the small size in-domain data was further divided into three genres: handwritten (HW), machine print (MP) and mixed-form (MX). The three genres have overlap in topics but are quite different in style. HW data are mainly fluent text and long sentences; MP data were extracted from printed forms and are mainly short phrases or segmented (diffluent) text; MX data were extracted from different forms with both printed and handwritten text, and are a more balanced mixture of fluent and diffluent text. Genre information was given at both document-level and segment level. On average, an HW document has over 95% HW segments, an MP document has over 85% MP segments, and an MX document is more balanced, but still has over 65% MX segments². It was required to report translation scores on each genre at document-level separately. Furthermore, the document distribution for these three genres is extremely unbalanced: 1929 HW documents, 590 MX documents and only 68 MP documents. Since MP data was so limited, we reserved all of them as the MP test set to ensure the reliability of the testing results.

In sum, our task is to build MT systems for three genres with limited in-domain data, one of which is completely missing its genre-matched training data.

² The MX segments cannot be automatically divided into handwritten and printed parts for translation purpose.

4 Nearest Neighbor Based Automatic Tune Set Generation

To automatically generate the matching tune set for the MP data, we used a nearest neighbor approach which was inspired by Utiyama *et al.* (2009). However, we developed a novel similarity metric and exploited different n -gram features, which we believe better fit our problem. This was confirmed by our experimental results.

4.1 Similarity Metric

We defined a similarity metric that looks like BLEU (Papineni *et al.*, 2002) but is significantly different in nature.

$$match_i(c, t) = \frac{\sum_{\{n_i \in c \cap t\}} \min(count_c(n_i), count_t(n_i))}{\sum_{\{n_i \in t\}} count_t(n_i)} \quad (1)$$

$$sim(c, t) = -\frac{|len(c) - len(t)|}{len(t)} + \frac{1}{N} \sum_{i=1}^N \log(match_i(c, t)) \quad (2)$$

where t is a given test segment and c is the candidate segment. n_i is any i order n -gram. $count_x(n_i)$ is the number of occurrences of n_i in segment x . $match_i(c, t)$ looks like the precision score of BLEU but we treat c as the “reference” and t as the “hypothesis”³. So unlike BLEU precision, this score is not affected by the length of a candidate segment.

$len(x)$ is the number of occurrences of l -gram’s in segment x . $-\frac{|len(c) - len(t)|}{len(t)}$ is the length

penalty score, which penalizes the longer and shorter candidates equally. $sim(c, t)$ is the similarity measure which combines the length penalty score and the n -gram matching scores in a way similar to BLEU. N is the highest order of n -grams used ($N=4$ in our case).

The major difference between our measure and BLEU is that it uses only a symmetric length penalty score to enforce length matching, while BLEU relies on its precision score to penalize longer hypotheses and a non-symmetric length penalty score to penalize shorter ones. Simple mathematical calculation shows that BLEU, by its nature, favors longer hypotheses (i.e., candidate segments) than shorter ones when they have equal numbers of overlapping n -grams with the

³ In practice, we omit the denominator of this item when ranking neighbors for a given test segment.

reference (the given test segment) and their length distances from the test segment are equal. This bias is not a big issue when measuring similarity among blocks of text but can be a problem when measuring segment-level similarity. This is why we designed a new similarity metric that handles length penalty in a different way.

Since it is likely to get zero-valued $match_i$ ’s at segment level, which will make their log values negative infinite, we use a non-parametric approach to smooth our n -gram matching measure by adding 1 to the numerator and denominator of equation (1), as in equation (1)’.

$$match_i(c, t) = \frac{1 + \sum_{\{n_i \in c \cap t\}} \min(count_c(n_i), count_t(n_i))}{1 + \sum_{\{n_i \in t\}} count_t(n_i)} \quad (1)'$$

We compared the length distribution of the tune sets generated by our method and by a BLEU-based similarity measure (Utiyama *et al.*, 2009). As shown in Fig. 1, the length distribution curve generated by our method (**MP-AG**, grey solid line) had less fluctuations than that generated by **BLEU** (black dotted line), compared with the curve of the MP test set (**MP-test**). Though length distribution is only one factor that impacts MT performance (to be discussed in Section 7.2), it gives us a clue that our measure is likely to achieve better MT performance (which was confirmed by our MT experiments).

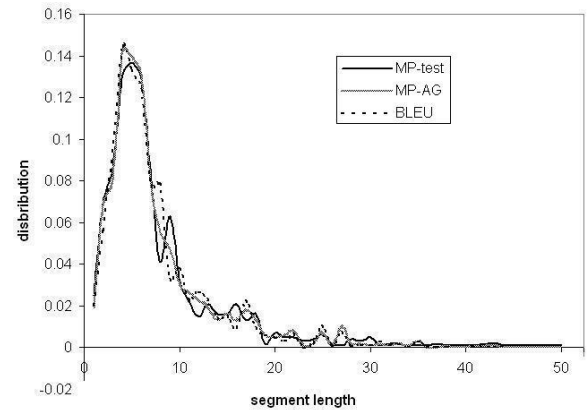


Figure 1. Length Distributions of the MP test set (MP-test), the tune sets generated by our method (MP-AG) and generated by BLEU

4.2 N-gram Features

We used two types of n -gram features: lexical based and morphological based.

Lexical n -grams are strong indicators for text similarity, which were used by many previous

works to measure segment-level or data set level similarity (Lü *et al.*, 2007; Utiyama *et al.*, 2009; Hui *et al.*, 2010). Intuitively, lexical 1-gram’s and 2-gram’s are good indicators of topical similarity and higher order n -grams ($n>2$) are more responsible to capture similarity in styles. We extracted lexically-based n -grams from the source side (Arabic text without tokenization) of each bilingual sentence-pair.

One issue with using lexical n -grams is that only exact matches are counted. In order to have a more generalized model, we also compute the similarity score using morphological tags. Arabic is a morphological rich language, so its morphological tags hopefully can provide us a good balance between accuracy and generalization.

We used Sakhr Morphological Analyzer, a proprietary rule-based software, to generate the morphological tags for Arabic. The Sakhr tags are similar to English part-of-speech tags but have richer information about a word. For example, a tag for an Arabic verb may indicate tense, number, gender and voice. We kept all this information in a tag (i.e., did not generalize further) when matching morphological n -gram’s.

Though less accurate, the generalization helps to capture more aspects in style similarity. For example, many MP segments contain names, dates and numbers. By using morphological features, our method can discover segments that share the same sentence structures with an MP segment but do not necessarily contain the same names or numbers.

We used these two types of features independently. That is, we always find $2 \times n$ ($n=1$ in our experiments) nearest neighbors, n by lexical features and n by morphological features.

4.3 Treatment of Duplicate Neighbors

Since the nearest neighbors were extracted for each test segment, the resulting tune set had duplicate segments. We kept duplicate instances because the number of duplicates naturally reflected to which degree a selected segment fit the whole test set. In the real implementation, we refined our MT system to support segment level weighting for tune sets. That is, we used the non-duplicate tune set with its segments weighted by the number of their duplicates. This sped up the training procedure, especially when there were many duplicates in the tune set or the system need to be tuned for much iteration. In Section 7, we only report the size of non-duplicate tune sets for all the experiments.

5 MT System Description

5.1 Baseline MT System

We used a state-of-the-art hierarchical decoder in our experiments (Shen *et al.*, 2008). The features it uses in decoding and n-best rescoring includes a small set of linguistic and contextual features, such as word translation probabilities, rule translation probabilities, language model scores, and target side dependency scores. In addition, it uses a large number of discriminatively tuned features, similar to those described in (Chiang *et al.* 2009). The system used a 3-gram language model (LM) for decoding and a 5-gram LM for rescoring. Both LMs were trained on billions of words of English text in news and web blogs. Feature scores are combined with a log-linear model. The feature weights were set by optimizing the BLEU score on the tune set.

5.2 Domain Adaptation

The general framework we used to adapt our baseline MT system to the new domain follows the line of Koehn and Schroeder (2007). We trained a separate language model using the target side of our in-domain parallel training data, and discriminatively estimated the interpolation weight with the standard language model. To adapt the translation model, we discriminatively estimate separate feature weights and penalties for rules extracted from the in-domain and out-of-domain parallel training.

This adaptation procedure improved the results on the HW test set by 8 points of BLEU and TER (see Table 1). We used this system in all the experiments on tune set generation.

Condition	BLEU	TER
Train: News Tune: News	19.99	61.58
Train: News+Field Tune: Field	28.23	53.33

Table 1. Baseline scores before/after adaptation

It is worth noting that the MT systems we developed will be applied on the output from a state-of-the-art optical character recognition (OCR) system. Because the OCR errors usually reduce MT performance significantly, we only used the transcribed text to develop our MT system and applied the final system on the OCR output with all the system parameters fixed. Therefore, we reported our experimental results mainly on the transcribed text, except that we

provided the MT performance scores on the OCR input of the MP genre in order to show the gain from using our method was applicable to the noisy input from OCR.

6 Experimental Setting

6.1 Data Sets

As introduced in Section 3, in our problem, there was limited in-domain data in the *Field Document* domain and the document distribution for the three genres was unbalanced.

We reserved all 68 MP documents for the MP test set. To create a tune set for this genre, we randomly picked MP-labeled segments from 153 HW documents and 229 MX documents. This formed the first baseline in our automatic Tune set generation experiments. The second baseline was the single big tune set by merging the MP, HW and MX tune sets (called ALL-tune).

The test and tune sets for the MX and HW genres were randomly picked documents with the same genre labels. The remaining documents were used as the parallel training data for extracting in-domain translation rules and training the in-domain LMs. Table 2 summarizes our data set division.

Data Set	Num of segments	Source
MP-test	1,093	MP
HW-test	3,150	HW
MX-test	2,400	MX
MP-tune	1,876	MX,HW
HW-tune	2,730	HW
MX-tune	2,522	MX
All-tune	7,091 ⁴	HW, MX
Parallel-training	25,864	MX,HW

Table 2. In-domain Data Division

We used the same parallel training data in all the experiments described in this paper to compare the pure effects from different tune sets. In practice, after we determine the specific tune set for each genre, we can add all the unselected data to parallel training to maximize the gains.

6.2 Experimental Conditions

In the MP experiments, we compared MT performance using our method (**Auto-Gen**) with the following baseline conditions:

- **MP**: MP-tune
- **ALL**: All-tune
- **BLEU-1**: tune set extracted from All-tune by duplicating the method described in (Utiyama *et al.*, 2009); use lexical features only
- **BLEU-2**: same as BLEU-1; use both lexical and morphological features

To separate various factors that impact the effectiveness of our method, we compared four **Auto-Gen** conditions where the segment-level similarity was measured in different ways:

- **Len**: only use the length penalty measure in Eq. 2 to measure the segment similarity
- **Len+Lex**: use the full Eq. 2 but only use lexical based n -grams
- **Len+Mrf**: use the full Eq. 2 but only use morphological based n -grams
- **Len+Lex+Mrf**: our complete method

We also tested our method on the other two genres in the three conditions similar to the MP experiments: **Auto-Gen**, **HW/MX**, **ALL**. However, because HW-tune and MX-tune are in the same genre as their test sets, they are actually *upper-bound* in some sense rather than baselines. **ALL** is also harder to beat because 1/3~2/5 tune-ALL segments are from the same genre as the HW (or MX) test set. Nevertheless, the results on these two genres can add evidence on how well our method works.

7 Results

Table 3 showed the results on the MP test set using automatic tune set generation. The system using our complete method (**Len+Lex+Mrf**) outperformed the system tuned on the MP tune set (**MP**) by 3.5 points in BLEU and 3 points in TER. Furthermore, the automatically generated tune set was more compact, with its size only about half of the MP tune set. Compared with using all the tuning data (**ALL**), our method achieved 1.2 points gain in the BLEU score and 0.9 point gain in TER. This gain was also significant, especially when considering that it only used about 1/6 of all the tuning data.

Surprisingly, MT performance using the MP tune set (**MP**), which was composed of MP segments from the HW and MX genres, was significantly lower than using all the tuning data (**ALL**). Further data analysis suggested that the unmatched length distribution between the MP tune set and the MP test set and the low vocabulary

⁴ The MP-tune and HW-tune sets have a small portion of overlap, so the number of segments in All-tune is slightly different from the sum of MP-tune, HW-tune and MX-tune.

coverage were the two culprits for the performance drop. The lesson learned here is we should not fully trust segment-level genre labels to find a matching tune set. We will discuss this in greater details in Sections 7.1 and 7.2.

We further compared our method with the method (**BLEU-1**) as described by Utiyama *et al.* (2009). They used the averaged BLEU- i scores ($i = 1, 2, 3, 4$) to measure segment-level similarity and extracted 2 nearest neighbors for each test segment. The results showed that our method performed better, with 1.5 point gain in BLEU and 1.2 point gain in TER.

To verify the appropriateness of the various considerations we had in designing our similarity measure, we compared our method with another method (**BLEU-2**) that used the averaged BLEU- i scores ($i = 1, 2, 3, 4$) as the similarity measure and used the same lexical and morphological n -gram features as ours. Compared with **BLEU-2**, our method had 1.1 point gain in BLEU and the same TER value. **BLEU-2** is better than **BLEU-1** (0.54 point gain in BLEU and 1.2 point gain in TER), suggesting that using morphological features is helpful. We will have more discussions in this aspect in Section 7.2.

Tune Set	Num Segs	BLEU	TER
MP	1,876	26.11	54.81
ALL	7,091	28.43	52.76
Len+Lex+Mrf	1,081	29.66	51.82
BLEU-1	1,168	28.03	53.04
BLEU-2	1,084	28.57	51.89

Table 3. MT Performance on Transcribed MP Test Set Using Different Tune Sets

Further experiments confirmed that the MT system developed on the automatically generated MP tune set achieved consistent gains on the input with OCR errors (word error rate=9.4%), as shown in Table 4.

Tune Set	BLEU	TER
MP	24.26	57.60
ALL	26.24	56.12
Len+Lex+Mrf	27.11	55.23

Table 4. MT Performance on OCR output of MP Test Set by Using Different Tune Sets

7.1 Effect of Length Distribution

To investigate the significant performance drop by using the MP tune set, we compared the segment-level length distribution of this set and the

MP test set. The difference was obvious (see Fig. 2, dotted line vs. black solid line). In contrast, the length distribution of the in-genre tune sets for the HW and MX data matched their test sets well (we omit the figures here due to space limits). This suggests that the MP segments from the HW and MX documents are significantly different from the MP test data.

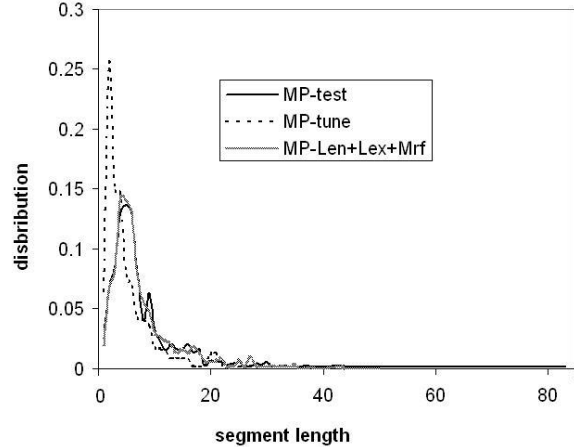


Figure 2. Length Distributions of MP-test, MP-tune and the tune set generated by Auto-Gen (Len+Lex+Mrf)

Intuitively, length distribution is a good indicator for the style of text from different sources. Therefore keeping similar length distribution is essential to getting a good matching tune set. Our similarity measure (as defined in Eq. 2) used a length penalty score to enforce length similarity among neighbors. The length distribution of the tune set generated by our method fit that of the MP test set very well (Fig. 2, grey line vs. black solid line). To separate the effect of this factor from other factors like n -gram based matching, we compared our method with a method that used only the length similarity (or penalty) scores to rank the neighbors of a test segment. For a fair comparison, we extracted two nearest neighbors for a test segment in both methods. If a test segment has more than two equally-nearest neighbors measured by length, we randomly picked two segments from them.

As expected, the length distribution of the tune sets generated by length-based sampling fit the MP test set well. The MT experiments (Table 5) showed that using the length similarity itself (**Len**) improved system performance by 1.4 points of BLEU and 1.1 points of TER scores over the **MP** baseline, but still significantly worse than using our complete method (**Len+Lex+Mrf**). These results suggest that

modeling length distribution is useful but by itself won't guarantee to find the best tune set.

Tune Set	Num Segs	BLEU	TER
MP	1,876	26.11	54.81
Len	1,338	27.58	53.79
Len+Lex+Mrf	1,081	29.66	51.82

Table 5. Effect of Length Distribution on Finding Matching Tune Sets

7.2 Lexical vs. Morphological N -grams

To separate the contributions from the lexical and morphological n -gram features, we compared our method with two other methods that used the same similarity measure but used only the lexical or the morphological features. The results (Table 6) showed that neither type of features (**Len+Lex** or **Len+Mrf**) was as effective as their combination (**Len+Lex+Mrf**) in improving the MT scores, though they all outperformed the **MP** baseline.

As discussed in Section 4.2, the lexical n -grams are expected to characterize topical similarity to a greater degree than the morphological features. To estimate the topical similarity among different data sets, we compared the out-of-vocabulary (OOV) rates⁵ (against the MP test set) of the tune sets generated by the above three methods. As shown in Table 6, the tune set generated by lexical n -gram matching had smaller OOV rate than morphological n -gram matching (36.34 vs. 36.84). Combining them reduced the OOV rate by over 4 percent to 32.18. The higher OOV rate of the MP tune set (45.51) further suggests that this set is less similar to the MP test set.

7.3 Effect of Increasing Neighbors

Given that **Len+Lex** was better than **Len+Mrf** in both the OOV rate and the MT performance, one may question if using only lexical features and 2 nearest neighbors will be better. In fact, this method (**Len+2Lex** in Table 6) was worse than **Len+Lex**, though it had a lower OOV rate. One possible reason is the noise introduced by using more, but less similar, neighbors. Further experiments (comparing $2 \times n$, $n=1, 2, 3, 5, 10$, nearest neighbors) showed that using more neighbors decreased the MT performance (Table 7). This result suggests a trade-off between precision (accurate matching) and recall (enlarging

⁵ The OOV rate numbers are high because the lexicons generated from the tune sets (several thousand segments) are small.

the vocabulary). Unlike training corpora creation where increasing vocabulary coverage had a privileged priority (Biçici and Yuret, 2011), accurate matching (similarity) is more important for tune set generation.

Tune Set	OOV (%)	Num Segs	BLEU	TER
MP	45.51	1,876	26.11	54.81
Len+Lex	36.34	682	28.16	52.99
Len+Mrf	36.84	616	27.18	52.70
Len+Lex+Mrf	32.18	1,081	29.66	51.82
Len+2Lex	32.98	1,176	27.32	52.62

Table 6. Effect of Lexical vs. Morphological Features on Finding the Matching Tune Sets

n	1	2	3	5	10
OOV	32.18	29.44	28.09	25.94	24.80
BLEU	29.66	28.78	28.31	27.99	27.75

Table 7. Effect of Increasing Neighbors (each experiment used $2 \times n$ nearest neighbors)

Tune Set	Num Segs	BLEU	TER
HW	2,730	28.23	53.33
ALL	7,091	28.09	52.85
Random	2,369	27.09	53.11
Len+Lex+Mrf	2,350	27.65	52.98

Table 8. MT Performance on Transcribed Handwritten Test Set Using Different Tune Sets

7.4 Experiments on HW and MX Test Sets

We also applied our tune set generation method on the HW and MX data. The results showed that the HW tune set (**HW**) outperformed our method (**Len+Lex+Mrf**) by 0.6 point BLEU and 0.4 point TER (Table 8) and the MX tune set (**MX**) outperformed by 0.9 point BLEU and 0.6 point TER (Table 9). The within 1 point performance drop was acceptable since the HW and MX tune sets, which were randomly picked from the same genres as their test sets, were similar to their test sets already (measured by the length distribution and the OOV rates). Comparing with the randomly generated tune sets (**Random**) in the same size, our method improved the MT performance by 0.6 BLEU points on the HW test set and 0.7 BLEU points on the MX test set.

Comparing with using all the tuning data (**ALL**), our method achieved close performance (within 0.1~0.4 points in BLEU and TER) while using much less data (1/4~1/3). The total amount

of CPU time required to run tuning is thus reduced to 1/4~1/3 of the original cost, since this time is directly proportional to the size of the tune set. The time required to run our tune set selection procedure is well over 100x faster than the tuning itself, so it is not a significant factor in the total run time. This added further evidence to the robustness and effectiveness of our method.

Tune Set	Num Segs	BLEU	TER
MX	2,522	37.05	42.95
ALL	7,091	36.48	43.71
Random	1,808	35.44	44.41
Len+Lex+Mrf	1,790	36.12	43.52

Table 9. MT Performance on Transcribed Mixed Test Set Using Different Tune Sets

8 Conclusions

This paper presents a novel method to automatically generate matching tune sets for MT tasks with limited in-domain data. With this method our MT system achieved significantly better performance (measured by BLEU and TER scores) than two baseline systems using significantly less tuning data. The performance gains were consistent on input text with OCR errors. This method also achieved competitive results on two other MT tasks with in-genre tune sets. In addition, we provide empirical evidence that length distribution modeling, lexical and morphological n -gram matching are all important factors contributing to the success of our method. They were able to capture topical and style similarities in different ways. We also showed that, compared with parallel training data extraction and generation, precision (accurate matching) was more important than recall (increasing vocabulary coverage). In the future, we hope to extend this method to training data creation for MT with limited in-domain data in an active learning framework.

References

Ergun Bici and Deniz Yuret. 2011. Instance selection for machine translation using feature decay algorithms. In *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, Edinburgh, England, July.

David Chiang, Kevin Knight, and Wei Wang. 2009. 11,001 new features for statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL*, pages 218–226.

Hui Cong, Zhao Hai, Lu Bao-Liang, and Song Yan. 2010. An empirical study on development set selection strategy for machine translation learning. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 42–46, Uppsala, Sweden, July. Association for Computational Linguistics.

George Foster and Roland Kuhn. 2009. Stabilizing minimum error rate training. In *Proceedings of the 4th Workshop on Statistical Machine Translation(WMT)*, Boulder, Colorado, USA.

Philipp Koehn and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 224–227.

Yajuan Lü, Jin Huang, and Qun Liu. 2007. Improving statistical machine translation performance by training data selection and optimization. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 343–350.

Spyros Matsoukas, Antti-Veikko I. Rosti, and Bing Zhang. 2009. Discriminative corpus weight estimation for machine translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Singapore.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Libin Shen, Jinxi Xu, and Ralph Weischedel. 2008. A new string-to-dependency machine translation algorithm with a target dependency language model. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 577–585.

Nicola Ueffing, Gholamreza Haffari, and Anoop Sarkar (2007). Transductive learning for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, 25–32.

Masao Utiyama, Hirofumi Yamamoto, and Eiichiro Sumita. 2009. Two methods for stabilizing MERT: NICT at IWSLT 2009. In *Proceedings of International Workshop on Spoken Language Translation(IWSLT)*, Tokyo, Japan.

Hua Wu, Haifeng Wang, and Chengqing Zong. 2008. Domain adaptation for statistical machine translation with domain dictionary and monolingual corpora. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 993–1000. Manchester, USA.

Domain Adaptation in SMT of User-Generated Forum Content Guided by OOV Word Reduction: Normalization and/or Supplementary Data?

Pratyush Banerjee, Sudip Kumar Naskar, Johann Roturier¹, Andy Way², Josef van Genabith

CNGL, School of Computing, Dublin City University, Dublin, Ireland

{pbanerjee, snaskar, josef}@computing.dcu.ie

¹ Symantec Limited, Dublin, Ireland

johann.roturier@symantec.com

² Applied Language Solutions, Delph, UK

andy.way@appliedlanguage.com

Abstract

This paper reports a set of domain adaptation techniques for improving Statistical Machine Translation (SMT) for user-generated web forum content. We investigate both normalization and supplementary training data acquisition techniques, all guided by the aim of reducing the number of Out-Of-Vocabulary (OOV) items in the target language with respect to the training data. We classify OOVs into a set of types, and address each through dedicated normalization and/or supplementary training material selection-based approaches. We investigate the effect of these methods both in an additive as well as a contrastive scenario. Our findings show that (i) normalization and supplementary training material techniques can be complementary, (ii) for general forum data, fully automatic supplementary training data acquisition can perform as well or sometimes better than semi-automatic normalization (although tackling different types of OOVs) and (iii) for very noisy data, normalization really pays off.

1 Introduction

Web-forums are rich sources of user-generated content on the web. The increasing popularity of technical forums have motivated major IT companies like Symantec to create and support forums around their products and services. For individual users or larger customers, such forums provide an easy source of information and a viable alternative to traditional customer service options. Being a

multinational company, Symantec hosts its forums in different languages (English, German, French etc), but currently the content is siloed in each language. Clearly, translating the forums to make information available across languages would be beneficial for Symantec as well as its multilingual customer base. This forms the primary motivation of techniques presented here.

Despite growing interest in translation of forum data (Flournoy and Rueppel, 2010), to date, surprisingly little research has actually focussed on forum data translation (Roturier and Bensadoun, 2011). Compared to professionally edited text, user-generated forum data is often more noisy, taking some liberty with commonly established grammar, punctuation and spelling norms. For our research, we use translation memory (TM) data from Symantec, which is part of their corporate documentation, professionally edited and generally conforming to the Symantec controlled language guidelines. On the other hand, our target data (forum) is only lightly moderated and does not conform to any publication quality guidelines. Hence despite being from the same IT domain, there is a significant difference in style between the training and the test data. In this paper, we focus our efforts on systematically reducing this difference through the use of both normalization and supplementary training material acquisition techniques.

Our research was conducted on English to German (En-De) and English to French (En-Fr) language directions. To identify the differences between the TM and forum data, we focus on the OOV words in the English forum data with respect to the source side (English) of the TM data. We classify OOVs into different categories which require independent attention. In order to optimally handle each individual category, different tech-

niques were developed to make the forum-based test sets better resemble the training data. For the *first category* – containing tokens such as URLs, paths, registry entries, and memory addresses – regular expressions were used to capture the tokens and replace them with unique place-holders. The *second category* included valid words inadvertently fused by punctuation characters (especially ‘.’) which required a training data-guided splitting technique. The *third category* comprising spelling errors were handled by an off-the-shelf automatic spell checker. Additionally the spell checker was trained with ‘in domain’ data to make it aware of the domain-specific terms to improve the quality of spell checking. For the *fourth category* of OOVs – valid words not occurring in the training data – various supplementary ‘out-of-domain’ bitext training data were automatically searched. For every OOV in this category, parallel sentence pairs from different ‘out-of-domain’ data were added to the ‘in-domain’ training data to improve the coverage of the translation models.

While improving translation quality by reducing OOVs is the primary objective of our research, we are particularly interested in the effect of spell checking on translation quality of forum data with various degrees of noise. Furthermore, we compare the relative improvements provided by the normalization to supplementary data selection to justify the effectiveness of the respective techniques. The rest of the paper is organized as follows: Section 2 briefly reviews relevant related work. Section 3 provides a detailed discussion on the normalization techniques as well as the acquisition of supplementary training material. Section 4 presents the datasets and the experiments and corresponding results, followed by our conclusions and pointers to future work in Section 5.

2 Related Work

The technique of using ‘out-of-domain’ datasets to supplement ‘in-domain’ training data has been widely used in domain adaptation of SMT. Information retrieval techniques were used by Eck et al. (2004) to propose a language model adaptation technique for SMT. Hildebrand et al. (2005) utilized this approach to select similar sentences from available bitext to adapt translation models, which improved translation performance. Habash (2008) used spelling expansion, morphological expansion, dictionary term expansion and proper name

transliteration to enhance or reuse existing phrase table entries to handle OOVs in Arabic–English MT. More recently an effort to adapt MT by mining bilingual dictionaries from comparable corpora using untranslated OOV words was carried out by Daume III and Jagarlamudi (2011).

Our current line of work is related to the work reported in Daume III and Jagarlamudi (2011) and that of Habash (2008). In our case, however, the target domain (web-forum) is different from the training data (Symantec TMs) more in terms of style rather than actual domain (Banerjee et al., 2011). Secondly, in contrast to mining comparable data for bilingual dictionary extraction (Daume III and Jagarlamudi, 2011), we exploit sentence pairs from available parallel training data to handle untranslated OOVs. Moreover, mining supplementary parallel data guided by OOVs is used as a technique complementing the normalization-based approaches to reduce specific types of OOVs in the target domain. We classify OOVs into different categories and treat each of them separately. In contrast to extending the phrase table entries (Habash, 2008) our normalization methods mostly comprise pre- and post-processing techniques. Finally we also present a comparison between the normalization and supplementary training data acquisition techniques for different error density-based scenarios of the target domain. To the best of our knowledge, the use of ‘domain-adapted’ spell checkers to reduce OOV rates in the target domain is novel, and is one of the other main contributions of the paper.

3 Normalization and Supplementary Data Selection Techniques

This section introduces the datasets used for the experiments followed by the adaptation techniques used in the experiments.

3.1 Datasets

The primary training data for our experiments consisted of En–De and En–Fr bilingual datasets in the form of Symantec TMs. Monolingual Symantec forum posts in German and French along with the target side of the TM training data served as language modelling data. In addition, we also had a collection of posts from the original Symantec English forums acquired over a period of two years which formed the basis of our OOV category estimation. The development (dev) and test sets used

in our experiments were randomly selected from this particular data set. Table 1 reports the amount of data used for all our experiments.

	Data Set	En-De	En-Fr
Bi-text	Symantec TM	832,723	702,267
	Development Set	500	500
	Test 1	2,022	2,022
	Test 2	600	600
Monolingual	English Forum	1,129,749	
	German Forum	42,521	
	French Forum	41,283	

Table 1: Number of Sentences for training, development and test sets, and forum data sets

As reported in Table 1, we used two different test sets, for our experiments. The first one (Test-1) was randomly chosen from the English forum data. Since one of our objectives was also to investigate a scenario with a high density of spelling errors, typical for some forum posts, the second test set (Test-2) was selected to simulate a higher proportion of noise (approximately one spelling error in every two test set sentences). This was achieved by flagging the remaining forum dataset (after removing the Test-1 sentences), using an automatic spell checker, and randomly selecting sentences with spelling errors followed by a manual review. Both these test sets were manually translated following basic guidelines for quality assurance. The randomly chosen dev set was translated using Google Translate,¹ and manually post-edited by professional translators following guidelines² for achieving ‘good enough quality’.

3.2 OOV Categorization

The remaining (after dev and test set selection) English forum data, comprising over 1.13M sentences (around 17.5M words), were used to compute OOV words in the forum domain with respect to the training data, using a unigram language model estimated on the source side of the training data. Manual inspection of the OOV word list identifies the following general categories:

1. Maskable Tokens (MASK): URLs, paths, registry entries, email addresses, memory locations, date and time tokens and IP addresses or version numbers.
2. Fused Words (FW): Two or more valid tokens concatenated using punctuation marks like ‘.’.

¹<http://translate.google.com/>

²<http://www.translationautomation.com/machine-translation-post-editing-guidelines.html>

3. Spelling Errors (SPERR): Spelling errors or typos.
4. Valid Words (VAL): Valid words not occurring in the training data.
5. Non-Translatable (NTR): Tokens comprising standalone product and service names and numbers (not part of Category-1 tokens) which ideally should not be translated.

Table 2 depicts the percentage of the OOV word categories in the English forum data and the two test sets with respect to the En-De and En-Fr TM-based source data sets. Comparing the category-wise percentage figures on the two test sets (Test-1 and Test-2) clearly show the distribution of the categories in Test-1 is similar to that of the original Forums. Test-2 shows a higher percentage of SPERR tokens as it had been consciously designed to have high spelling error density. The figures also depict the relative importance of the specific OOV categories in forum-style data, with non-translatable (NTR) and maskable tokens (MASK) covering nearly 75% of the OOV range.

OOV Type	En-De			En-Fr		
	Forum	Test-1	Test-2	Forum	Test-1	Test-2
MASK	25.68	21.33	9.93	25.47	19.43	9.83
FW	8.89	4.11	2.05	8.75	3.71	2.00
SPERR	10.41	12.64	52.91	10.45	12.29	52.67
VAL	6.38	14.06	12.33	6.74	18.86	12.17
NTR	48.64	47.87	22.77	48.60	45.71	23.33

Table 2: Category-based percentage of OOVs in the English forum and two test data sets

Different normalization techniques used to independently address each of these OOV categories are detailed below.

3.3 Regular Expression-based Normalization

For the normalization of MASK OOVs we developed a set of regular expressions to identify tokens. These were replaced with unique placeholders. These replacements were then applied uniformly over all data sets (TM and forum) in a pre-processing step. Most of the tokens in this category were multi-word tokens, and this method allowed them to be treated as single tokens during the translation process. This not only helped in maintaining the internal ordering of words within such tokens but also ensured that none of the terms within such a token were translated.

3.4 Fused Word Splitting

To handle FW tokens which comprise two or more valid words fused using a period (‘.’) symbol, we

identified all tokens which had a period symbol flanked by alphabetic characters. However, since a large number of valid file names, website names or abbreviations (e.g. N.I.S., explorer.exe, shopping.aol.com, etc.) were also identified, we used heuristics based on the training data to identify the valid ones. Lists of known file extensions (e.g. exe, jar, pdf, etc.) and website domain extensions (e.g. com, edu, net, gov, co.uk, etc.) were used to filter out file names and website names. Finally we used a dictionary built on the training data. Every split was validated against the dictionary, with the constraint that all its constituent splits had to occur in this dictionary. This normalization was only applied on the dev and test sets as the TM training data was assumed to be clean of such fused words.

3.5 spell checker-based Normalization

A considerable amount of the OOVs in the unnormalized forum data comprise spelling errors or typos (SPERR). We used an off-the-shelf spell checker (cf. Section 4.2) to identify and correct these tokens so that they mapped to valid words (preferably in the training data). While the ready-to-use spell checker worked well for most of the spelling errors in general-purpose English words, it flagged a lot of ‘in-domain’ (technical) words. Hence we adapted the spell checker to the domain. This was achieved by generating glossary lists from the source side of the TMs and adding them to the spell checker dictionary. Furthermore, the spell checking models had to be retrained using the source side of ‘in-domain’ data from TMs. The adaptation of the spell checker helped us to eliminate most of the false positives flagged by the original unadapted spell checker. The errors flagged by the spell checker were replaced with the highest ranking suggestion from the spell checker. As in Section 3.4, the spelling corrections were applied only to the test sets to ensure a reduction in the number of spelling error-based OOVs.

3.6 Supplementary Data Selection

To take care of the VAL tokens which are valid words but absent in the training data, we explored techniques of mining supplementary data to improve the chances of successfully translating these tokens. We used the following freely available parallel data collections as potential sources of supplementary data:

1. Europarl (Koehn, 2005): Parallel corpus comprising of the proceedings of the European

Parliament.

2. News Commentary Corpus: Released as a part of the WMT 2011 Translation Task.³
3. OpenOffice Corpus: Parallel documentation of the Office package from OpenOffice.org, released as part of the OPUS corpus (Tiedemann, 2009).
4. KDE4 Corpus: A parallel corpus of the KDE4 localization files released as part of OPUS.
5. PHP Corpus: Parallel corpus generated from multilingual PHP manuals also released as part of OPUS.
6. OpenSubtitles2011 Corpus:⁴ A collection of documents released as part of OPUS.
7. EMEA Corpus: A parallel corpus from the European Medical Agency also released as part of OPUS corpus.

To select relevant parallel data, we queried each of the parallel corpora with the VAL OOV words and added sentence pairs containing the OOVs into the existing ‘in-domain’ parallel corpora. During the selection process, the number of parallel sentences selected for any particular OOV item was restricted to a threshold of 500 for En–De and 67 for En–Fr. This was done to limit the size of the selected ‘out-of-domain’ supplementary data such that it did not exceed the size of the TM-based (in-domain) training data. The target sentences of the selected parallel data were added to the language model to ensure language model adaptation. This process allowed us to cover 87.55% and 92.13% of VAL OOVs for En–De and En–Fr language pairs, respectively.

3.7 OOV Tokens Unsuitable for Translation

The last remaining category of OOVs (NTR) represents tokens for which translation was usually unnecessary. Most of these comprised product or service names, names of the forum users or numeric tokens. This class of tokens was not explicitly handled under the assumption that due to their absence from the training data (and hence from the phrase table), they would be preserved during the translation process in the standard SMT setup.

³<http://www.statmt.org/wmt11/translation-task.html>

⁴<http://www.opensubtitles.org/>

4 Experiments and Results

4.1 Pre- and Post-Processing

Prior to training, all the bilingual and monolingual data were subjected to tokenization and lower casing using the standard Moses pre-processing scripts. However, for the regular expression-based normalization, the standard tokenizer is slightly modified to ensure that unique placeholders (Section 3.3) are not tokenized. During the replacement process a mapping is maintained between the unique placeholders, the line number and the actual token replaced. This mapping file is used later in the post-processing step to substitute the actual tokens in the position of the unique placeholders. For target sentences having multiple placeholders of the same type, the corresponding actual tokens are replaced in the order in which they appeared in the source.

4.2 Tools

For all our translation experiments we used OpenMaTrEx (Dandapat et al., 2010), an open source SMT system which wraps the standard log-linear phrase-based SMT system Moses (Koehn et al., 2007). Word alignment was performed with Giza++ (Och and Ney, 2003). The phrase and reordering tables were built on the word alignments using the Moses training script. The feature weights for the log-linear combination of the feature functions were tuned using Minimum Error Rate Training (Och, 2003) on the devset in terms of BLEU (Papineni et al., 2002). We used 5-gram language models in all our experiments created using the IRSTLM (Federico et al., 2008) language modelling toolkit using Modified Kneser-Ney smoothing. Results of translations in every phase of our experiments were evaluated using BLEU and TER (Snover et al., 2006).

For the spell checking task we used a combination of two off-the-shelf spelling correction toolkits. Using the ‘After the Deadline toolkit’ (AtD)⁵ as our primary spell checker, we also used a Java wrapper on Google’s spellchecking API⁶ to supplement the AtD spell checking results. However, the ‘in-domain’ adaptation of the spell checker (Section 3.5) could only be achieved for the AtD spell checker.

⁵<http://open.afterthedeathline.com/>

⁶<http://www.google.com/tbproxy/spell?lang=en&hl=en>

4.3 Experimental Results

Table 3 shows the different BLEU and TER scores for translations subject to each category of normalization and supplementary data selection, along with the percentage of OOV word reduction they result in, for both the test sets under consideration. The last row of the table reports the results for translating only regular expression-based normalized test sets (without the other normalizations) using supplementary training data enhanced models.

The experiments were carried out in five different phases, each focussing on reducing one category of OOV words in the English forum data. For the baseline translation and language models, the TM and forum data was subjected to only basic clean-up such as dropping empty lines and very long sentences (more than 100 tokens). The baseline testsets were then subjected to the following adaptations in a cumulative step-by-step manner:

1. Regex: Regular Expression-based normalization for the reduction of MASK OOVs.
2. Wrd-Split: Heuristic-based tokenization for normalization of FW OOVs.
3. Spell-Chk: Off-the-shelf spell checking based normalization for reducing SPERR.
4. Adapted-Spell-Chk (Ada SpChk): spell checking using domain adapted spell checkers to reduce false positive flags.
5. Sup-data: Supplementary data selection and addition to enrich existing models to reduce VAL OOVs.

The final experimental step (Regex+Sup) did not involve any specific normalization, but was rather performed to investigate the effect of supplementary data selection on regex-based normalized test sets without any other normalizations.

As the results in Table 3 show, regular expression-based normalization results in a 0.55 absolute (2.12% relative) BLEU point improvement in En-De translations and a 0.66 absolute (1.93% relative) BLEU point improvement for En-Fr translations for Test-1. For Test-2, the improvements are 0.31 absolute (1.45% relative) BLEU points and 0.38 absolute (1.26% relative) BLEU points for En-De and En-Fr, respectively. While the Test-1 improvements are statistically significant at $p=0.05$ level using bootstrap resampling (Koehn, 2004), the Test-2 improvements are not statistically significant. The TER scores also

Normaliz- ation	En-De						En-Fr					
	Test-1			Test-2			Test-1			Test-2		
	OOV	BLEU	TER	OOV	BLEU	TER	OOV	BLEU	TER	OOV	BLEU	TER
Baseline	-	25.98	0.6407	-	21.32	0.6361	-	34.14	0.5250	-	30.27	0.5405
Regex	21.33	26.53*	0.6372	9.42	21.63	0.6332	19.43	34.80*	0.5179	9.67	30.65	0.5402
Wrd-Split	3.48	26.59	0.6380	1.54	21.68*	0.6284	3.14	34.89	0.5178	1.50	30.77*	0.5386
Spell-Chk	8.06	26.78	0.6365	37.16	22.50*	0.6279	8.57	35.10	0.5158	36.17	31.60*	0.5303
Ada-SpChk	4.27	26.92	0.6299	11.30	23.17*	0.6174	3.57	35.33	0.5121	11.00	32.28*	0.5128
Sup-data	13.74	27.86*	0.6207	13.53	24.08*	0.5923	17.43	36.04*	0.5024	15.17	33.75*	0.5043
Regex-Sup	13.74	27.45	0.6242	13.53	23.01	0.6191	17.43	35.55	0.5068	15.17	31.96	0.5178

Table 3: Translation Results after normalization and supplementary data selection. The OOV column indicate the percentage of total OOVs reduced in each step. * denote statistically significant improvement over the scores in previous row.

show a decreasing trend which also suggest translation quality improvement. The reason behind this may be attributed to the larger percentage of category-1 tokens in Test-1 compared to Test-2. The number of OOV words is reduced by 135 and 136 on Test-1 and 55 and 58 on Test-2 with respect to different training data sets. The improvements result from the fact that this normalization helps to maintain intra word ordering within MASK tokens and avoid translation of constituent sub-tokens. The first example in Table 4 clearly depicts this particular behaviour for MASK tokens.

Using the fused word splitting technique on the regex-processed testsets, the scores improve only by 0.06 absolute (0.23% relative) BLEU points and 0.09 (0.26% relative) absolute BLEU points on Test-1 over the previous normalization scores, for En-Fr and En-De respectively. For Test-2 the improvements are 0.05 absolute (0.23% relative) BLEU points and 0.12 absolute (0.39%) BLEU points for En-De and En-Fr translations, respectively. Despite the marginal improvement, the improvements for Test-2 were statistically significant at $p=0.05$ level. Improvements in Test-1 were not significant. The reason for the marginal improvement becomes apparent when observing the low percentage of OOV’s (Table 3) reduced by this mechanism. However, the percentage of category-2 tokens in test-2 is nearly double that of Test-1 which may explain the statistical significance of the improvements gained.

As expected, handling the spelling errors using spell checkers had a profound effect on the reduction of OOV words for the high density spelling error testset, Test-2. Using the adapted spell checker on this test set, we achieve an improvement of 1.49 absolute (6.87% relative) BLEU points for En-De and 1.51 absolute (4.9%) BLEU points for En-Fr translations. This corresponds to a total reduction (combining reductions for unadapted and adapted spell checking) of 283 OOVs for both En-De and

En-Fr test sets. The overall improvement when using spell checkers over the previous normalization results were statistically significant at the $p=0.05$ level. However, for Test-1, with spelling error density reflecting that of average forum data, the improvements are much lower. Adapted spell checking results in a total improvement of 0.33 absolute (1.24% relative) BLEU points for En-De and 0.44 absolute (1.26% relative) BLEU points for En-Fr translations. These are not statistically significant and correspond to a reduction of 78 and 85 OOVs for En-De and En-Fr test sets, respectively. The TER scores also reflect the same level of improvements across the two different test sets.

The fourth phase of experiments, where different parallel data sources are mined guided by the list of VAL OOV words, results in further reduction in the OOV rates and improvement in translation scores. The guided selection process improves the scores by 0.94 absolute (3.49% relative) and 0.71 absolute (2.01% relative) BLEU points for En-De and En-Fr translations, respectively on Test-1. For Test-2 the improvement figures are 0.91 absolute (3.93% relative) BLEU points and 1.47 absolute (4.55% relative) BLEU points for En-De and En-Fr translation, respectively, over the previous normalization results. The TER scores also show similar improvements for both language pairs and test sets. All improvements are statistically significant at the $p=0.05$ level. Furthermore, this technique further reduces the number of OOVs by 79 for the En-De test set and 91 counts for the En-Fr on Test-2. The corresponding reductions for Test-1 are 87 and 122 for En-De and En-Fr, respectively.

In summary, using supplementary data selection techniques to complement the normalization resulted in statistically significant overall improvements of 1.88 absolute (7.24% relative) and 1.9 absolute (5.57% relative) BLEU points over the baseline scores on Test-1. On Test-2, the im-

provements were 2.76 absolute (12.95% relative) and 3.48 absolute (11.49% relative) BLEU points for En–De and En–Fr translations, respectively. Translating the regex-normalized test sets (without word splitting and spell checking) with the supplementary data-enhanced models, we aimed to assess the impact of supplementary data selection technique in contrast to that of the normalization methods. For Test-1, the results show that this process results in scores slightly better (0.53 absolute BLEU on En–De and 0.22 absolute BLEU for En–Fr) than those achieved by complete normalization (adapted spell checking scores, row 5 in Table 4.3). For Test-2 however, the scores are lower than the adapted spell checking scores by 0.16 and 0.32 absolute BLEU points for En–De and En–Fr, respectively. Overall results clearly show that for general forum data (with average spelling error density), fully automatic supplementary training data acquisition can perform as well and sometimes better than semi-automatic normalization although they target different types of OOVs. Finally for very noisy data, normalization complemented with supplementary data selection really pays off.

Type	Sentence
Src	5. click on the folder button and navigate to c : \documents and settings \all users \application data \and select the carbonite folder
Ref	5. klicken sie auf die ordnerschaltfläche und öffnen sie den ordner " c : \documents and settings \all users \application data \carbonite "
Baseline	5. klicken sie auf den ordner " und navigieren sie zu c : \dokumente und einstellungen \alle benutzte \anwendungsdaten \ und wählen sie die carbonite ordner
Regex	5. klicken sie auf die schaltfläche " und wechseln sie zum ordner c : \documents and settings \all users \application data \carbonite und wählen sie die carbonite ordners
Src	re : nis09 did not detect 8 threats & 23 infected objects.and 16 suspicious objects ?
Ref	re : nis09 n' a pas détecté 8 menaces , 23 objets infectés et 16 objets suspects ?
Baseline	re : nis09 n' a pas détecté 8 menaces et 23 infecté objects.and 16 les objets ?
Wrd-Split	re : nis09 n' a pas détecté 8 menaces et 23 infecté objets . et 16 les objets ?
Src	and no for something completely different .
Ref	und nun zu etwas völlig anderem .
Baseline	und keine für something completely anders .
Spck	und nicht für etwas völlig anders .
Src	pretty disappointed with nis parental control not blocking websites on blocked list as well as through their category of websites to block .
Ref	je suis assez déçu que le contrôle parental de nis ne bloque pas les sites web figurant dans la liste bloqués aussi bien que ceux de la catégorie des sites web à bloquer .
Baseline	assez disappointed avec contrôle parental de nis pas le blocage de sites web sur liste bloqués ainsi que par l' intermédiaire de leur catégorie de sites web à bloquer .
Sup	assez déçu de contrôle parental de nis pas le blocage de sites web sur liste bloqués ainsi que dans leur catégorie de sites web à bloquer .

Table 4: Translation examples for each normalization and supplementary data selection Technique

In order to substantiate the improvements observed on the automatic evaluation scores, we present some examples from our test sets (both Test-1 and 2), to depict how the normalization or data selection methods actually affect the translations. Table 4 presents 4 different examples of translations each highlighting the effect of a single normalization or data selection technique. The first example clearly shows how regular expression-based masking allows internal parts of the path

structure to be left untranslated, unlike in the baseline set-up. The second sentence (row 5) is an example of the fused word splitting technique enabling better translation of the token ‘objects.and’ which had been treated as an OOV in the baseline. The third example (rows 9-12) highlights the effect of spell checking on the translation quality of the source sentence. Automatic spell checking changes the tokens ‘something completely’ into ‘something completely’ thereby allowing them to be translated. The final set of sentences is an example of how supplementary data selection allows the translation of the valid yet OOV word ‘disappointed’ appearing in the source sentence. As is evident from the examples, the normalization techniques discussed in the paper do work towards better translations for sentences with specific OOV types. However, the relative densities of each type leads to varied improvements in scores reported in Table 4.3.

5 Conclusion and Future Work

In this paper we have explored a set of normalization techniques to achieve better translation quality for user-generated forum content. We have shown that supplementary data selection techniques positively complement normalization in terms of translation quality. For test data with spelling error density representative of the overall forum data (Test-1), supplementary data selection on its own can produce improvements similar to those achieved through normalisation (targeting different OOVs). While data normalization carried out at the level reported in this paper (with different OOV categories and different normalisation approaches for each) is a semi-automatic process which requires some manual analysis, supplementary data selection is fully automatic and involves much less overall effort. Thus, for moderately noisy datasets (such as Test-1), normalization may not always be worth the effort. For more noisy datasets (e.g. Test-2) however, normalization does improve translation quality more effectively than data supplementation.

In this research, the classification of OOV words was done in a semi-automatic fashion. Using automatic classification techniques to identify the different categories in OOV words would be one of the prime future directions here. Furthermore, a detailed investigation of the individual contributions of multiple resources used for supplementary

data selection is required to better understand the cause of the improvements in scores. Finally we would also like to work towards developing automatic threshold detection techniques for optimal supplementary data selection.

Acknowledgments

This work is supported by Science Foundation Ireland (Grant No. 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at Dublin City University. We thank the reviewers for their insightful comments.

References

- Banerjee, Pratyush, Sudip Kumar Naskar, Johann Roturier, Andy Way, and Josef van Genabith. 2011. Domain Adaptation in Statistical Machine Translation of User-Forum Data using Component Level Mixture Modelling. In *Proceedings of the Thirteenth Machine Translation Summit*, pages 285–292, Xiamen, China.
- Dandapat, S., M. L. Forcada, D. Groves, S. Penkale, J. Tinsley, and A. Way. 2010. OpenMaTrEx: A Free/Open-Source Marker-Driven Example-Based Machine Translation System. In *Proceedings of the 7th International Conference on Natural Language Processing (IceTAL 2010)*, page 121–126, Reykjavík, Iceland.
- Daume III, Hal and Jagadeesh Jagarlamudi. 2011. Domain adaptation for machine translation by mining unseen words. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 407–412, Portland, Oregon, USA.
- Eck, Matthias, Stephan Vogel, and Alex Waibel. 2004. Language model adaptation for statistical machine translation based on information retrieval. In *Proceedings of 4th International Conference on Language Resources and Evaluation, (LREC 2004)*, pages 327–330, Lisbon, Portugal.
- Federico, Marcello, Nicola Bertoldi, and Mauro Cettolo. 2008. IRSTLM: an open source toolkit for handling large scale language models. In *InterSpeech 2008: 9th Annual Conference of the International Speech Communication Association*, pages 1618–1621, Brisbane, Australia.
- Flournoy, Raymond and Jeff Rueppel. 2010. One Technology : Many Solutions. In *AMTA 2010: Proceedings of the Ninth Conference of the Association for Machine Translation in the Americas*, pages 6–12, Denver, Colorado, USA.
- Habash, Nizar. 2008. Four techniques for online handling of out-of-vocabulary words in arabic-english statistical machine translation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 57–60, Columbus, Ohio.
- Hildebrand, Almut Silja, Matthias Eck, Stephan Vogel, and Alex Waibel. 2005. Adaptation of the Translation Model for Statistical Machine Translation based on Information Retrieval. In *10th EAMT Conference: Practical Applications of Machine Translation, Conference Proceedings*, pages 119–125, Budapest, Hungary.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *ACL 2007, Proceedings of the Interactive Poster and Demonstration Sessions*, pages 177–180, Prague, Czech Republic.
- Koehn, Phillippe. 2004. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, (EMNLP 2004)*, pages 388–395, Barcelona, Spain.
- Koehn, P. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *MT Summit X: The 10th Machine Translation Summit*, pages 79–86, Phuket, Thailand.
- Och, Franz Josef and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29:19–51.
- Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, pages 160–167, Sapporo, Japan.
- Papineni, Kishore, Salim Roukos, Todd Ward, and Weijng Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *40th Annual Meeting of the Association for Computational Linguistics, (ACL 2002)*, pages 311–318, Philadelphia, Pennsylvania.
- Roturier, Johann and Anthony Bensadoun. 2011. Evaluation of MT Systems to Translate User Generated Content. In *Proceedings of the Thirteenth Machine Translation Summit*, pages 244–251, Xiamen, China.
- Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas*, pages 223–231, Cambridge, MA.
- Tiedemann, Jörg. 2009. News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. pages 237–248.

Long-distance reordering during search for hierarchical phrase-based SMT

Fabienne Braune

Anita Gojun

Alexander Fraser

Institute for NLP
Universität Stuttgart
Pfaffenwaldring 5b

D-70569 Stuttgart, Germany

braunefe, gojunaa, fraser@ims.uni-stuttgart.de

Abstract

Long-distance reordering of syntactically divergent language pairs is a critical problem. SMT has had limited success in handling these reorderings during inference, and thus deterministic preprocessing based on reordering parse trees is used. We consider German-to-English translation using Hiero. We show how to effectively model long-distance reorderings during search. Our work is novel in that we look at reordering distances of up to 50 words, and conduct a detailed manual analysis based on a new gold standard.

1 Introduction

Word reordering is a well-known issue in SMT. One successful approach has been to use rule-based preprocessing to reorder parse trees. We would like to perform reordering during inference. Phrase-based hierarchical models (Chiang, 2007) have helped, but reordering over long distances is still a difficult open problem. Consider the following German sentence and English output taken from the hierarchical component of the Moses toolkit (Hoang et al., 2009). These sentences illustrate the successful reordering of the participle *geeinigt* (*agreed*) from the end of the German clause, to be next to the English auxiliary *have*.

- (1) deutschland (germany) , frankreich (france) , israel (israel) und (and) die (the) usa- (us) *haben* (*have*) sich (themselves) im (in) mai (may) 2006 darauf (on) *geeinigt* (*agreed*) , es (it) zu (to) tun (do).
- (2) germany , france , israel and the us *have agreed* in may 2006 , to do it .

This reordering involves a word movement over 5 tokens and is therefore not a long-distance reorder-

ing. However, there can be many more words between the German auxiliary and participle, so the movement required can become arbitrarily large. Restrictions on reordering distance are typically used with hierarchical systems like Hiero because previous experiments have shown some evidence that long-distance reorderings are not effective (Chiang, 2007). We are not aware of careful explorations focusing exclusively on long-distance reorderings in search, prior to our work.

We present the first step towards solving the problem of long-distance reorderings during search. We first analyze the rule geometry required for long-distance German-to-English movement and modify extraction of Hiero's SCFG rules to focus on these rules. We then introduce a new idea, span-width-specific rules in the grammar. By *span*, we denote the number of tokens that are allowed to be covered by a non-terminal symbol (usually "X") in the source language side of an SCFG rule. We define long-distance reordering as occurring over spans containing 11 to 50 source words, and define a new set of rules which apply over spans of 11 to 50 words, which we call *long spans*. We combine these rules (applied on 11 to 50 word spans) with the standard Hiero X rules (applied on 1 to 10 word spans). We further restrict our rules by applying a basic POS-based filtering so that long-span rules contain verbs. Finally, we introduce another innovation to Hiero, which is to block our long-span rules from crossing clause boundaries. We release the source code changes to Hierarchical Moses and our annotated test set for further study by other research groups.

2 Previous Work

The long-distance reordering issue has been considered in phrase-based SMT as well as in syntax-

based SMT. The basic phrase-based model is able to handle word movement up to six tokens but a decrease of performance is observed at higher distortion limits (Koehn et al., 2007). Many reordering methods use a distortion limit between 6 and 9 words (e.g., (Tillmann and Xia, 2003; Koehn et al., 2007; Galley and Manning, 2008)). Green et al. (2010) implement a future cost function and a distortion model that outperform a standard phrase-based system using a distortion limit of 15. We work with longer distances.

Collins et al. (2005) discuss an approach combining rule-based transformations with (phrase-based) SMT. In a preprocessing step, the source language is reordered using parse trees. The restructured output is then provided to a phrase-based MT system. Deterministic preprocessing has several drawbacks such as high sensitivity to parsing errors or the propagation of wrong phrase correspondences (created by incorrect reordering of the training data) into the learned translation probabilities. Preprocessing also does not allow the interaction of long-distance reordering decisions with nearby translation decisions via the language model.

In syntax-based SMT, the size of reordering is given by the span of the grammar rules. In approaches which do not use linguistic syntactic labels (such as ITG (Wu, 1997) or Hiero, where only the start symbol S and the non-terminal X are used), the maximal span size allowed in implementations is often between 10 and 15 tokens, because using wider spans has (in experiments done in the past) resulted in decreased translation quality (e.g., (Chiang, 2007)). Zollmann et al. (2008) expand the span size to 15 only for the translation of short sentences. We present work within the hierarchical phrase-based MT framework that considers rules allowed to span up to 50 words.

Approaches using linguistic syntactic labels (obtained from a source language or target language parser, or both) sometimes also use such span restrictions. However, systems which use source-side-syntactic parses of the test set sometimes do not use such a restriction because they force a match with a syntactic constituent (in the source language parse). There have been many approaches looking at backing off from hard source-side constraints on syntactic labels to Hiero-style X rules (e.g., (Venugopal et al., 2007; Hoang and Koehn, 2010; Mylonakis and Sima'an, 2011)).

Due to the diversity of possible structures for German clauses and to poor parse accuracy on long sentences we restrict our study to Hiero, with a view towards integrating soft syntactic constraints (Marton and Resnik, 2008; Chiang, 2010) in the future. Hard syntactic constraints would suffer from too many errors (and too much sparsity) to improve performance in our approach. Our study looks at the specific phenomenon of long-distance reordering in a hierarchical-phrase based framework, by modifying Hiero to support span-width specific rules. We consider exactly the reorderings required for the German-to-English clause reordering problem and focus particular attention on ensuring that the correct reorderings can be considered during search. We employ simple low-knowledge techniques to improve the chances that the correct translation is not only considered but also chosen, but we expect that implementing soft syntactic constraints will improve this further.

The question of handling long-distance movements in hierarchical MT has also been addressed by Sudoh et al. (2010) who present a method that deals with reordering involving connecting together several embedded clauses. Our work differs from (Sudoh et al., 2010) because we handle long-distance reordering inside of a single clause. Moreover, the method by (Sudoh et al., 2010) divides the source language into clauses in a preprocessing step and re-unifies the obtained translations in a post-processing step. In our approach, reordering is performed during inference.

3 Long-Distance Reorderings

In this section, we discuss the type of reordering Hiero is not able to handle, given the constraints used by Chiang (2007). Then we present an analysis of the frequency of such reorderings in a commonly used test set for German-to-English translation. We break this down by the pattern of non-terminals and terminals that will be needed to carry these reorderings out.

Problems with the Hiero Constraints. We first show why hierarchical Moses with standard settings is not able to perform long-distance reordering. To keep the presentation simple, we present example reorderings over distances between 10 and 20 words but our approach handles word movement over 50 words. Consider a German input and its reference translation:¹

¹This example is from the WMT 2009 test set, see section 5.

- (3) der (the) preis (price) der (of) täglichen (day-to-day) verbrauchsartikel (consumer goods) in den (the) hypermärkten (giant supermarkets) *ist (is)* in weniger (less) als (than) 20 monaten (months) um (by) mehr als (over) 30 prozent (percent) *gestiegen (increased)*.
- (4) in the giant supermarkets, the price of day-to-day consumer goods *soared* by over 30 percent in less than 20 months.

In order to obtain the reference English translation, the German verbal complex *ist ... gestiegen* has to be translated as a unit into *soared*. The only way to perform this movement using Hiero consists in producing a derivation including a rule of the form:

- (5) $X \rightarrow < \text{ist (is)} X_{10}^{14} X_{15}^{19} \text{ gestiegen (increased)} ; \text{soared } X_2 X_1 >$

where the indices on the source non-terminals denote the positions of the source sentence tokens which are covered by X_i^j when the rule is applied, e.g., X_{10}^{14} covers the source language segment *in weniger als 20 monaten*, and the target non-terminals are annotated because they have been swapped (we only annotate the target language side if there is a reordering). The span-width of rule (5) is 12, which corresponds to the sum of the span-widths of the non-terminals and the number of terminals in the rule. In Hiero such a rule cannot be picked during decoding, because only rules with a maximal span of 10 words are allowed. Therefore the translation of the verbal complex *ist ... gestiegen* has to be performed in two steps. Possible rules are:²

- (6) $X \rightarrow < \text{ist (is)} X_{10}^{14} ; \text{is } X >$
- (7) $X \rightarrow < \text{um mehr (over)} X_{17}^{19} \text{ gestiegen (increased)} ; \text{have increased by more } X >$

where X_{17}^{19} covers *als 30 prozent*. The complete decoding process yields the malformed English sentence:

- (8) the price of daily used in the hypermärkten **is** in less than 20 months **have increased** by more than 30 % .

Besides the movement of the German participle from the end of a German clause to be next to the English auxiliary, other problematic phenomena include the movement of German clause-final particles to be next to the English verb or the reordering of subordinate clauses.

Long-Distance Rule Patterns. We present an analysis of the frequency and shape of sentence pairs in which a correct reordering requires movement over more than 10 tokens. Within the 450 first sentences in the test set of the ACL WMT

²For such a translation hierarchical Moses can produce a derivation containing more than two rules. To keep the presentation simple, we combine these rules into the two presented.

Segmentation	Pattern	nb. sent	ratio
One non-term	$t^+ X t^+$	40	0.42
Two non-terms	$t^+ X X t^+$	23	0.24
More non-terms	$t^+ X X^+ t^+$	17	0.18
Inversions	$X^+ t^+ , t^+ X^+$	8	0.08
Others	<i>No pattern</i>	8	0.08
Total found sentences		96	1

Figure 1: Patterns of long-distance reordering rules

2009 German-to-English shared task, we have selected sentence pairs in which the minimal sequence of German tokens on which a hierarchical rule has to be applied to obtain the reference is greater than 10. For instance, in sentence (3), the segment beginning at *ist* and ending at *gestiegen* is the minimal segment in which a reordering has to take place in order to obtain the reference translation (4). The rule for this has to be anchored at the beginning and end of this segment. In other words, its source language side must have the general shape "ist X gestiegen". In the remainder of this paper, we call terminal symbols around a gap *anchor points*. We found 96 sentence pairs in which long-distance reordering is required, which is just over 21% of the sentences we considered. We classify the shapes required into patterns which represent the anchor points as well as the necessary segmentation of the material between those points. Consider the following German sentence and English reference.

- (9) der (the) ezb (ecb) zufolge (according to) *wird (will)* die (the) inflation (inflation) im (in the) jahr (year) 2008 von (from) 2,1 auf (to) 2,5 prozent (percent) *steigen (rise)*.
- (10) according to the ecb , inflation *will rise* from 2.1 to 2.5 in the year 2008

A correct reordering of sentence (9) into (10) requires the translation of the segment *die inflation* to move towards the (English) verbal complex while the segment *im jahr 2008 von 2,1 auf 2,5 prozent* has to move behind the complex. Consequently, the source side of a hierarchical rule has to segment these units for reordering them. The pattern of such a rule is the first anchor point (*wird*), two non-terminals covering each reordered segment, followed by the second anchor point (*steigen*). This minimal German pattern can be represented by $tXXt$. We capture rules that involve several terminals around non-terminals by generalizing our patterns (e.g., $t^+ X X t^+$). The patterns for long-distance reorderings in the 450 sentence set are shown in Figure 1.

4 Decoding with Large-Span Rules

We have shown that hierarchical rules for long-distance reordering have a particular shape on source language side. Basing on this observation, we modify the grammar and decoding procedure of hierarchical Moses to build a system which can capture the specificity of such reorderings.

Creating special rules for long-distance reordering. In a first step, we extract rules designed for long-distance reordering, that is rules that have a more specific geometry than standard hierarchical rules. By "specific geometry", we denote rules that match the patterns presented in section 3. We want these rules only to be considered when long-distance reordering is required. In order to achieve this, we define different spans on which our rules are allowed to be used during decoding. In other words, we build a Hiero grammar consisting of two subsets which apply on different spans during decoding. The first set contains all Hiero rules extracted using the standard procedure. Rules belonging to this set apply to spans having size from 1 to 10. The second set contains rules with the following properties:

- (i) Instead of having one aligned terminal on each side of a rule, we require each source side non-terminal except the first to have at least one aligned terminal on its left and one on its right.
- (ii) In each rule extracted following constraint (i) we allow non-terminal symbols to be further split into adjacent non-terminals.

Rules extracted following constraints (i) and (ii) build an SCFG grammar with rules having the same shape, on the source language side, as the patterns presented in section 3. Note that because we allow the first non-terminal of each rule to have no terminal on its left, we also capture patterns of the form X^+t^+ but not t^+X^+ . Because these rules are specifically designed for long-distance reordering they are only used on spans having size between 11 and 50 in decoding.

The creation of a specific SCFG grammar for large spans allows the handling of long-distance reordering while keeping the set of hierarchical rules acceptably small. Our set of rules for long-distance reordering are extracted on spans from 1 to 10. The decision to extract on small spans is based on the observation that most rules needed for the long-distance reorderings required to reorder

German clauses can be found in short span examples. We found that rules extracted from longer spans were noisy and rarely correct and that the rules for many examples of long-distance reorderings which are present in the training data can not be extracted because noisy alignments incorrectly block extraction. Rules are scored by computing maximum likelihood estimation using phrase counts as described in (Chiang, 2007).

Let us illustrate the functioning of large-span rules. Consider again the sentence pair presented in section 3. In a system containing large-span rules, the rule $\rightarrow \langle \text{ist (is) } X \ X \ \text{gestiegen (increased) ; soared } X_2 \ X_1 \rangle$ is extracted in training and applied on spans between 11 and 50 at decoding. Hence, the rule necessary for a correct translation of our example sentence is available in our extended system.

Decoding with long-distance rules. The hierarchical Moses decoder allows the user to work with multiple sets of hierarchical rules having different maximal span sizes. However, the possibility to decode using rules with span greater than a minimal threshold was not implemented in hierarchical Moses.³ In order to overcome this problem, we have defined a new type of grammar for which the lookup procedure only selects rules greater than a given minimal span.

Making long-distance rules reachable. Creating a set of rules applying on spans 11 to 50 during decoding is not sufficient to allow our modified system to effectively use large-span rules. In order to be applied, a hierarchical rule must be reachable, meaning that there must be a valid derivation for the subtree covered by the non-terminal X_i^j in the source language side of the considered rule. Because hierarchical rules can only apply on spans up to 10, those rules can only cover sub-spans of X_i^j when $j - i$ is smaller than 11. Even when allowing adjacent non-terminals, this size is likely to be greater than 10. If no other grammar is accessible to the decoder, these partial translations cannot be combined sequentially and for spans greater than 10, large-span rules have to be applied recursively. This massively restricts the applicability of long-distance rules. It is important to note here that in hierarchical Moses glue rules can only be applied

³This is due to the fact that rule-lookup is done in an incremental fashion. For each type of grammar provided to the decoder, the lookup procedure only selects rules for which all subspans have already been explored.

on partial translations of an entire sentence.⁴ In other words, a glue rule has the form $S \rightarrow \langle S X ; S X \rangle$, where S corresponds to the beginning of a sentence whereas a rule for sequentially combining segments under a considered span should have the shape $X \rightarrow \langle XX; XX \rangle$. To make our large-span rules reachable, we augment our system with this rule. In summary, our decoder has access to four different grammar rule tables: (i) the two standard “S” rules (ii) the full set of Hiero rules on spans smaller than 10 (iii) rules with specific geometry on spans of size 11 to 50 (iv) an $X \rightarrow \langle XX; XX \rangle$ rule on spans of size 1 to 50.

Filtering out poorly informative rules. Even when performing a rule extraction procedure with a constraint on non-terminals the following rules are part of the extracted grammar.

(11) $X \rightarrow \langle \text{der (the) } X , ; \text{ the } X , \rangle$

(12) $X \rightarrow \langle , X . ; , X . \rangle$

Such rules are not useful for achieving long-distance reordering. Moreover, they tend to get high translation scores and are likely to be chosen often during decoding. This factor contributes to the fact that after tuning with MERT, the weight assigned to the count feature of large-span rules is too low to allow the required reordering to take place. We address this problem by using a very simple filter on the grammar operating on large spans. We only keep rules that contain at least one verb on source and target language side.⁵

Clausal boundary restriction. The hierarchical patterns for long-distance reordering rules identified in section 3 are intra-clausal patterns. This means that they only apply inside of a single clause, which can, however, contain embedded clauses. In other words, when a pattern of the form $t^+ X^+ t^+$ in the source language side of the rule is matched to a segment which begins in one clause and ends in another clause, then the rule is likely to be wrongly anchored. As an illustration, consider source language sentence (13) and rule (14) where the non-terminal X covers token 3 to 12.

(13) er (he) **ist (is)** als (as) solist (solist) unterwegs (travelling) und (and) **hat (has)** seine (his) karriere (career) eher (rather) im westen (in the west) **aufgebaut (built)** .

(14) $X \rightarrow \langle \text{ist (is) } X_3^{12} \text{ aufgebaut (built) ; is built } X \rangle$

⁴This corresponds to Chiang’s definition of glue-rules in (Chiang, 2007).

⁵We tag the German and English parallel training corpus with TreeTagger, and discard extracted rule tokens which do not contain a verb on both sides; we then delete the POS tags.

The anchor points *ist* and *aufgebaut* match two verbs that do not belong to the same complex. This erroneously reorders the participle *built* next to the verb *is* (instead of *has*). Consequently, a malformed sentence like (15) is generated.

(15) he **is built** as solist traveling and **has** his career more in the west

This can be avoided by forcing rules to be applied inside of a single clause. This is achieved by extracting clause boundaries from the parse tree of each source language sentence.⁶ Clauses are then represented as intervals delimited by the identified boundaries. The constraint we enforce regarding clause boundaries works as follows: if the first terminal of a rule is inside of a clause, then the last terminal of the same rule has to be inside of the same clause. In our example, if the starting point of a rule is at a position between 0 and 5, then its end position has to be smaller or equal to 6. This restriction allows the avoidance of all wrong anchoring related to the crossing of clause boundaries. For instance, in sentence (13) above, rule (14) would not be allowed to apply because its first terminal is at position 1 in the source sentence while its second terminal is at position 12. Note that this also handles embedded clauses correctly.

5 Experimental Setup

The baseline system for our experiments is hierarchical Moses with a span size up to 50 tokens instead of 10 in the standard settings. Enabling hierarchical Moses to reorder over long distances involves two main modifications. First, hierarchical rules have to be extracted for spans having a maximal size of 50 tokens instead of 10. Second, the decoder has to be allowed to pick rules with span size 50. Extraction of hierarchical rules on spans containing up to 50 tokens is intractable in terms of cpu time and disk space. In order to nevertheless work with such a system we adopt the same strategy as described in section 4: we extract rules on spans up to 10 and allow the obtained grammar to apply to spans up to 50 words during decoding. The modified system presented in section 4 will be evaluated against this baseline. Note that choosing a baseline with extended span size allows us to evaluate our approach against a system enabled to perform long-distance reordering. The results obtained by hierarchical Moses with standard settings

⁶We use BitPar (Schmid, 2004) to extract clause boundaries. Boundaries correspond to the position of the token labeled by the opening and closing S-Nodes in the parse tree.

on all test sets is also provided, but since it can not perform long-distance reorderings we provide no further analysis.

The translation model has been trained using 1,502,301 bilingual sentences after length ratio filtering. GIZA++ (Och and Ney, 2003) has been used for generating the word alignments, combined with the grow-diag-final-and heuristic (Koehn et al., 2007). We trained our monolingual 5-gram language model using the English side of the training data. Feature weights are tuned using Pairwise-Ranked optimization (Hopkins and May, 2011) followed by standard MERT line search (for fine tuning of the length penalty). We evaluate two tasks. For the ACL WMT 2009 German-to-English shared task, we use news-dev2009a as our dev set, and news-dev2009b as our test set. To reduce the effect of data sparsity for the difficult task of long-distance reordering, we also consider a Europarl translation task, using the same system (with the same training data), but using Europarl test2007 as our dev set, and Europarl dev2006 as our test set.

6 Evaluation

We perform a two step evaluation procedure. First the compared systems are evaluated using automatic metrics. In a second step we compare the systems using a manually annotated test set.

Automatic Evaluation. As a first automatic evaluation metric, we use 4-gram BLEU (Papineni et al., 2002). Because BLEU does not consider the positions of matched n-grams and does not capture the distance of erroneous reorderings, we use LRscore (Birch and Osborne, 2011) as a second metric to evaluate reordering quality. This method compares the alignments between input and reference with the alignments between input and system output (Kendall’s Tau over permutations is used as the distance metric). We provide two measures (i) LRscore as proposed in (Birch and Osborne, 2011) where the interpolation parameter⁷ α is set to 0.2623 (ii) reordering performance only, i.e., $\alpha = 1$.

Figure 2 shows the results for all systems on the Europarl and ACL WMT 2009 tasks. Our improved hierarchical system is denoted by Improved-50. Hierarchical Moses with span sizes up to 50 tokens is Std-50. Hierarchical Moses with standard settings is denoted by Std. On the

⁷This parameter controls the trade-off with BLEU.

Europarl translation task, Std-50 and Improved-50 achieve a similar performance in terms of BLEU while Improved-50 obtains a 0.31 better LRscore when considering the reordering distance only ($\alpha = 1$). When using interpolated LRscore, Improved-50 is 0.26 better than Std-50. On a test set belonging to the same genre as the training set, improved-50 provides better reordering quality. On ACL WMT 2009, Improved-50 obtains a 0.4 worse BLEU score than Std-50 together with a 0.4 improvement in LRscore when considering the reordering distance only. The interpolated LRscore metric shows a 0.2 improvement of Improved-50 over Std-50. On a test set belonging to a genre different than the training set, Improved-50 causes a small decrease in BLEU together with somewhat better reordering. The decrease in BLEU observed is mainly bad lexical choice caused by using rules on a different domain.

Manual Evaluation. In a second step, we report the amount of correct and incorrect long-distance reordering performed by the evaluated systems on a manually annotated test set. Our test set consists of the 450 sentences presented in section 3. For counting correct reordering, we consider each sentence in our set and evaluate the translation of its source language pattern. We look at the anchor points t as well as the segments represented by X . We provide two types of counts (i) **reference matches** and (ii) **human matches**. A **reference match** requires the translation of the anchor points t to be in the same order and have the same surface form as in the reference translation. We also require the segments covered by X to be in the same order as in the reference translation. A **human match** includes translations where the reordering of the anchor points t is the same as in the reference, but we don’t require the translation of t to have the same surface form as in the reference. We also allow the ordering of the segments covered by X to be different than in the reference as long as it is considered as correct by our human annotator. As an illustration for the difference between reference and human matches, consider again source sentence (3) and reference (4). Also consider the following possible translation of (3):

- (16) the price of day-to-day consumer goods in supermarkets increased in less than 20 months by over 30 percent.

Sentence (16) cannot be considered as a **reference match** because it translates *ist ... gestiegen* into *increased* instead of *soared* and because the seg-

System	BLEU (dev)	BLEU (test)	LRscore ($\alpha = 1$)	LRscore ($\alpha = 0.2326$)
Improved-50 (Europarl)	29.24	28.32	70.38	60.60
Std-50 (Europarl)	29.49	28.27	70.07	60.34
Std (Europarl)	29.13	28.00	70.82	60.68
Improved-50 (ACL WMT 2009)	18.86	18.91	67.92	56.52
Std-50 (ACL WMT 2009)	18.77	19.30	67.52	56.33
Std (ACL WMT 2009)	18.54	19.30	67.54	56.32

Figure 2: Europarl and ACL WMT 2009 German-to-English shared tasks

ments *in less than 20 months* and *by over 30 percent* are reversed. This sentence is, however, a **human match**. Each reference match is also a human match and all human matches are counted as **correct**. We make the simplifying assumption that each reordering involving a large-span rule on a sentence which is not in our set is **wrong**. We provide a further count denoted by **pattern match** which includes all cases where the source side pattern has been matched using a large-span rule but where the system nevertheless yielded an incorrect translation. As will be shown below this measure allows us to evaluate the potential of a grammar to apply long-distance reordering rules even when the translation is wrong. We also report cases where a system is able to reorder over distances greater than 10 words by gluing together rules that translate the edges of the reordering. A correct translation of sentence 19 can be obtained, for instance, by using rules 17 and 18:

- (17) $X \rightarrow < \text{wird (will)} X_5^6 ; X \text{ will} >$
(18) $X \rightarrow < X_7^8 \text{ 2008 } X_{10}^{14} \text{ steigen (increase) ; increase } X X \text{ 2008} >$
(19) according to the ecb , inflation *will rise* from 2.1 to 2.5 in the year 2008

Note that this strategy only allows the performance of a restricted amount of long-distance reorderings: sentences similar to 3 cannot be reordered in this way, and word movements cannot be over a distance of more than 22 words.

Figure 3 shows the amount of correct and incorrect long-distance reordering performed by Std-50 and Improved-50 on our manually annotated test set.⁸ For Std-50 we observed 14 cases where long-distance reordering is performed where not required (on sentences outside of our selected sentences). Std-50 correctly reorders 9 sentences with the gluing strategy described above. Std-50 is able to correctly match a source side pattern in only 17 cases. When a pattern has been matched, the system is generally able to correctly translate it. The

⁸Because Std cannot perform any long-distance reordering (because of its span restriction), it has no matches.

17 pattern matches of Std-50 yield 13 correct translations. Reference matches are very rare. This is mainly due to the fact that the translation of the anchor points t in the source side of a rule have a different surface form than in the reference. The accuracy of Std-50 in applying large-span rules on sentences where long-distance reordering has to be performed is poor: the amount (14) of reorderings performed on wrong sentences is approximately the same as the amount (17) of German pattern matches. This last observation also explains the poor reordering quality observed on Europarl. Improved-50 matches twice as many source language patterns as Std-50 while performing half as many reorderings on wrong sentences (Figure 3). Improved-50 does a better job in identifying the correct context for application for large-span rules. This system also performs correct long-distance reordering in 24 cases compared to only 14 for Std-50. Again, this represents an improvement over Std-50, but the amount of pattern matches still represents only 35.4% of our test set. Further study is required to determine if this is primarily due to having no rules that could match, or instead because monotonic derivations have a better score. Finally, out of 34 correct pattern matches, 24 yield a correct translation, so the translation is correct in 70% of the matches. We plan to improve the ability of our system to provide a correct translation when a correct source language pattern match is made. We observed 7 cases where long-distance reordering is erroneously performed on sentences outside of our annotated set. The system correctly reorders 8 sentences with the gluing strategy described above. Overall, Improved-50 outperformed Std-50, indicating we have made progress on the difficult problem of long-distance reordering, but there is more work to be done.

7 Conclusion

Long-distance reorderings are required in about 21% of the German sentences in news-test2009b. Simply dropping the span restriction of hierarchi-

Pattern	Std-50			Improved-50		
	Reference	Human	Pattern Match	Reference	Human	Pattern Match
$t^+ X t^+$	2	9	10	2	11	19
$t^+ X X t^+$	0	1	2	0	2	5
$t^+ X X^+ t^+$	0	1	1	0	6	6
$X^+ t^+$ or $t^+ X^+$	1	1	2	1	1	1
<i>No general pattern</i>	1	2	1	1	2	1
Total	4	14	17	4	23	36

Figure 3: Evaluation of the reorderings in our 450 sentence set, broken down by pattern type. Std-50 performs 14 reorderings on sentences where no reordering is necessary; Improved-50 performs 7.

cal Moses results in poor long-distance reordering. We presented an improved version of hierarchical Moses including (i) a specific set of rules for long-distance reordering made reachable and adequately filtered (ii) a decoding procedure using different span-widths (iii) clausal boundary restrictions. Our improved system performs more long-distance reorderings, accurately selects the context of application of large-span rules, and also correctly translates in many cases.

Acknowledgments

This work was funded by Deutsche Forschungsgemeinschaft grant Models of Morphosyntax for Statistical Machine Translation. This work was supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886.

References

Birch, Alexandra and Miles Osborne. 2011. Reordering metrics for mt. In *ACL*.

Chiang, David. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2).

Chiang, David. 2010. Learning to translate with source and target syntax. In *ACL*.

Collins, Michael, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *ACL*.

Galley, Michel and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *EMNLP*.

Green, Spence, Michel Galley, and Christopher D. Manning. 2010. Improved models of distortion cost for statistical machine translation. In *NAACL-HLT*.

Hoang, Hieu and Philipp Koehn. 2010. Improved translation with source syntax labels. In *ACL WMT*.

Hoang, Hieu, Philipp Koehn, and Adam Lopez. 2009. A unified framework for phrase-based, hierarchical, and syntax-based SMT. In *IWSLT*.

Hopkins, Mark and Jonathan May. 2011. Tuning as ranking. In *EMNLP*.

Koehn, Philipp, H Hoang, A Birch, C Callison-Burch, M Federico, N Bertoldi, B Cowan, W Shen, C Moran, R Zens, C Dyer, O Bojar, A Constantin, and E Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL*.

Marton, Yuval and Philip Resnik. 2008. Soft syntactic constraints for hierarchical phrased-based translation. In *ACL-HLT*.

Mylonakis, Markos and Khalil Sima'an. 2011. Learning hierarchical translation structure with linguistic annotations. In *ACL-HLT*.

Och, Franz Josef and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1).

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Schmid, Helmut. 2004. Efficient parsing of highly ambiguous context-free grammars with bit vectors. In *COLING*.

Sudoh, Katsuhito, Kevin Duh, Hajime Tsukada, Tsutomu Hirao, and Masaaki Nagata. 2010. Divide and translate: Improving long distance reordering in statistical machine translation. In *ACL WMT*.

Tillmann, Christoph and Fei Xia. 2003. A phrase-based unigram model for statistical machine translation. In *NAACL-HLT*.

Venugopal, Ashish, Andreas Zollmann, and Stephan Vogel. 2007. An efficient two-pass approach to synchronous-CFG driven statistical MT. In *NAACL*.

Wu, Dekai. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3).

Zollmann, Andreas, Ashish Venugopal, Franz Och, and Jay Ponte. 2008. A systematic comparison of phrase-based, hierarchical and syntax-augmented statistical MT. In *COLING*.

Mixture-Modeling with Unsupervised Clusters for Domain Adaptation in Statistical Machine Translation

Rico Sennrich

Institute of Computational Linguistics
University of Zurich
Binzmühlestr. 14
CH-8050 Zürich
sennrich@cl.uzh.ch

Abstract

In Statistical Machine Translation, in-domain and out-of-domain training data are not always clearly delineated. This paper investigates how we can still use mixture-modeling techniques for domain adaptation in such cases. We apply unsupervised clustering methods to split the original training set, and then use mixture-modeling techniques to build a model adapted to a given target domain. We show that this approach improves performance over an unadapted baseline, and several alternative domain adaptation methods.

1 Introduction

As the availability of parallel data for Statistical Machine Translation (SMT) increases, new opportunities and challenges for domain adaptation arise. Some corpora may contain text from a variety of domains, especially if they are built from heterogeneous resources such as crawled web pages. Many domain adaptation techniques do not operate on a single text, but require multiple models which are then mixed.

We investigate domain adaptation in a scenario where we have a known target domain, including development and test data from this domain, but where there is only a single heterogeneous training corpus. While this training corpus does contain in-domain data, we assume that we have no supervised means of extracting it.

Our basic approach is divided into two steps. Firstly, we perform unsupervised clustering on the parallel training data to obtain a given number of clusters. Secondly, we apply domain adaptation

algorithms to compute a model from these clusters that is adapted to the development set.

2 Related Work

The general idea in domain adaptation is to obtain models that are specifically optimized for best performance in one domain, with a potentially negative effect on its performance for other domains. The classical domain adaptation scenario consists of a (small) in-domain corpus, a (large) out-of-domain corpus, and in-domain development and test sets. Mixture-modeling approaches such as (Koehn and Schroeder, 2007; Foster and Kuhn, 2007; Sennrich, 2012) fall into this category.

We will here give an overview of adaptation techniques that assume less prior knowledge about the training set and/or target domains.

Yamamoto and Sumita (2008) operate without any predetermined domains, and without assuming that either the training or the test data is homogeneous. They cluster the training text into k clusters, and use unsupervised domain selection to translate each test set sentence by a cluster-specific model.

Finch and Sumita (2008) distinguish between two classes of sentences: questions and declaratives (i.e. non-questions). They split the training corpus automatically according to a simple rule (does the target sentence end with '?'), and for decoding use a linear interpolation of the class-specific and a general model, the interpolation weight depending on the class membership of each sentence.

Banerjee et al. (2010) focus on a scenario in which the domains of the training texts are known, whereas the test sets are a mix of two domains. They use a sentence-level classifier to translate each sentence with a domain-specific SMT system.

$$\bar{K}_m(x, x') = \sum_{n=1}^m \sum_{u \in \Sigma^n} \frac{f_x(u)}{\sqrt{\sum_{v \in \Sigma^n} f_x(v)}} \frac{f_{x'}(u)}{\sqrt{\sum_{v \in \Sigma^n} f_{x'}(v)}} \quad (1)$$

$$\bar{K}_m(x, x') = \sum_{n=1}^m \sum_{u \in \Sigma^n} \sqrt{\frac{f_x(u)}{\sum_{v \in \Sigma^n} f_x(v)} \frac{f_{x'}(u)}{\sum_{v \in \Sigma^n} f_{x'}(v)}} \quad (2)$$

Eck, Vogel and Waibel (2004) use information retrieval techniques to find the sentences in a parallel corpus that are closest to the translation input, then use the corresponding target sentences to build a language model. Their approach is similar to that of Yamamoto and Sumita (2008) in that both try to adapt models in a fully unsupervised manner. The main difference is that Yamamoto and Sumita (2008) compute the clusters (and the cluster-specific models) offline, and only do cluster prediction online, whereas in (Eck et al., 2004), the whole adaptation process, i.e. selecting a subset of training data, training a model, and translating with the specific model, happens online.

We will focus on a scenario which is slightly different from these prior studies in that we want to build a translation system for a specific target domain, but with in-domain and out-of-domain training data being mixed in a heterogeneous training set. For such a scenario, none of the outlined approaches are a perfect fit. Mixture-modeling techniques presume the existence of multiple models to mix, a condition which is not met in this scenario. The unsupervised methods, on the other hand, do not use sophisticated adaptation techniques, mostly because the target domain is unknown. We will test a hybrid approach that combines unsupervised methods to cluster the training text with known mixture-modeling techniques to obtain a model adapted to the target domain.

3 Clustering

We compare two unsupervised sentence clustering algorithms in order to split the training text into clusters that can later be recombined. Both algorithms are instances of k -means clustering, but with different distance functions. Yamamoto and Sumita (2008) use language models as centroids, trained on all sentences in a cluster, and the language model entropy as the distance between each sentence and cluster. Andrés-Ferrer et al. (2010) use word-sequence-kernels (WSK) (Cancedda et al., 2003) as distance metric between two docu-

ments. We initially followed their proposed normalization of the WSK, reproduced in equation 1. $f_x(u)$ is the frequency of the n -gram u in document/sentence x .¹ Unfortunately, the normalization in the proposed equation is flawed and causes a bias towards assigning sentences to the largest cluster. The WSK should be normalized so that the string a is (at least) as similar to itself as to $a a$ (if we only consider unigrams). However, $\frac{1}{\sqrt{1}} \frac{1}{\sqrt{1}} < \frac{1}{\sqrt{1}} \frac{2}{\sqrt{2}}$. We use an alternative normalization, shown in equation 2, that has no such numerical bias.

Both algorithms are initialized with randomly generated clusters, and both can be expanded to clustering sentence pairs by taking the sum of the distance on both language sides. In terms of n -gram length, we follow the respective authors' practice, using unigram models for the implementation of (Yamamoto and Sumita, 2008), and $m = 2$ for equation 2. Note that the clustering algorithm has the objective of minimizing LM entropy, whereas the WSK is a similarity function and thus is maximized.

3.1 Exponential Smoothing

One drawback of sentence-level clustering is that cluster assignment is made on the basis of very little information, i.e. the sentence itself. If we assume that the domain of a text does not rapidly change between sentences, it is sensible to consider a larger context for clustering.

We achieve this by using an exponentially decaying score for cluster assignment.² In the baseline without exponential decay (equation 3), we assign the sentence pair i to the cluster c that

¹For the full motivation of the equation, see (Andrés-Ferrer et al., 2010). In short, for all n -grams up to a maximum length m , the kernel sums over the product of their normalized frequency in two given documents.

²The most similar use of an exponential decay that we are aware of is by Zhong (2005), who proposes exponential decay to reduce the contribution of history data in a text stream clustering algorithm. However, the exponential decay affects a different component, namely the centroids, and does not serve the same purpose as our proposal.

minimizes the distance (i.e. the LM entropy or the negative WSK score).

$$\hat{c}_i = \arg \min_c d(i, c) \quad (3)$$

In equation 4, the distance of sentence pair i to cluster c is smoothed by the weighted average of the distance of each sentence j to c , with the weight exponentially decaying as the textual distance between i and j increases, and with the decay factor λ determining how fast the weight decays.

$$\hat{c}_i = \arg \min_c \sum_{j=1}^n d(j, c) \cdot \lambda^{|i-j|} \quad (4)$$

Note that the equation is two-sided, meaning that both previous and subsequent sentences are considered for the assignment.

Algorithmically, two-sided exponential smoothing only slows down cluster assignment by a constant factor; we do not need to sum over all sentences for each assignment, but can store the weighted distance of all previous sentences in a single variable. Algorithm 1 shows the smoothed assignment step for n sentences and k clusters.

Algorithm 1 Cluster assignment with decay

Ensure: $0 \leq \text{decay} \leq 1$

```

1: let  $d(x, y)$  be a distance function for a sentence
    $x$  and a centroid  $y$ 
2: let  $d\_min[n], d\_curr[n], \hat{c}[n]$  be arrays
3: set all elements of  $d\_min$  to  $\infty$ 
4: for  $c = 0$  to  $k$  do
5:    $cache \leftarrow 0$ 
6:   set all elements of  $d\_curr$  to 0
7:   for  $i = 0$  to  $n$  do
8:      $cache \leftarrow \text{decay} * cache$ 
9:      $cache \leftarrow cache + d(i, c)$ 
10:     $d\_curr[i] \leftarrow cache$ 
11:   end for
12:    $cache \leftarrow 0$ 
13:   for  $i = n$  to 0 do
14:      $cache \leftarrow \text{decay} * cache$ 
15:      $d\_curr[i] \leftarrow d\_curr[i] + cache$ 
16:     if  $d\_curr[i] < d\_min[i]$  then
17:        $d\_min[i] \leftarrow d\_curr[i]$ 
18:        $\hat{c}[i] \leftarrow c$ 
19:     end if
20:      $cache \leftarrow cache + d(i, c)$ 
21:   end for
22: end for

```

Note that the decay factor λ determines the extent of smoothing, i.e. how strongly context is taken into account for the assignment of each sentence. A decay factor of 0 corresponds to the unsmoothed sentence-level score (with $0^0 = 1$). With a decay factor of 1, the algorithm returns the same distance for all sentence pairs. We use a decay factor of 0.5 throughout the experiments. This is a relatively fast decay: one third of the score is determined by the sentence itself; two thirds by the sentence and its two neighboring sentences. What decay factor is optimal may depend on the properties of the text, i.e. how quickly documents and/or domains change, so we will not evaluate different decay factors in this paper.

We could extend the algorithm to reset the cache to 0 whenever we cross a known document boundary, and thus implement document-level scoring (with a decay factor of 1), or a hybrid (with a decay factor between 0 and 1). We did not do this since we want to demonstrate that the approach does not require document boundaries in the training text.

Another point to note is that we slightly modify the LM entropy method by normalizing entropy by sentence length, which ensures that longer sentences have no inflated effect on their neighbors' cluster assignment.

4 Model Combination

Having split the training text into clusters, there are various possibilities to exploit them. Yamamoto and Sumita (2008) use each cluster to train a cluster-specific model, which they interpolate with a general model, using a constant interpolation coefficient. Translating a text then consists of predicting the cluster of each sentence, then translating it with this cluster-specific model. If we make the assumption that the test set is relatively homogeneous, with all sentences belonging to the same domain, we can perform a more sophisticated adaptation to this target domain.

One potential shortcoming of the algorithm in (Yamamoto and Sumita, 2008) is that their domain prediction has little information to base its prediction on, and thus may not choose the best cluster. Additionally to predicting the domain for each sentence, we will test a document-level domain prediction, i.e. selecting the cluster with the shortest distance to the whole test set. Even this might be suboptimal if the number of clusters is high. In this case, we can expect relevant data to

be distributed over multiple clusters, in which case it might be beneficial to not be restricted to one cluster-specific model.

A second shortcoming is the lack of model optimization. Yamamoto and Sumita (2008) set the interpolation weights between the cluster-specific model and the general one manually after some preliminary experiments, and re-used the model parameters from the general model for all experiments. Specifically, they use linear interpolation with interpolation coefficients of 0.7 and 0.3 for the cluster-specific and the general translation model, respectively, and a log-linear combination for language models, with a slightly lower weight for the domain-specific (0.4) than the general (0.6) model.

Both the inability to consider multiple relevant datasets and the need to manually set model weights can be solved by using automatic mixture-model methods. We will experiment with automatic adaptation methods that use perplexity minimization to produce domain-specific models given a development set from the domain. The first step is again to train cluster-specific translation and language models, which we then recombine into a single adapted model. We use a linear interpolation with the interpolation coefficients set through perplexity minimization for language model and translation model adaptation, which has been demonstrated to be a successful technique in SMT (Foster and Kuhn, 2007). For translation model interpolation, we use the approach described in (Sennrich, 2012), optimizing each translation model feature separately on a parallel development set.

The optimization itself is convex, which means that we can easily apply it to a high number of clusters. The biggest risk is that the weight vector will be overfitted if we optimize it for a high number of small models. Finally, we set new log-linear SMT weights through MERT (Och and Ney, 2003) for each experiment.

5 Experiments

The main questions that we want to answer in our experiments are:

1. How well does unsupervised clustering split a heterogeneous training text according to its domains? How are the results affected by different distance functions and smoothing?

Data set	sentences	words (fr)
Alpine (in-domain)	200k	4 400k
Europarl	1 500k	44 000k
JRC Acquis	1 100k	24 000k
OpenSubtitles v2	2 300k	18 000k
Total train	5 100k	90 400k
Dev (perplexity)	1424	33 000
Dev (MERT)	1000	20 000
Test	991	21 000

Table 1: Parallel data sets for German – French translation task.

2. How much translation quality do we lose or gain from mixture-modeling based on unsupervised clusters, compared to a scenario where we start with multiple domain-specific corpora.

5.1 Data and Methods

We perform the experiments on a German–French data set. The parallel data sets used are listed in table 1. The in-domain corpus is a collection of Alpine Club publications (Volk et al., 2010). As parallel out-of-domain data sets, we use Europarl, a collection of parliamentary proceedings (Koehn, 2005), JRC-Acquis, a collection of legislative texts (Steinberger et al., 2006), and OpenSubtitles v2, a parallel corpus extracted from film subtitles³ (Tiedemann, 2009).

For language model training, we used the same 90 million word corpus, plus, on the target side, the news corpus from WMT 2011 (appr. 610 million tokens), and appr. 8 million tokens monolingual in-domain data. We used the following language model settings: for clustering, unigram language models. For domain selection, 3-gram language models with Good-Turing smoothing. For translation, 5-gram language models with interpolated Kneser-Ney smoothing. We clustered additional target language data with the method described in (Yamamoto and Sumita, 2008), i.e. one cluster assignment step, starting from the bilingual clusters, and not assigning any sentences which are closest to the general LM.

For the clustering experiments, these data sets are concatenated to simulate a heterogeneous training set. The relative amount of in-domain data in the training sets is 2% (monolingual) and 4% (parallel). Note that this makes success of our method

³<http://www.opensubtitles.org>

more likely than in scenarios where there is no in-domain training data in the training set. We do not claim that any heterogeneous training text is equally suited for domain adaptation.

In (Andrés-Ferrer et al., 2010), clustering quality is measured intrinsically, i.e. by calculating the intra-cluster language model perplexity. In our evaluation, we use an extrinsic evaluation that compares the resulting clusters to the original four parallel datasets. For this evaluation, we assume that clustering is felicitous if it clusters sentences from the same original data set together. We measure this using entropy (equation 5), with N being the total number of sentence pairs and $orig(i)$ being the corpus to which sentence i originally belonged. $p_c(orig(i))$ is the probability that a sentence in cluster c is originally from corpus $orig(i)$, estimated through relative frequency.

$$H(X) = - \sum_{c=0}^k \sum_{i \in c} \frac{1}{N} \log_2 p_c(orig(i)) \quad (5)$$

If a cluster only contains sentences from one corpus, its entropy is 0. The baseline is a uniform distribution, which corresponds to an entropy of 1.698 (with the data sets from table 1).

The second evaluation is a translation task. In terms of tools and techniques used, we mostly adhere to the work flow described for the WMT 2011 baseline system⁴. The main tools are Moses (Koehn et al., 2007), SRILM (Stolcke, 2002), and GIZA++ (Och and Ney, 2003), with settings as described in the WMT 2011 guide. One exception is that we additionally filter the phrase table according to statistical significance tests, as described by (Johnson et al., 2007). We use two different development sets, one for domain adaptation (through perplexity optimization) and one for MERT, in order to rule out that MERT gives too much weight to the language and translation model which are optimized on the same dataset.

We measure translation performance through BLEU (Papineni et al., 2002) and METEOR 1.3 (Denkowski and Lavie, 2011). All results are lowercased and tokenized, measured with five independent runs of MERT (Och and Ney, 2003). We perform significance testing with MultEval (Clark et al., 2011), which uses approximate randomization to account for optimizer instability. Note that there are other causes of instability unaccounted

⁴<http://www.statmt.org/wmt11/baseline.html>

distance	k	entropy		itr. (avg)
		mean	stdev	
no smoothing				
WSK	10	0.727	0.022	21.4
LM	10	0.439	0.034	20.2
LM	100	0.344	0.008	38.8
exponential smoothing				
WSK	10	0.263	0.048	13.8
LM	10	0.112	0.016	10.4
LM	100	0.064	0.013	9.0

Table 2: Entropy comparison between clustering with different distance functions (with or without smoothing), and different numbers of clusters (k). Mean, standard deviation, and average number of iterations out of 5 runs are reported. WSK: word sequence kernels; LM: language model entropy

for, e.g. the randomness of clustering. Word alignment has been kept constant across all experiments.

5.2 Results

In all experiments, we perform k -means clustering with $k = 10$ and $k = 100$. A higher number of clusters typically increases the homogeneity of the resulting clusters, and may boost performance by allowing us to give high weights to very specific subdomains of the training set. On the downside, clusters will be smaller on average, which exacerbates data sparseness problems. In the trivial case, having one sentence per cluster results in an entropy of 0, but this granularity would be unsuitable for the domain adaptation methods that we evaluate because of data sparseness.

Table 2 shows entropy of both sentence-level clustering and exponential smoothing with word sequence kernels and LM entropy as distance functions. All methods achieve a strong reduction of entropy over the uniform baseline (1.698), but LM entropy as a distance measure outperforms word sequence kernels, with a mean entropy of 0.439 compared to 0.727 for 10 clusters. In all experiments, exponential smoothing reduces the entropy of the resulting clusters even further. With LM entropy as distance function, it is reduced from 0.439 to 0.112 for $k = 10$, and from 0.344 to 0.064 with $k = 100$. A second advantage of smoothing is that the algorithm converges faster, and reduces the number of iterations by a factor of 2–4. Thus, smoothing seems a good choice because

system	BLEU	METEOR
general	18.5	37.3
adapted TM	18.8	37.8
adapted LM	18.8	37.8
adapted TM & LM	18.6	37.9

Table 3: Baseline SMT results DE–FR. Concatenation of all data and using domain adaptation with original four datasets.

the smoothed algorithm is both faster and better at clustering sentences from the same original dataset into the same cluster. Whether this leads to better SMT performance is tested in the evaluation of translation performance.

We can compare translation performance to four baselines, shown in table 3. The general system (without domain adaptation) performs worst, with a BLEU score of 18.5 and a METEOR score of 37.3. Both TM and LM adaptation significantly increase scores by 0.3 BLEU and 0.5 METEOR points. The system that combines TM and LM adaptation is not significantly different from the systems with only one model adapted in terms of BLEU, but performs best in terms of METEOR (0.6 points better than the general model).

For the experimental systems, we limit ourselves to LM entropy as distance function, and vary a number of parameters. k , the number of clusters, is 10 in table 4, and 100 in table 5. For both k , we test clustering without smoothing (sentence-level clustering) and with exponential smoothing and a decay factor of 0.5. For each variation of these parameters, we pick a single clustering run at random. For model combination, we contrast the approach by Yamamoto and Sumita (2008) (i.e. domain prediction with a fixed interpolation), and the mixture models described in section 4, i.e. perplexity-minimization to find the optimal weights for the linear interpolation of the language and translation model (Sennrich, 2012).

In sections 3.1 and 4, we have identified possible shortcomings of the original approach by (Yamamoto and Sumita, 2008), and will now reiterate and discuss them.

Firstly, we have hypothesized that unsmoothed sentence-level clustering may fail to cluster in-domain data together, and have proposed exponential smoothing. The entropy results in table 2 support this hypothesis; if we look at translation results with document-level domain predic-

tion, the performance differences are small. A look at the clusters that are selected in domain prediction shows that smoothing improved homogeneity (180 000 in-domain / 20 000 out-of-domain sentence pairs) over an unsmoothed sentence-level clustering (146 000 in-domain / 90 000 out-of-domain), but both approaches cluster the majority of the 200 000 in-domain sentence pairs together and outperform the unadapted baseline.

Secondly, we suspected that domain prediction on a sentence-level would suffer from similar data-sparseness problems, and not pick the optimal cluster for translation. With 10 clusters, there is little difference between sentence-level and document-level domain prediction, both in terms of performance and the cluster that is predicted in domain prediction. With (smoothed or unsmoothed) sentence-level prediction, 80-90% of test set sentences are predicted to belong to the same cluster. With 100 clusters, the opposite of our hypothesis is true. Document-level domain prediction performs worse than (smoothed or unsmoothed) sentence-level domain prediction, and no better than the unadapted baseline. For the interpretation of this result, we must also consider the mixture-modeling results.

Adapting models through perplexity optimization performs better than or equally well as the methods with domain prediction and a fixed interpolation between the domain-specific and the general model. This is true for both domain prediction methods, and both smoothed and unsmoothed clustering. The best result is obtained with $k = 10$ and smoothed clustering, with a BLEU score of 19.2 and a METEOR score of 38.3, which is 0.7 BLEU points and 1 METEOR points above the unadapted baseline. The system also beats the adapted baseline, which uses the same model combination algorithm on the original four datasets, by 0.6 BLEU points and 0.4 METEOR points, and the approach by (Yamamoto and Sumita, 2008) (sentence-level clustering and domain prediction) by 0.3 BLEU points and 0.4 METEOR points.

With 100 clusters, perplexity minimization yields no further performance gains, but remains significantly better than the systems with domain prediction and the baseline systems. As to the reason why document-level domain prediction performs poorly with 100 clusters, the main problem is that relevant data is spread out over multiple clusters, and that only a small amount of relevant

clustering	domain prediction	model combination	adapted TM		adapted TM & LM	
			BLEU	METEOR	BLEU	METEOR
sentence-level	sentence-level	fixed weights	18.7	37.6	18.9	37.9
	document-level	fixed weights	18.8	37.7	18.9	37.9
	-	perplexity	18.8	38.0	18.9	38.2
smoothed	smoothed	fixed weights	18.9	37.8	19.0	38.0
	document-level	fixed weights	18.9	37.8	19.0	38.1
	-	perplexity	19.1	38.3	19.2	38.3

Table 4: SMT results DE–FR based on clustered training data ($k = 10$).

clustering	domain prediction	model combination	adapted TM		adapted TM & LM	
			BLEU	METEOR	BLEU	METEOR
sentence-level	sentence-level	fixed weights	18.8	37.7	18.6	37.6
	document-level	fixed weights	18.5	37.5	18.5	37.5
	-	perplexity	19.0	38.0	19.0	38.3
smoothed	smoothed	fixed weights	18.6	37.5	18.5	37.5
	document-level	fixed weights	18.6	37.5	18.4	37.4
	-	perplexity	19.1	38.1	19.1	38.2

Table 5: SMT results DE–FR based on clustered training data ($k = 100$).

data can be considered with document-level domain prediction. Sentence-level domain prediction avoids this problem by choosing different cluster-specific models to translate different sentences, the perplexity mixture-models by being able to give high weights to multiple cluster-specific models.

6 Conclusion

We demonstrate that it is possible to apply mixture-modeling techniques to models that are obtained through unsupervised clustering of a heterogeneous training text. We obtained a modest performance boost from applying mixture-modeling on the clusters rather than the original parallel corpora. The main advantage of the clustering step, however, is that it reduces the requirements for mixture-modeling, eliminating the need for a homogeneous, in-domain training corpus, and only requiring a development set from the target domain. It is thus more general and could be applied to monolithic, heterogeneous data collections.

Compared to the fully unsupervised method by (Yamamoto and Sumita, 2008), we observed small performance improvements of up to 0.3 BLEU points. In a closed-domain setting, the approach also has the advantage of moving the domain adaptation cost into the offline phase, and not requiring a domain prediction phase and multiple models during decoding. To support multiple target do-

main, the approach could be combined with that of (Banerjee et al., 2010), who discuss the problem of translating texts that contain sentences from multiple (known) domains.

We also propose exponential smoothing during cluster assignment to better capture slow-changing textual properties such as their domain membership, and to combat data sparseness issues when having to do an assignment decision based on short sentences. While the effects on our translation experiments were small, the increased homogeneity of the resulting clusters and the faster speed of convergence indicate that smoothing is a beneficial enhancement to sentence-level k -means clustering.

Acknowledgments

This research was funded by the Swiss National Science Foundation, grant 105215_126999.

References

- Andrés-Ferrer, Jesús, Germán Sanchis-Trilles, and Francisco Casacuberta. 2010. Similarity word-sequence kernels for sentence clustering. In *Proceedings of the 2010 joint IAPR international conference on Structural, syntactic, and statistical pattern recognition*, pages 610–619, Berlin, Heidelberg. Springer-Verlag.
- Banerjee, Pratyush, Jinhua Du, Baoli Li, Sudip Kumar Naskar, Andy Way, and Josef Van Genabith. 2010.

- Combining multi-domain statistical machine translation models using automatic classifiers. In *9th Conference of the Association for Machine Translation in the Americas (AMTA 2010)*.
- Cancedda, Nicola, Eric Gaussier, Cyril Goutte, and Jean Michel Renders. 2003. Word sequence kernels. *J. Mach. Learn. Res.*, 3:1059–1082, March.
- Clark, Jonathan H., Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 176–181, Portland, Oregon, USA, June. Association for Computational Linguistics.
- Denkowski, Michael and Alon Lavie. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. In *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*.
- Eck, Matthias, Stephan Vogel, and Alex Waibel. 2004. Language model adaptation for statistical machine translation based on information retrieval. In *4th International Conference on Languages Resources and Evaluation (LREC 2004)*.
- Finch, Andrew and Eiichiro Sumita. 2008. Dynamic model interpolation for statistical machine translation. In *Proceedings of the Third Workshop on Statistical Machine Translation, StatMT '08*, pages 208–215, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Foster, George and Roland Kuhn. 2007. Mixture-model adaptation for SMT. In *Proceedings of the Second Workshop on Statistical Machine Translation, StatMT '07*, pages 128–135, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Johnson, Howard, Joel Martin, George Foster, and Roland Kuhn. 2007. Improving translation quality by discarding most of the phrasetable. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 967–975, Prague, Czech Republic, June. Association for Computational Linguistics.
- Koehn, Philipp and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation, StatMT '07*, pages 224–227, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June. Association for Computational Linguistics.
- Koehn, Philipp. 2005. Europarl: A parallel corpus for statistical machine translation. In *Machine Translation Summit X*, pages 79–86, Phuket, Thailand.
- Och, Franz Josef and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.
- Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.
- Sennrich, Rico. 2012. Perplexity minimization for translation model domain adaptation in statistical machine translation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 539–549, Avignon, France. Association for Computational Linguistics.
- Steinberger, Ralf, Bruno Pouliquen, Anna Widiger, Camelia Ignat, Tomaz Erjavec, Dan Tufis, and Daniel Varga. 2006. The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006)*.
- Stolcke, A. 2002. SRILM – An Extensible Language Modeling Toolkit. In *Seventh International Conference on Spoken Language Processing*, pages 901–904, Denver, CO, USA.
- Tiedemann, Jörg. 2009. News from OPUS - a collection of multilingual parallel corpora with tools and interfaces. In Nicolov, N., K. Bontcheva, G. Angelova, and R. Mitkov, editors, *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia, Borovets, Bulgaria.
- Volk, Martin, Noah Bubenhofer, Adrian Althaus, Maya Bangerter, Lenz Furrer, and Beni Ruef. 2010. Challenges in building a multilingual alpine heritage corpus. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).
- Yamamoto, Hirofumi and Eiichiro Sumita. 2008. Bilingual cluster based models for statistical machine translation. *IEICE - Trans. Inf. Syst.*, E91-D:588–597, March.
- Zhong, S. 2005. Efficient streaming text clustering. *Neural Networks*, 18(5-6):790–798, July.

Extending CCG-based Syntactic Constraints in Hierarchical Phrase-Based SMT

Hala Almaghout
CNGL, School of Computing
Dublin City University
Dublin, Ireland

halmaghout@computing.dcu.ie

Jie Jiang
Applied Language Solutions
Delph
UK

jie.jiang@appliedlanguage.com

Andy Way
Applied Language Solutions
Delph
UK

andy.way@appliedlanguage.com

Abstract

In this paper, we describe two approaches to extending syntactic constraints in the Hierarchical Phrase-Based (HPB) Statistical Machine Translation (SMT) model using Combinatory Categorical Grammar (CCG). These extensions target the limitations of previous syntax-augmented HPB SMT systems which limit the coverage of the syntactic constraints applied. We present experiments on Arabic–English and Chinese–English translation. Our experiments show that using extended CCG labels helps to increase nonterminal label coverage and achieve significant improvements over the baseline for Arabic–English translation. In addition, combining extended CCG labels with CCG-augmented glue grammar helps to improve the performance of the Chinese–English translation over the baseline systems.

1 Introduction

Hierarchical Phrase-Based (HPB) Statistical Machine Translation (SMT) (Chiang, 2005) has been demonstrated to be one of the most successful SMT approaches nowadays. Its main idea is to imitate Context-Free Grammar (CFG) production rules in modelling translation rules while maintaining the strength of statistically extracted phrases. However, HPB SMT only models the hierarchical aspect of the language and does not use any linguistic information in rule extraction. A set of approaches (Zollmann and Venugopal, 2006; Almaghout et al., 2010) have tried to incorporate

syntactic information extracted according to different grammar theories in the HPB SMT model by annotating phrases and nonterminals with syntactic labels. These systems face many challenges in integrating their syntax-based constraints with the syntax-free statistically extracted HPB SMT translation grammar, which limits the coverage of these syntactic constraints and thus minimizes the benefit obtained from applying them.

In this paper, we try to extend the scope of target-side syntactic constraints in syntax-augmented HPB SMT. More specifically, we try to exploit the flexibility of Combinatory Categorical Grammar (CCG) (Steedman, 2000) to increase the coverage of syntactic labels used to label phrases and nonterminals in hierarchical rules. In addition, we augment HPB glue grammar rules with CCG combinatory operators with the aim of directing the decoding process towards building a full parse tree of the translation output. We apply these constraints in a soft manner through a feature in the log-linear model (Venugopal et al., 2009).

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 gives an introduction to HPB SMT. Section 4 introduces CCG. Section 5 describes our approach. Section 6 presents our experiments. Finally, Section 7 concludes and provides avenues for future work.

2 Related Work

Syntax Augmented Machine Translation (SAMT) (Zollmann and Venugopal, 2006) tries to improve the grammaticality of the HPB SMT translation output by attaching syntactic labels to target-side phrases and nonterminals. These labels are extracted from context-free phrase structure grammar parse trees of the target-side of the parallel corpus. The function of these

syntactic labels is to impose syntactic constraints on phrases replacing nonterminals during decoding, allowing this replacement only when the labels of the nonterminal and the replacing phrase match. CCG-augmented HPB (Almaghout et al., 2010) follows the SAMT approach in labelling nonterminals with syntactic labels. It extracts CCG-based labels from CCG forest trees of the target-side of the parallel corpus. Almaghout et al. (2011) use contextual information presented in CCG categories to extract syntactic labels for nonterminals and phrases in the HPB SMT translation model. Birch et al. (2007) use CCG supertags as a source and target factor in the factored Phrase-Based (PB) SMT translation model (Koehn and Hoang, 2007). Hassan et al. (2009) integrate target-side CCG incremental parsing in the Direct Translation Model (DTM2). They also extract a set of syntactic features based on CCG supertags, combinatory operators and parsing states. This helps to build a fully connected parsing structure during decoding and prune hypotheses which do not constitute a valid parsing state.

Recently, applying syntactic constraints in syntax-augmented HPB SMT systems in a soft manner has been demonstrated to improve the performance of these systems (Venugopal et al., 2009; Chiang, 2010). This means that the derivations which violate the syntactic constraints imposed by the model are not prevented per se, but the system learns to favour more grammatical translations. Strong syntactic constraints impose restrictions on the translation search space and consequently have a negative impact on performance. Venugopal et al. (2009) transform the syntactic constraints in the SAMT translation model to a syntactic feature integrated into the log-linear model. They use an unlabelled translation model during decoding. Another SAMT-based syntactic model, which measures the probability of different labellings of each hierarchical rule, is used to calculate the value of the syntactic feature at each nonterminal replacement during decoding.

3 Hierarchical Phrase-Based SMT

HPB SMT (Chiang, 2005) is a tree-based model which extracts a synchronous CFG automatically from the training corpus. HPB SMT extracts hierarchical rules – the fundamental translation units in the HPB model – from phrases extracted according to the PB model (Koehn et al., 2003). Thus,

hierarchical rules have the strengths of statistically extracted continuous phrases plus the ability to translate discontinuous phrases and learn phrase-reordering without a separate reordering model. The HPB SMT model has two types of rules: hierarchical rules and glue grammar rules. Hierarchical rules are rewrite rules with aligned pairs of right-hand sides, taking the following form:

$$X \rightarrow \langle \alpha, \beta, \sim \rangle \quad (1)$$

where X is a non-terminal, α and β are both strings of terminals and non-terminals, and \sim is a one-to-one correspondence between non-terminal occurrences in α and β . Hierarchical rules are extracted from the training corpus by subtracting continuous phrase-pairs attested in the translation table recursively from longer phrases and replacing them with the nonterminal symbol X . Nonterminals in hierarchical rules act as placeholders that are replaced with other phrases during translation in a bottom-up fashion.

Glue grammar rules perform monotone phrase concatenation, which means that they combine target phrases together without performing any reordering. They consist of the following two rules:

$$S \rightarrow \langle S X, S X \rangle \quad (2)$$

$$S \rightarrow \langle X, X \rangle \quad (3)$$

Their main role is to produce translation when no possible hierarchical rule can be applied. They are also used to reduce the complexity of chart decoding by limiting the application of the hierarchical rules to a certain limit (12 words in our experiments, cf. Section 6) above which only glue grammar rules are applied. Glue grammar rules can also be applied below this limit but their application cannot alternate with hierarchical rules, and they always form a left-balanced binary tree on top of the hierarchical rules in the derivation tree.

4 Combinatory Categorical Grammar

CCG (Steedman, 2000) is a grammar formalism which consists of a lexicon that pairs words with lexical categories (supertags, cf. Bangalore and Joshi (1999)) and a set of combinatory rules which specify how the categories are combined. A supertag is a rich syntactic description that specifies the local syntactic context of the word at the lexical level in the form of a set of arguments. Most of the CCG grammar is contained in the lexicon,

which is why CCG has simpler combinatory rules compared to CFG productions.

CCG categories are divided into atomic and complex categories. Examples of atomic categories are S (sentence), N (noun), NP (noun phrase), etc. Complex categories such as $S \backslash NP$ and $(S \backslash NP) / NP$ are functions which specify the type and directionality of their arguments and results. CCG builds a parse tree for a sentence by combining CCG categories using a set of binary combinatory operators. Since most of the CCG grammar resides in the lexicon, CCG has a simple set of combinatory operators. Figure 1 shows a CCG parse tree of the English sentence *Would you like cream and sugar in your coffee ?*

4.1 CCG and SMT

CCG has many unique qualities which make it an attractive grammar formalism to be incorporated into SMT systems. First, CCG allows for flexible structures thanks to its combinatory operators. Thus it is possible to assign a CCG category to phrases which do not represent standard syntactic constituents. This is an important feature for SMT systems as SMT phrases are statistically extracted, and do not necessarily correspond to syntactic constituents. Second, CCG supertags present rich syntactic information at the lexical level about the dependents and local context of each word in the sentence. Therefore, CCG supertags reflect important information about the syntactic structure of the sentence without the need to build a full parse tree. This allows SMT systems to build grammaticality metrics based on examining sequences of CCG supertags of the words of the translation output. Finally, CCG can be efficiently parsed thanks to the process of supertagging (Bangalore and Joshi, 1999), which assigns supertags to the words of the sentence before parsing. This reduces the parsing search space significantly and is especially important for computationally complex SMT systems.

5 Our Approach

5.1 Motivation

Although incorporating syntax into HPB SMT has been demonstrated to improve its translation quality (Zollmann and Venugopal, 2006), the coverage of the syntactic constraints in syntax-augmented HPB SMT systems is limited because they include only part of the phrases and the grammar in the model. The mismatch between the notion of the

phrase in SMT systems and grammar formalisms leaves many phrases in syntax-augmented HPB SMT systems unlabelled. Almaghout et al. (2010) show that CCG-augmented HPB SMT and SAMT systems fail to label 30% and 50% of the total phrases in the training corpus, respectively. Furthermore, syntax-augmented HPB SMT systems have always focused efforts on augmenting hierarchical rules with syntax, ignoring the other important part of the grammar which is glue grammar rules. Glue grammar rules constitute about 30% to 40% of the total rules used in the derivations, which means that they play an important role in the translation process. Bearing in mind that hierarchical rules have a limited span to reduce the complexity of chart decoding, the application of syntactic constraints is also limited for the same reason. Ignoring these aspects limits the scope of syntactic constraints in syntax-augmented HPB systems which in turn limits their effect on improving the grammaticality of translation output.

In our approach, we try to expand the scope of syntactic constraints in our CCG-augmented HPB system. To achieve this we follow a two-fold approach. First, we try to extend the notion of the syntactic label attached to nonterminal labels and phrases with the aim of increasing label coverage. Secondly, we augment glue grammar rules with CCG combinatory operators. We apply these enhancements in a soft way under the Preference Grammars paradigm for applying soft syntactic constraints in HPB SMT (Venugopal et al., 2009). Thus, we add a syntactic feature to the log-linear model which judges the grammaticality of each nonterminal replacement and glue grammar rule application. We will describe each research strand in detail in the following sections.

5.2 Extended CCG-based Syntactic Labels

In SAMT, a set of CCG-like binary operators are used to increase the coverage of nonterminal labels. This is necessary as SAMT labels are extracted using phrase structure grammar, which has a small set of constituent labels that are insufficient to cover all the different syntactic structures of extracted phrases. Almaghout et al. (2010) use single-category CCG labels as nonterminal labels. Although CCG flexible structures allow a better label coverage than phrase structure grammar-based labels, using single-category CCG labels fails to label about one third of the total phrases. In or-

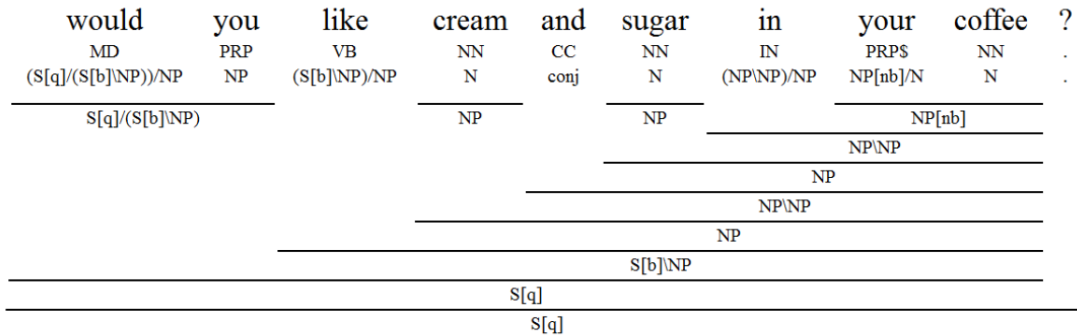


Figure 1: An example of a complete CCG parse tree of an English sentence.

cream and	N+conj
you like cream	S[b]
like cream and	S[b]\NP+conj
sugar in your	N+(NP\NP)/NP+NP[nb]/N

Figure 2: A set of phrases along with their extended CCG labels extracted from the CCG tree in Figure 1.

der to increase label coverage, we extend the definition of the nonterminal label to be composed of more than one CCG category. Therefore, if there is no single CCG category at the root of the trees which cover a phrase, the highest-scoring sequence of categories with a minimum number of CCG categories is extracted from CCG trees covering the phrase and used as the phrase label. Figure 2 shows a set of phrases extracted from the sentence illustrated in Figure 1 along with their extended CCG labels. In this example, the phrase *like cream and* has an extended CCG label composed of two categories: $S[b]\backslash NP+conj$. The CCG categories of the words *like* and *cream* are combined into the CCG category $S[b]\backslash NP$. However, this category cannot be combined with the *conj* category.

We define the degree of the extended label to be the number of CCG categories in the label. In our previous example, the extended CCG label of the phrase *sugar in your* is of degree three while the phrase *cream and* is of degree two. The degree of the system which uses extended CCG labels is defined to be the maximum degree of the labels in the model.

5.3 CCG-augmented Glue Grammar

Instead of concatenating phrases during glue grammar rule application without applying any syntactic constraints, we try to augment glue grammar rules with CCG combinatory operators. CCG

combinatory operators are binary operators, which makes them suitable to be applied on glue grammar rules which are also binary rules. First, we change the definition of the glue grammar rule (2) as follows:

$$X \rightarrow \langle X X, X X \rangle \tag{4}$$

This removes the left-balance constraints from the construction of glue-grammar rule application. Additionally, this rule allows the application of glue grammar rules and hierarchical rules to alternate, which gives better flexibility. Secondly, we build a metric which judges the grammaticality of concatenating two phrases at each glue grammar rule application based on their extended CCG labels. The calculation of this grammaticality metric is based on an extended CCG label model. This model is extracted using relative frequency counts from the target-side of the training corpus which is annotated with extended CCG labels for each subphrase in each sentence.

Whenever two phrases are concatenated under glue grammar rule application, the following steps are applied to calculate the grammaticality features for each extended CCG label pair L1 and L2 from the first and second phrase, respectively:

- Simplify $L1+L2$ by applying all possible CCG combinatory operators on $L1+L2$ to derive the extended CCG label L with the minimum number of CCG categories.
- If the resulting label L from the previous step is composed of one CCG category, the two phrases are likely to constitute a grammatical phrase and the grammaticality feature is set to 1.
- Otherwise, the grammaticality feature is set to the probability of L according to the extended CCG labels model.

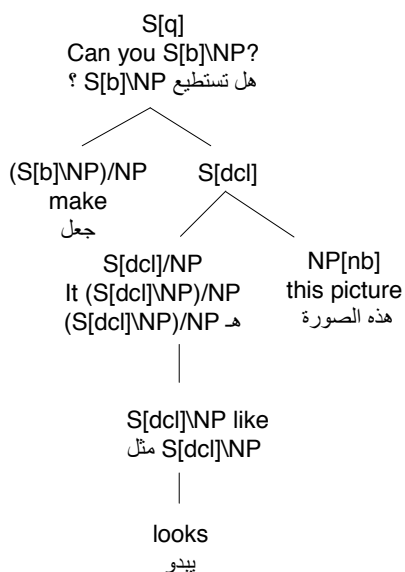


Figure 3: A derivation tree which shows the application of CCG-augmented glue grammar rules.

- Assign L to the phrase resulting from glue grammar rule application.

Augmenting glue grammar rules with CCG combinatory operators enables the building of a full parse tree of the translation output and extends the scope of the syntactic constraints to cover the whole translation output. The grammaticality feature helps to guide the decoding process by awarding the application of hierarchical and glue grammar rules which yield a grammatical output.

Figure 3 shows a derivation tree we obtain when translating a sentence from Arabic into English. Each node in the tree usually has more than one CCG label but the figure shows only the most probable label for the sake of simplicity. The resulting English translation is: *can you make it looks like this picture ?*. Although the translation is not totally grammatical, having the verb *look* in the wrong form *looks*, we can see how hierarchical and glue grammar rules participate in building a full parse tree which covers the whole translation output.

6 Experiments

In our experiments, we try to explore the effect of each method for extending the syntactic constraints in the HPB SMT system presented in Section 5. Sections 6.3.1 and 6.3.2 give the results for

each individual method. We then conduct experiments which examine the effect of combining both approaches in a single system.

6.1 Data Used

We used the data provided by the IWSLT 2010 evaluation campaign.¹ The Chinese–English training corpus consists of 55k sentence pairs from the IWSLT 2010 Chinese–English training data for the DIALOG task. The development and test sets are IWSLT evaluation data sets (500 sentence pairs for each) provided for the Chinese–English DIALOG task for 2008 and 2009. The development set has 15 references and the test set has 7 references. The Arabic–English training corpus consists of 20k sentence pairs from the IWSLT 2010 training data provided for the Arabic–English BTEC task. The development and test sets are the IWSLT evaluation data sets provided for the BTEC task for 2007 and 2008 evaluations, with 489 sentence pairs in the development set and 507 sentence pairs in the test set, respectively. The development set has 7 references and the test set has 16 references. All the English data used in our experiments is lower-cased and tokenized. The Arabic data is segmented according to the D3 segmentation scheme using MADA (Morphological Analysis and Disambiguation for Arabic).²

6.2 Baseline Systems

We have two baseline systems in our experiments: the HPB SMT baseline system and the CCG-augmented HPB SMT baseline system which uses single-category CCG labels and applies strong syntactic constraints (Almaghout et al., 2010). We built our HPB SMT baseline system using the Moses Chart Decoder.³ The GIZA++ toolkit⁴ is used to perform word and phrase alignment and the “grow-diag-final” refinement method is adopted (Koehn et al., 2003). Maximum phrase length and maximum rule span are both set to 12 words. The maximum span for the chart during decoding is set to 20 words, above which only glue grammar rules are applied. Hierarchical rules extracted contain up to 2 nonterminals. Minimum error rate training (Och, 2003) is performed to tune all our SMT systems. The 5-gram language model in all experiments was trained on the target side

¹<http://iwslt2010.fbk.eu/node/27>

²<http://www1.ccls.columbia.edu/MADA/>

³<http://www.statmt.org/ Moses/?n=Moses.SyntaxTutorial>

⁴<http://fjoch.com/GIZA++.html>

of the parallel corpus using the SRILM toolkit⁵ with modified Kneser-Ney smoothing. Our CCG-augmented HPB system was also built using the Moses Chart Decoder, which has an option to extract syntax-augmented rules from an annotated corpus. We used the same rule extraction and decoding settings as for the HPB baseline system. We use the CCG parser from C&C tools⁶ to parse the training data for our CCG-augmented HPB system experiments and to combine CCG categories during glue grammar rule application.

6.3 Experimental Results

6.3.1 Extended CCG Labels Experiments

In this section we examine the effect of our extended CCG labels. We try out extended labels of degrees ranging from one to five under soft syntactic constraints. Tables 1 and 2 show BLEU, TER and METEOR scores of extended CCG labels systems along with the number of different nonterminal labels estimated in thousands and the percentage of unlabelled nonterminals in the rule table of each system for Arabic–English and Chinese–English translation, respectively.

From Table 1, we can see that the 5-category CCG-augmented HPB SMT system is the best-performing system in terms of BLEU and TER scores, outperforming the HPB and CCG-augmented HPB baseline systems by 0.86 and 2.22 absolute BLEU points, which corresponds to a 1.63% and 4.3% relative improvement, respectively. The result of the paired bootstrap resampling test (Koehn, 2004) demonstrates that the improvement achieved over both baseline systems is statistically significant at p-level=0.05. Table 1 also shows that using soft syntactic constraints leads to significant improvements over the CCG-augmented HPB SMT baseline, which uses strong syntactic constraints. Furthermore, using extended CCG labels significantly decreases the percentage of unlabelled nonterminals in the rule table from 28% in the single-category system to 0.05% in the 5-category system.

Table 2 shows that the 3-category CCG-augmented HPB SMT system is the best-performing system in terms of BLEU and TER. The 3-category CCG-augmented HPB SMT system outperformed the HPB and CCG-augmented HPB SMT baseline systems by 1.65 and 3.93

System	BLEU	TER	MET	Lab	%X
HPB	52.90	31.06	71.51	-	-
CCG	51.54	32.32	70.33	0.5	28
CCG1	52.83	31.13	70.77	0.5	28
CCG2	53.38	30.92	70.60	8.0	6.3
CCG3	53.10	30.76	70.77	18	1.3
CCG4	53.09	30.76	70.62	23	0.3
CCG5	53.76	30.76	71.05	24	0.05

Table 1: Experimental results of CCG-augmented HPB systems with extended CCG labels from different degrees compared to the baseline systems for Arabic–English translation. Lab indicates the number of different labels used by each system (in thousands). %X indicates the percentage of unlabelled nonterminals in the rule table.

absolute BLEU points, which corresponds to a 3.4% and 8.5% relative improvement, respectively. The paired bootstrap resampling test demonstrates that these improvements are both significant at p-level=0.05. Similar to our Arabic–English experiments, using soft syntactic constraints helps to achieve significant improvements over the strong-constraints CCG-augmented HPB baseline system.

System	BLEU	TER	MET	Lab	%X
HPB	48.29	35.28	65.85	-	-
CCG	46.01	34.86	63.01	0.6	31
CCG1	49.73	34.04	66.66	0.6	31
CCG2	48.19	35.46	64.67	12	7.6
CCG3	49.94	34.02	66.29	30	1.7
CCG4	48.32	34.54	65.07	40	0.4
CCG5	49.44	34.10	65.76	43	0.09

Table 2: Experimental results of CCG-augmented HPB systems with extended CCG labels from different degrees compared to the baseline systems for Chinese–English translation along with the number of different labels and the percentage of unlabelled nonterminals in the model of each system.

6.3.2 CCG-augmented Glue Grammar Experiments

We examined the application of our CCG-augmented glue grammar rules using the single-category CCG labels under soft syntactic constraints. Tables 3 and 4 show the results of using CCG-augmented glue grammar for Arabic–

⁵<http://www-speech.sri.com/projects/srilm/>

⁶<http://svn.ask.it.usyd.edu.au/trac/candc/>

English and Chinese–English translation, respectively.

System	BLEU	TER	METEOR
CCG glue1	53.06	31.42	71.00

Table 3: Experimental results of the CCG-augmented HPB system which uses CCG-augmented glue grammar rules with single-category CCG labels for Arabic–English translation.

For both language pairs, CCG-augmentation for glue grammar rules failed to achieve any improvement over the best-performing systems obtained using extended CCG labels. Furthermore, we observe that using CCG-augmented glue grammar rules leads to a significant decrease in BLEU score for Chinese–English translation, even below the baseline performance.

System	BLEU	TER	METEOR
CCG glue1	45.65	36.64	62.91

Table 4: Experimental results of the CCG-augmented HPB system which uses CCG-augmented glue grammar rules with single-category CCG labels for Chinese–English translation.

6.3.3 Extension Approaches in Combination

In this section we try to combine both approaches to extending syntactic constraints described in this paper, namely (i) extended CCG labels and (ii) CCG-augmented glue grammar rules. We try to use CCG-augmented glue grammar rules with the best-performing systems obtained in Section 6.3.1, namely the 5-category and 3-category CCG-augmented HPB SMT systems for Arabic–English and Chinese–English translation, respectively. Tables 5 and 6 show BLEU, TER and METEOR scores when using CCG-augmented glue grammar rules in these systems. Using CCG-augmented glue grammar rules for Arabic–English leads to an improvement of 0.38 absolute TER points, which corresponds to a 1% relative improvement. Using CCG-augmented glue grammar rules for Chinese–English leads to an increase of 0.79 absolute BLEU points over the 3-category CCG-augmented HPB system, which corresponds to a 1.6% relative improvement. This

result is corroborated by improvements with respect to TER and METEOR. The paired bootstrap resampling test shows that our CCG-augmented glue grammar system outperforms the 3-category CCG-augmented HPB SMT system in 93 out of 100 samples. However, this improvement is not statistically significant at p-level=0.05.

System	BLEU	TER	METEOR
CCG glue5	53.51	30.38	70.81

Table 5: Experimental results of the CCG-augmented HPB system which uses CCG-augmented glue grammar rules with 5-category extended CCG labels for Arabic–English translation.

We attempted to understand why using CCG-augmented glue grammar rules led to an improvement using 3-category extended labels, but caused a performance degradation when used with single category labels for Chinese–English translation. Accordingly, we measure the percentage of glue grammar rule application in the derivation trees that yield the translation output of each system. We found that glue grammar rules constitute 13.76% of the total rules used by the single-category CCG-augmented HPB SMT system which uses CCG-augmented glue grammar rules, compared to 4.8% used by the 3-category CCG-augmented HPB SMT system which uses CCG-augmented glue grammar rules. We think that this increased usage of glue grammar rules is due to restrictions imposed on the single-category system, which result from the restricted set of the single-category labels. This forces the system to use more glue grammar rules, which perform no reordering, causing the performance of the system to degrade. We think that the reason why using CCG-augmented glue grammar rules did not improve the performance for Arabic–English translation might be because of the small size of the training data (20k only), which increases the sparsity of translation rules extracted.

System	BLEU	TER	METEOR
CCG glue3	50.73	33.50	66.67

Table 6: Experimental results of the CCG-augmented HPB system which uses CCG-augmented glue grammar rules with 3-category extended CCG labels for Chinese–English translation.

7 Conclusion and Future Work

In this paper, we presented two syntactic extensions to HPB SMT system using CCG. The first extension tries to increase the coverage of syntactic labels used to label nonterminals in hierarchical rules by using complex CCG-based labels composed of more than one CCG category. The second extension tries to build a full parse tree which covers the whole translation output by augmenting glue grammar rules with CCG combinatory operators. We presented experiments on Arabic–English and Chinese–English translation. Our experiments showed that using extended CCG labels achieved the best performance for Arabic–English translation, while using a combination of CCG-augmented glue grammar rules and extended CCG labels led to the best performance for Chinese–English translation.

In future work, we will try to integrate the application probability of CCG combinatory operators performed during glue grammar rule application in the grammaticality feature. Furthermore, we will try to integrate more syntactically aware CCG-based evaluation metrics in tuning and evaluation, which enables a higher accuracy in evaluating improvements on the grammaticality of the translation output.

Acknowledgments

This work is supported by Science Foundation Ireland (Grant No. 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at Dublin City University.

References

- Almaghout, H., J. Jiang, and A. Way. 2010. CCG augmented hierarchical phrase-based machine translation. In *Proceedings of the 7th International Workshop on Spoken Language Translation*, pages 211–218, Paris, France.
- Almaghout, H., J. Jiang, and A. Way. 2011. CCG contextual labels in hierarchical phrase-based SMT. In *proceedings of the 15th conference of the European Association for Machine Translation*, pages 281–288, Leuven, Belgium.
- Bangalore, S. and A. Joshi. 1999. Supertagging: An approach to almost parsing. *Computational Linguistics*, 25(2):237–265.
- Birch, A., M. Osborne, and P. Koehn. 2007. CCG supertags in factored statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 9–16, Prague, Czech Republic.
- Chiang, D. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 263–270.
- Chiang, D. 2010. Learning to translate with source and target syntax. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1443–1452, Uppsala, Sweden.
- Hassan, H., K. Sima'an, and A. Way. 2009. A syntactified direct translation model with linear-time decoding. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, EMNLP '09, pages 1182–1191, Singapore.
- Koehn, P. and H. Hoang. 2007. Factored translation models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL-2007)*, pages 868–876, Prague, Czech Republic.
- Koehn, P., F.J. Och, and D. Marcu. 2003. Statistical phrase-based translation. In *HLT-NAACL 2003: conference combining Human Language Technology conference series and the North American Chapter of the Association for Computational Linguistics conference series*, pages 48–54, Edmonton, Canada.
- Koehn, P. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference of Empirical Methods in Natural Language Processing (EMNLP-2004)*, pages 388–395, Barcelona, Spain.
- Och, F.J. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, pages 160–167, Sapporo Convention Center, Japan.
- Steedman, M. 2000. *The syntactic process*. MIT Press, Cambridge, MA, USA.
- Venugopal, A., A. Zollmann, N.A. Smith, and S. Vogel. 2009. Preference grammars: softening syntactic constraints to improve statistical machine translation. In *Proceedings of Human Language Technologies: the 2009 annual conference of the North American Chapter of the ACL*, pages 37–45, Boulder, Colorado.
- Zollmann, A. and A. Venugopal. 2006. Syntax augmented machine translation via chart parsing. In *Proceedings of the Workshop on Statistical Machine Translation*, StatMT '06, pages 138–141.

MOSES CORE

Moses Open Source Evaluation and Support Co-ordination for Outreach and Exploitation

**European Union
FP7 ICT-2011-7
Coordination and Support Action
288487
<http://www.mosescore.eu>**

List of partners
University of Edinburgh (Coordinator), United Kingdom
TAUS, The Netherlands
Fondazione Bruno Kessler, Italy
Charles University in Prague, Czech Republic
Applied Language Solutions, United Kingdom

Project duration: February 2012 — January 2015

Summary

MosesCore aims to encourage the development and use of open source machine translation (MT). The project hosts three different types of events: (i) Machine Translation Marathons where MT researchers and developers can gather to discuss and implement the latest MT techniques; (ii) Workshops in Machine Translation (with shared tasks) for researchers to present their latest work, and compare techniques with other groups; and (iii) Industrial Outreach events to provide tutorials and a knowledge sharing platform for current and potential users of MT. As well as providing the events, MosesCore is funding the appointment of a full-time Moses Coordinator, based in Edinburgh. The Moses Coordinator is responsible for overseeing and facilitating the development of Moses, helping to integrate new research, and widening the developer and user base. The three academic partners in MosesCore (Edinburgh, FBK and Charles University) have extensive experience in developing and supporting open source NLP software, including the Moses toolkit. They are joined by TAUS, a think tank for the translation industry with an established record as a Moses promoter and extensive membership in the industry, and Applied Language Solutions, a Moses power user who is applying their software engineering expertise to making Moses more robust and user-friendly.



Machine Translation Enhanced Computer Assisted Translation

Funding Agency: European Union

Call: FP7-ICT-2011-7

Project Type: Strep

Project ID: 287688

Website: <http://www.matecat.com>

Consortium
Fondazione Bruno Kessler, Italy (coordinator)
Translated Srl, Italy
Université du Maine, Le Mans, France
University of Edinburgh, United Kingdom

Project duration: November 2011 — October 2014

Summary

MateCat aims to integrate statistical Machine Translation (MT) and collaborative Translation Memories (TM) within the human translation workflow. The objective is to increase the productivity of professional translators and to enhance their work experience with MT.

MateCat will go beyond the state-of-the-art by investigating new research issues related to the integration of MT into CAT, namely: self-tuning MT, user adaptive MT, and informative MT.

MateCat will develop an enhanced Web-based CAT tool integrating new MT functionalities. The project will build on state-of-the-art and widely adopted MT and CAT technologies developed by the project partners, such as Moses, IRSTLM, and MyMemory. All results of MateCat will be made publicly available under an open source licence.

Progress in MateCat will be measured by field tests evaluating the utility and usability of MT enhanced CAT. Key performance indicators will compare productivity of real users employing CAT with and without the new MT functionalities developed in the project.

At this time, a first field test was completed to evaluate a reference baseline system based on a commercial CAT tool, integrating commercial TM and MT engines. We collected and analyzed log files of 16 professional translators that worked on real translation projects in two directions, EN>IT and EN>DE, and two domains, legal and information technology. Results reported a significant increase in productivity when the TM is integrated with suggestions generated by the MT engine.










We are currently developing the first version of the MateCat Web-based CAT tool and a new statistical MT server that dynamically adapts from translations generated by the users.



SUMAT: An online service for SUBtitled by MACHine Translation

European Commission
Information and Communication Technologies Policy Support Programme
CIP-ICT-PSP.2010.6.2 - Multilingual online services
Pilot Type B
270919

<http://www.sumat-project.eu/>

List of partners	
	Vicomtech, Spain (coordinator)
	Titelbild Subtitling and Translation, Germany
	Athens Technology Center, Greece
	Univerza V Mariboru, Slovenia
	Invision Ondertiteling, The Netherlands
	Voice & Script International Limited, United Kingdom
	Deluxe Digital Studios, United Kingdom
	Applied Language Solutions, United Kingdom
	<u>Subcontracted</u> : TextShuttle, Switzerland

Project duration: April 2011 — March 2013

Summary

SUMAT aims to increase the efficiency and productivity of the European subtitling industry while enhancing the quality of its results via the effective introduction of SMT technologies into subtitling processes. In order to achieve this, we will develop an online subtitle translation service addressing nine different European languages divided into the following 14 language pairs: English-German; English-French; English-Spanish; English-Dutch; English-Swedish; English-Portuguese; Slovenian-Serbian. During the first year of the project the consortium's subtitling companies have provided large amounts of professionally produced parallel and monolingual subtitle data, which have been processed into a form suitable for training SMT systems. Baseline SMT systems are being created using the Moses SMT training scripts and decoder and the IRSTLM toolkit. In the near future, subtitles will be enriched with linguistic information and the baseline SMT systems for subtitling will be built upon by: augmenting language models with extra monolingual target data and improved use of linguistic information; enhancing translation models through the use of POS tagged data and factored models; using compound splitters, named entity recognizers and additional lexica to deal with unknown words; and investigating hierarchical decoding to make use of syntactic dependencies.



transLectures: Transcription and Translation of Video Lectures

Funding agency: European Commission

Funding call identification: FP7-ICT

Type of project: STREP

Project ID number: 287755

<http://www.translectures.eu/>

List of partners
Universitat Politècnica de València, Spain (coordinator)
Xerox Research Centre Europe, France
Jozef Stefan Institute and Knowledge for All Foundation, Slovenia and UK
RWTH Aachen University, Germany
European Media Laboratory GmbH, Germany
Deluxe Digital Studios Limited, UK

Project duration: November 2011 — October 2014

Summary

Online educational repositories of video lectures are rapidly growing. An example of this is *VideoLectures.NET*, a free and open access educational video lectures repository. Transcription and translation of video lectures in *VideoLectures.NET* is needed to make them accessible to a wider audience. However, similarly to other repositories, most lectures are neither transcribed nor translated.

The aim of **transLectures** is to develop innovative, cost-effective solutions to produce accurate transcriptions and translations in *VideoLectures.NET*, with generality across other Matterhorn-related repositories. The starting hypothesis is that there is only a relatively small gap for the current technology on automatic speech recognition and machine translation to achieve accurate enough results in the object collections we are considering; and that it can be closed by achieving the following three objectives:

1. *Improvement of transcription and translation quality by massive adaptation.*
2. *Improvement of transcription and translation quality by intelligent interaction.*
3. *Integration into Matterhorn to enable real-life evaluation.*

The main result of **transLectures** will be a set of cost-effective tools to produce accurate transcriptions and translations in *VideoLectures.NET* and other Matterhorn-related repositories. Indeed, these tools will be tested in *VideoLectures.NET* and in a smaller repository of Spanish video lectures, *poliMedia*. For transcription, we will consider English and Slovenian in *VideoLectures.NET* (more than 90%) and Spanish in *poliMedia*. For translation, we will consider the language pairs: en↔es, en↔sl, en→fr and en→de.

Upon successful achievement of the project, its techniques will probably spread over many educational repositories, enabling them to overcome language barriers and reach wider audiences.

ACCURAT: Analysis and Evaluation of Comparable Corpora for Under Resourced Areas of Machine Translation

**Seventh Framework Programme
Call FP7-ICT-2009-4, ICT-2009.2.2: Language-based interaction
Small or medium-scale focused research project (STREP)
Grant Agreement n° 248347
<http://www accurat-project.eu>**

List of partners
Tilde, Latvia (coordinator)
University of Sheffield, Computer Science Department, NLP Group, UK
University of Leeds, Centre for Translation Studies, UK
Institute for Language and Speech Processing, Greece
University of Zagreb, Faculty of Humanities and Social Sciences, Department of Linguistics, Croatia
DFKI, LT Lab, Germany
Romanian Academy, Research Institute for Artificial Intelligence, Romania
Linguattec, Germany
Zemanta, Slovenia

Project duration: January, 2010 — June, 2012

Summary

Lack of sufficient parallel data for many languages and domains is currently one of the major obstacles to further advancement of automated translation. The ACCURAT project addresses this issue by researching methods for using comparable corpora as resources for machine translation (MT). The objectives of the ACCURAT project are to develop methods allowing to measure the comparability of source and target language documents in comparable corpora; to research methods for the alignment and extraction of lexical, terminological, and other linguistic data from comparable corpora; to research methods for automatic acquisition of a comparable corpus from the Web and to analyse how acquired data can improve MT systems. The project particularly targets a number of under-resourced languages, i.e., Croatian, Estonian, Greek, Latvian, Lithuanian, and Romanian, and evaluates applicability of data extracted from comparable corpora for adapting MT to specific narrow domains.

Several novel approaches for building comparable corpora from the Web have been researched and evaluated for under-resourced languages including: (1) monolingual crawling and bilingual pairing of news texts and (2) focused monolingual crawling of narrow domain texts using seed terms and URLs.

ACCURAT has developed comparability metrics which identify similar documents in comparable corpora and indicate their degree of similarity by computing a comparability score. Tests performed on a gold standard show that scores obtained from the metrics reliably reflect comparability levels, as the average scores for higher comparability levels are always significantly larger than for lower levels.

ACCURAT also proposes new methods for extraction of parallel data from comparable corpora. These methods are implemented in an open source ACCURAT Toolkit. The toolkit identifies (maps) and extracts parallel sentences, translation dictionaries, bilingual terminology, and named entities.

Data collected and extracted using ACCURAT tools are being integrated into baseline SMT systems (trained on available parallel data) to evaluate the applicability of ACCURAT tools for improving the quality of MT. Several successful proof-of-concept experiments for narrow domains were carried out showing that even small amounts of parallel domain specific data will help improve a SMT system.



CoSyne, a Project on Multilingual Content Synchronization with Wikis

**7th Framework Programme
FP7-ICT-Call 4
small or medium-scale-focused research project
248531
<http://www.cosyne.eu>**

University of Amsterdam (UvA), The Netherlands (coordinator)
Fondazione Bruno Kessler (FBK), Italy
Dublin City University (DCU), Ireland
Heidelberg Institute for Technical Studies (HITS), Germany
Deutsche Welle (DW), Germany
Netherlands Institute for Sound & Vision (NISV), The Netherlands
WikiMedia Foundation NL, The Netherlands

Project duration: 1 March 2010 — 28 February 2013

Summary

CoSyne aims at automating the dynamic multilingual synchronization process of wikis. It deals with automating the process of analyzing and mutually enriching different wiki pages on the same subject. This includes, but is not limited to, user-generated content. The CoSyne project focuses on robust machine translation and synchronization in six designated languages. Because of the synchronization aspect, the work is done in a wiki environment, and the different services and components use the open-source MediaWiki platform.

The strength of the CoSyne system compared to its competitors is the combination of machine translation with synchronization. The system automatically recognizes which parts of text are not present in the other language version(s) and translates or augments only those parts that are considered different using segment-specific adaptive modeling.

At present the project is entering its third year of activity. In the first year, the focus was on a limited set of language combinations: German-English, Dutch-English and Italian-English (all in both directions). The second year this was expanded to include a combination among all four languages (English, German, Dutch and Italian). The third year will focus on enhanced prototyping, machine learning with identification of factual changes and style changes, and analysis of logged user edits to teach and improve the system. Two additional languages (Turkish and Bulgarian) will be used to test the system for new languages, in the final phase of the project. Contact: c.monz@uva.nl www.cosyne.eu



LT-Innovate

European Commission

Support Action

288202

<http://www.lt-innovate.eu>

List of partners
Inmark (coordinator)
European Multimedia Forum (EMF)
Language Technology Centre (LTC)
IDC Research Esapana
Europe Unlimited (EUN)

Project duration: November 2011 — February 2014

Summary

LT-Innovate is a Support Action designed to link together stakeholders in the Language Technology industry, to facilitate technology transfer and consequent market uptake of products and services resulting amongst others from RTD initiatives from EU programmes; and to contribute to the design and implementation of future Research & Innovation related programmes and initiatives in the Language Technology field.

TOSCA-MP: Task-oriented search and content annotation for media production

Funding Agency: European Union

Call: FP7-ICT-2011-7

Project Type: Strep

Project ID: 287532

Website: <http://tosca-mp.eu>

Consortium
JOANNEUM RESEARCH, Austria (coordinator)
Deutsche Thomson OHG, Germany
Fraunhofer-Gesellschaft, Germany
Union Européenne de Radio-Télévision, Switzerland
Vlaamse Radio en Televisieomroeporganisatie, Belgium
Institut fuer Rundfunktechnik GmbH, Germany
RAI – Radiotelevisione Italiana S.p.A, Italy
Fondazione Bruno Kessler, Italy
playence KG, Austria
Katholieke Universiteit Leuven, Belgium

Project duration: October 2011 — March 2014

Summary

The TOSCA-MP project aims to develop user-centric content annotation and search tools for professionals in networked media production and archiving (television, radio, online), addressing their specific use cases and workflow requirements. The project brings together 10 partners from 5 European countries including industry partners providing solutions for the media industry, public service broadcasters as well as their European association, a university and research centres. TOSCA-MP investigates scalable and distributed content processing methods performing advanced multimodal information extraction and semantic enrichment. Other key technology areas include search methods across heterogeneous networked content repositories and novel user interfaces. An open standards based service oriented framework integrates the components of the system. TOSCA-MP enables professionals in media production and archiving to seamlessly access content and indexes from distributed heterogeneous repositories in the network. This will be achieved by providing technologies that allow instant access to a large network of distributed multimedia databases, including beyond state-of-the-art metadata linking and alignment. The distributed repositories can be accessed through a single user interface that provides novel methods for result presentation, semi-automatic annotation and means of providing implicit user feedback. The networked approach of TOSCA-MP enables content holders to leverage scalable distributed processing in the network, using both in-house or external service models. The project will develop models of key user tasks in the audiovisual media production workflow. These models are used to adapt the components of the system to the specific and dynamic requirements of real user tasks in the media production domain, and to evaluate the tools in a cost-effective way.



Organic.Lingua: Demonstrating the potential of a multilingual Web portal for Sustainable Agricultural & Environmental Education.

European Commission
CIP-ICT-PSP.2010.6.2 - Multilingual Online Services
The Information and Communication Technologies Policy Support Programme
Project ID number
<http://www.organic-lingua.eu>

List of partners
The University of Alcalá, Spain (coordinator)
OU Miksike, Estonia
Know-Center GmbH, Austria
Bruno Kesler Foundation, Italy
Language & Information Technology, Italy
Xerox Research Centre Europe, France
Agro-Know Technologies, Greece
Birmingham City University, United Kingdom
University of Cukrova, Turkey
Technology and Sustainability Research Institute, Spain
French National Institute for Agricultural Research, France

Project duration: March 2011 — February 2014

Summary

The Organic.Lingua project aims to provide an automated multi-lingual service that facilitates the usage, exploitation and extension of digital educational content related to Organic Agriculture and Agroecology.

The project builds upon the existing Organ-ic.Edunet Web portal/online service that was developed in the context of the eContentplus project "Organic.Edunet: A Multilingual Federation of Learning Repositories with Quality Content for the Awareness and Education of European Youth about Organic Agriculture and Agroecology" (www.organic.edunet.eu/).

The Organic.Edunet portal currently sup-ports sixteen languages, but the current pro-cess makes translation error prone and time consuming, and misses opportunities to enhance the quality and efficiency of cross-language functions with available technology. The existing so-lution relies completely on human effort in translating and does not use existing linguistic re-sources and tools to help make resources and descriptions available in different languages or to enable cross-lingual search. In addition, resource retrieval is restricted as a consequence of the fragmentation of resource descriptions in several languages.

Organic.Lingua aims to capitalize on inter-national demand for Organic.Edunet by trans-forming it into a truly multilingual service. The main outcome of the Organic.Lingua project will be an automated multi-lingual service that will facilitate the usage, exploitation and ex-tension of digital educational con-tent related to Organic Agriculture and Agroecology.

Poster Session 4 – Research Papers

Flexible finite-state lexical selection for rule-based machine translation

Francis M. Tyers, Felipe Sánchez-Martínez, Mikel L. Forcada

Departament de Llenguatges i Sistemes Informàtics

Universitat d'Alacant

E-03071 Alacant

{ftyers, fsanchez, mlf}@dlsi.ua.es

Abstract

In this paper we describe a module (rule formalism, rule compiler and rule processor) designed to provide flexible support for lexical selection in rule-based machine translation. The motivation and implementation for the system is outlined and an efficient algorithm to compute the best coverage of lexical-selection rules over an ambiguous input sentence is described. We provide a demonstration of the module by learning rules for it on a typical training corpus and evaluating against other possible lexical-selection strategies. The inclusion of the module, along with rules learnt from the parallel corpus provides a small, but consistent and statistically-significant improvement over either using the highest-scoring translation according to a target-language model or using the most frequent aligned translation in the parallel corpus which is also found in the system's bilingual dictionaries.

1 Introduction

This paper presents a module for lexical selection to be used in rule-based machine translation (RBMT). The module consists of an XML-based formalism for specifying lexical-selection rules in the form of constraints, a compiler which converts the rules written in this format to a finite-state transducer, and a processor which applies the rule transducer to ambiguous input sentences. The paper also presents a method of learning lexical-selection rules from a parallel corpus.

Lexical selection is the task of choosing, given several source-language (SL) translations with the

same part-of-speech (POS), the most adequate translation among them in the target language (TL). The task is related to the task of word-sense disambiguation (Ide and Véronis, 1998). The difference is that its aim is to find the most adequate translation, not the most adequate sense. Thus, it is not necessary to choose between a series of fine-grained senses if all these senses result in the same final translation.

The dominant approach to MT for language pairs with sufficient training data is phrase-based statistical machine translation; in this approach, lexical selection is performed by a combination of cooccurrence in the phrase table, and score from the target-language model (Koehn, 2010). There have however been attempts to improve on this by looking at global lexical selection over the whole sentence, see e.g. (Venkatapathy and Bangalore, 2007; Carpuat and Wu, 2007).

In order to test different approaches to lexical selection for RBMT, we use the Apertium (Forcada et al., 2011) platform. This free/open-source platform includes 30 language pairs (as of February 2012).

Sánchez-Martínez et al. (2007) describe a method to perform lexical selection in Apertium based on training a source-language bag-of-words model using TL cooccurrence statistics. This approach was tested, but abandoned as it produced less adequate translations than using the translation marked as default by a linguist in the bilingual dictionary.

Other possible solutions would be to generate all possible combinations of translations, and score them on a language model of the target language. This approach is taken in the METIS-II system (Melero et al., 2007). This has the benefit of being easy to implement, and only requiring a bilingual dictionary and a monolingual target language corpus. It has the drawbacks of being both slow – many

translations must be performed – and not very customisable – control over the final translation is left to the TL model.

Another possible solution, and one that is already used in some Apertium language pairs (Brandt et al., 2011; Wiecheteck et al., 2010) is to use constraint grammar (Karlsson et al., 1995) rules to choose between possible alternative translations. An advantage of this is that the constraint grammar formalism is well known, and powerful, allowing context searches of unlimited size. However, it is too slow to be able to be used for production systems, as the speed is in the order of a few hundred words per second as opposed to thousands of words per second for the slowest Apertium module.

Another approach not requiring a parallel corpus is presented by Dagan and Itai (1994). They first parse the SL sentence and extract syntactic relations, such as verb + object, they then translate these with a bilingual dictionary and use collocation statistics from a TL corpus to choose the most adequate translation. While this method does not rely on the existence of a parallel corpus, it does depend on some way of identifying SL syntactic relations – which may not be available in all RBMT systems.

The rest of the paper is laid out as follows: Section 2 presents some design decisions that were made in the development of the module. Section 3 describes in detail the rule formalism, the representation of rules as a finite-state transducer, and the algorithm for applying the rules to an ambiguous input sentence. Section 4 shows how rules for the module may be learnt from a parallel corpus, and then evaluated on a standard test set for MT. Finally, section 6 offers some concluding remarks and ideas for future work.

2 Lexical selection in Apertium

Apertium is an free/open-source platform for creating shallow-transfer RBMT systems. The platform is being widely used to build MT systems for a variety of language pairs, especially in those cases (mainly with related-language pairs) where shallow transfer suffices to produce good quality translations. It has, however, also proven useful in assimilation scenarios with more distant pairs involved.

The platform is designed to be: *fast*, in the order of thousands of words per second on a normal desktop computer; *easy* to develop; and *standalone*, no need for existing data or large parallel corpora to build a system.

Apertium uses a Unix pipeline architecture (see Figure 1) to perform translation: text is first stripped of format and morphologically analysed, then morphologically disambiguated. Then the unambiguous analyses are passed through lexical and structural transfer and finally morphological generation. This translation strategy is very similar to other transfer-based MT systems.

The Apertium platform does not currently have a specific module for lexical selection. Some translation ambiguity can be handled using multi-word expressions (MWEs) encoded in the dictionaries of the system, but the status quo is that for any given SL word, the most frequent, or most general translation is given. This poses a translation problem, as often it may be difficult to choose the most frequent or the single most adequate translation of a word, or the selection strongly depends on the context.

2.1 Requirements

The requirements of a lexical selection module are:

- It should be efficient and fast, that is, it should process thousands of words per second on a normal desktop computer. For rule sets of tens of thousands of rules.
- It should not require any advanced resources, such as parallel corpora, but should be able to take advantage of them if available.
- The functioning of the module should be traceable. In any given translation, it should be possible to identify the rules used.
- The rules should be in a form suitable for reading and writing by human beings so that users can immediately change or add rules.

In the next section we describe a lexical selection module which fulfils these requirements.

In order to accommodate the new lexical selection module, a minor change was made to the pipeline (Figure 1). Where previously lexical transfer was performed at the same time as structural transfer, now lexical transfer is performed as a separate process before the structural transfer stage.

3 Methodology

3.1 Rule formalism

The rule formalism is based on context rules, containing a sequence of the following features,

- A pattern matching a single SL lexical form

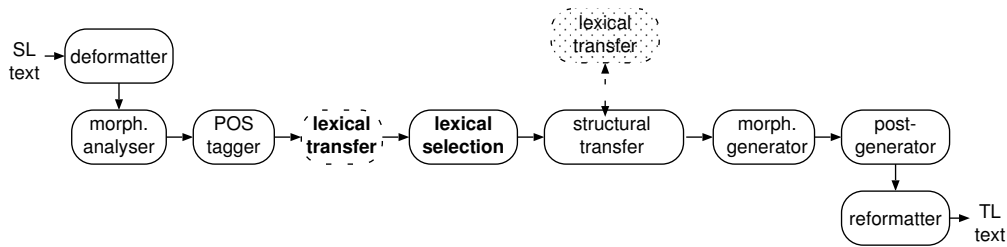


Figure 1: The Apertium architecture. The lexical transfer module (shaded) has been moved from being called from the structural transfer module to being a module in its own right (in bold face) and the lexical selection module has been inserted between lexical transfer and structural transfer.

- A pattern matching a single TL lexical form
- One of the following operations:
 - select** chooses the TL translation which matches the lexical-form pattern and removes all translations which do not match.
 - remove** removes the TL translation which match the given lexical-form pattern; and
 - skip** makes no changes and passes all the translations through unchanged; this is used when specifying the context of the rule.

The features are expressed by regular expressions, which may match any part of the input word string (e.g. either the lemma, the tags or a combination of both). As with the rest of the modules in the Apertium platform, the rules are written in an XML-based format, which is processable by both humans and machines.

Figure 2 presents some examples of rules written in this formalism. Each rule is enclosed in a `rule` element, with an optional `c` attribute for comments. The `rule` tag may have one or more `match` elements which describe sequences of SL context. Each `match` element may have either a `lemma` or a `tags` attribute, neither (in which case it will match any word) or both.

A `match` element may also contain a lexical selection operation, `select` or `remove`, the default one being `skip`.

The rules can be written by hand to solve specific translation issues with a given context, for example, given the Spanish word *estación* ‘station, season’ with a default translation of ‘station’, we may write rules (see Figure 2) which say that we want to translate the word as ‘season’ if it is followed by an adjective such as *seca* ‘dry’ or *lluviosa* ‘rainy’, or if it is followed by the preposition *de* ‘of’, a determiner (e.g. *el* ‘the’), and the noun *año* ‘year’.

A weak point of the formalism is that rules can only take into account fixed-length, ordered contexts, so it is not possible to e.g. make a rule which selects a given translation based on a given word at any position in the sentence (e.g. treating the sentence, or part of it, as a bag of words). However, a strength is that the rules may be compiled into a compact finite-state transducer, which is traceable; for each translation, it is possible to know exactly which rules were called.

3.2 Rule compilation

The set of rules R expressed in XML is not processed directly; they are compiled into a finite-state transducer (see Figure 3). In this transducer, each transition is labelled with a symbol representing an SL pattern and a symbol representing an operation on a TL pattern. Both SL and TL patterns are compiled into regular expressions (finite-state recognisers), and stored in a lookup table.

The transducer is defined as $\langle Q, V, \delta, q_0, q_F \rangle$, where Q is the set of states, $V = \Sigma \times \Gamma$ is the alphabet of transition labels, where Σ is the set of input symbols and Γ is the set of output symbols, $\delta : Q \times V \rightarrow Q$ is the transition function, q_0 is the initial state (nothing matched); and q_F is the final state indicating that a complete pattern has been matched. Rules in R are paths from q_0 to q_F .

3.3 Rule application

In order to apply the rules on an input sentence, we use a variant of the best coverage algorithm described by Sánchez-Martínez et al. (2009). We try to cover the maximum number of words of each SL sentence by using the longest possible rules; the motivation for this is that the longer the rules, the more accurate their decisions may be expected to be because they integrate more context.

To compute the best coverage a dynamic-programming algorithm (Alg. 1) is applied, which starts a new search in the automaton at every new word in the sentence to be translated, and uses a

```

<rule c="default translation">
  <match lemma="estación"><select lemma="station"/></match>
</rule>
<rule>
  <match lemma="estación"><select lemma="season"/></match>
  <or>
    <match lemma="seco">
      <match lemma="lluvioso">
    </or>
  </rule>
<rule>
  <match lemma="estación"><select lemma="season"/></match>
  <match lemma="de">
    <match tags="det.*"/>
    <match lemma="año">
  </rule>
...

```

Figure 2: An example of the rules written by hand in the XML formalism for describing lexical selection rules. The formalism is the same for both hand-written and learnt rules. The order of rules is only important in calculating the rule number for tracing.

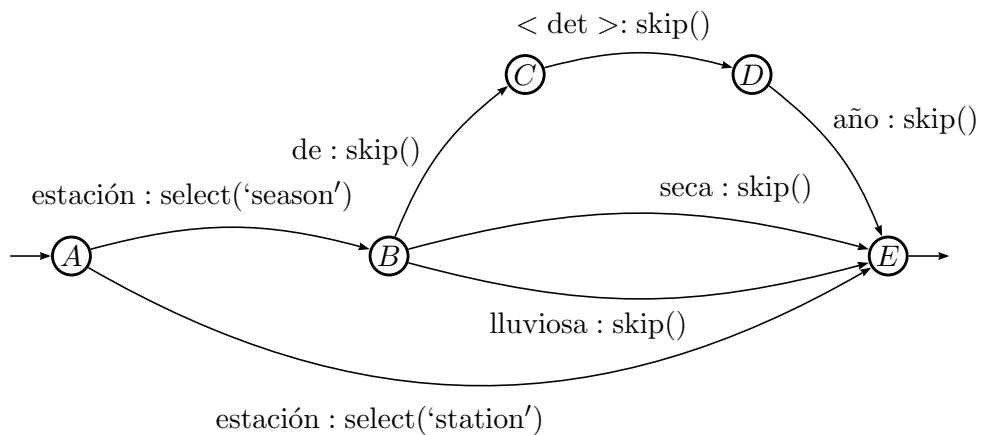


Figure 3: A finite-state transducer representing four lexical selection rules; each arc is a transition between a pattern matching an SL lexical form, and an operation with a pattern matching a TL lexical form.

set of alive states A in the automaton and a map M that, for each word in the sentence, returns the best coverage up to that word together with its score.

Algorithm 1 uses four external procedures: `WORDCOUNT(s)` returns the number of words in the string s ; `RULELENGTH(c)` returns the number of words of the rule matched by state c ; `NEWCOVERAGE(cov, c)` computes a new coverage by adding to coverage cov the rule recognised by state c ; finally, `BESTCOVERAGE(a, b)` receives two coverages and returns the one using the least possible number of rules.

In the current implementation, if two different coverages use the same number of rules, then the former is overwritten. This may not be the most adequate approach to dealing with the problem, and we intend to study other approaches.

4 Experiment

In order to test the flexibility of the module, we decided to learn rules from an existing knowledge source, i.e. a parallel corpus, and test the module on a well-known task for the evaluation of MT.

The experimental setup follows the training of the baseline system in the shared task on MT at WMT11 (Callison-Burch et al., 2011), with the following differences: In place of the default Moses perl-based tokeniser, tokenisation was done using the Apertium morphological analyser (Cortés-Vaíllo and Ortiz-Rojas, 2011). The corpus was also not lowercased; instead the case of known words was changed to the dictionary case as found in the Apertium monolingual dictionary.

We use version 6.0 of the EuroParl corpus (Koehn, 2005), and take the first 1.4 million lines for training.¹ We used the Apertium English to Spanish pair `apertium-en-es`² as it is one of the few pairs that has dictionaries with more than one alternative translation per word.³

4.1 Learning lexical selection rules from a parallel corpus

The procedure to learn rules from a parallel corpus is as follows: We first morphologically analyse and disambiguate for part-of-speech both the SL and TL sides of the corpus. These are then word-aligned with GIZA++ (Och and Ney, 2003).

¹The remaining lines were held out for future use.

²Available from <http://wiki.apertium.org/wiki/SVN>; SVN revision: 35684

³The lexical selection module is available as free/open-source software in the package `apertium-lex-tools`. This paper uses SVN revision: 35799

We then pass the SL side of the corpus through the lexical-transfer stage of the MT system we are learning the rules for; this gives three sets of sentences: the tagged SL sentences, the tagged TL sentences and the possible translations of the SL words into the TL yielded by the bilingual dictionary.

We take these three sets, and extract from the parallel corpus those sentence pairs for which at least one lexically ambiguous SL word is aligned to a word in the TL which is also found in the bilingual dictionary. This step is necessary as in order to be translated by the rest of the system, the alternative translation must appear in the bilingual dictionary. After extracting these sentence pairs we have 332,525 sentences for training, that is around 24% of them.

For each of these extracted sentences, we extract n -grams (trigrams and five-grams) of context around the ambiguous SL word(s) which belong to the categories of adjective, noun and verb. We then count up how many times we see this context appearing along with each of the translations in the TL. If a given possible translation appears aligned to a word in a given context more frequently than other possible translations, then we generate a rule which selects the aligned translation in that same context over other translations in that context.

4.2 Systems

To evaluate the lexical selection module, and our method for obtaining rules from a parallel corpus, we compare it against four baseline systems:

- **freq**: Frequency defaults; the MT system is tested with rules that select the most frequent translation in the TL corpus. This is equivalent to a unigram TL model.
- **alig**: The TL word which is most frequently aligned to the given SL word is chosen. This correspondence must also appear in the bilingual dictionary of the MT system.
- **ling**: The linguistic defaults, here the translations considered ‘most adequate’ by the human linguist who wrote the system, are selected.
- **tlm**: The highest scoring translation out of the possible translations for the whole sentence as chosen by a 5-gram language model of the Spanish side of the EuroParl corpus trained with IRSTLM (Federico et al., 2008).

Algorithm 1 OPTIMALCOVERAGE: Algorithm to compute the best coverage of an input sentence.

Require: s : SL sentence to translate $A \leftarrow \{q_0\}$ $i \leftarrow 1$ **while** ($i \leq \text{WORDCOUNT}(s)$) **do** $M[i] \leftarrow \emptyset$ **for all** $q \in A$ **do****for all** $c \in Q \exists t : \delta(q, (s[i] : t) = c)$ **do** $A \leftarrow A \cup \{c\}$ **if** $c = q_F$ **then** $M[i] \leftarrow \text{BESTCOVERAGE}(M[i], \text{NEWCOVERAGE}(M[i - \text{RULELENGTH}(c)], c))$ **end if****end for** $A \leftarrow A - \{q\}$ **end for** $i \leftarrow i + 1$ $A \leftarrow A \cup \{q_0\}$ /* To start a new search from the next word */**end while****return** $M[i - 1]$

We also tested three different sets of rules in our lexical-selection module:

- **all**: No filtering. All of the generated rules are included.
- **filt1**: The rules where contexts which only appear once in the training corpus are removed.
- **filt2**: Rules which include the tags for subordinating conjunction and full stop are excluded as well as rules where the translation selected is under half of the total frequency of the word. So for example if a word has three translations with frequency 10 and one translation with frequency 15, the rule selecting this translation would be excluded as $15 < (45 / 2)$ even though it is the most frequent.

The motivation for excluding rules which contain subordinating conjunctions and full stops is that they are likely to be noisy. The motivation for excluding rules with under half of the total frequency of the word is to try and keep only those rules that we are really sure will improve translation quality overall. These are rather coarse heuristics, and the subject of rule filtering merits further investigation (see section 6).

5 Evaluation

To evaluate the systems, we extracted the set of sentences from the 2,489-sentence News Commentary corpus which contained at least one ambiguous open-category word in the SL aligned with a TL

word in the reference translation which could be generated by the MT system. The alignments between SL and TL words in the corpus were obtained by adding it to a separate copy of the EuroParl corpus to the one used for training, and running GIZA++ again.

In total, this gave 434 sentences (9,463 tokens) to be evaluated (approximately 17%). The average number of translations per word was 1.08.⁴ We performed two evaluation tasks, the first was the error rate of the lexical selection module, and the second was a full translation task.

For the first, we made a labelled corpus (similar to that in (Vickrey et al., 2005)) by disambiguating the lexical transfer output using the reference translation. Out of the 434 sentences this gave us a total of 604 disambiguated words. This could be considered an *oracle*, that is the best result the MT system could get if it just chose the translation looking at the reference translation. The column **Error** in Table 2 gives the lexical-selection error rate over this test corpus, that is the number of times the given system chooses a translation which is not equivalent to what the oracle would choose.

The second task was to compare the systems using the common evaluation metrics BLEU (Papineni et al., 2002) and Word error rate (WER), based on the Levenshtein distance (Levenshtein, 1965).

This second task is not ideal for evaluating the task of a lexical selection module as the perfor-

⁴This number is low and indicates that there is work to be done on expanding the dictionaries of the system for lexical choice.

src:	If it doesn't reduce social benefits ...
ref:	Si no reduce los <i>subsídios</i> sociales ...
alig:	Si no reduce <i>beneficios</i> sociales ...
filt2:	Si no reduce <i>prestaciones</i> sociales ...

Table 1: Translation of segment #56 in the News Commentary corpus by two of the systems.

mance of the module will depend greatly on (a) the coverage of the bilingual dictionaries of the RBMT system in question, and (b) the number of reference translations. It is included only as it is a common metric used to evaluate MT systems.

In addition, when there is only one reference translation (such as in the News Commentary corpus), the system may easily generate a more adequate translation of a word, which is then not found in the reference. For example, in Table 1, *prestaciones* ‘benefits, provision, assistance’ is a more adequate translation for ‘benefits’ than *beneficios* ‘profit, advantage, benefits’, but as it does not appear in the reference, this translation improvement is not counted. However, without annotating a corpus manually with all possible translation possibilities, or using several reference translations it is difficult to see how this problem may be overcome.

Table 2 reports the 95% confidence interval for the BLEU, WER and ERROR scores achieved on the test set by the seven systems. Confidence intervals were calculated through the bootstrap resampling (Efron and Tibshirani, 1994) method as described by (Koehn, 2004; Zhang and Vogel, 2004). Bootstrap resampling was carried out for 1,000 iterations.

Given the small differences in score between the individual systems, we also performed pair bootstrap resampling between the two highest scoring systems (**alig** and **filt2**) to see if the difference was statistically significant. Over 1,000 iterations, the **filt2** system was shown to offer an improved translation 95% of the time for both the BLEU and ERROR scores.

6 Concluding remarks

We have presented a lexical-selection module suitable for inclusion in a RBMT system, and shown how the rules it uses may be learnt from a parallel corpus. In pair bootstrap resampling, the system offers a statistically significant improvement in translation quality over the next highest scoring system.

In the future we would like to investigate the following: The first is the possibility of learning the rules without any parallel corpus. We aim to

follow the same principles as (Sánchez-Martínez et al., 2008) where a monolingual TL corpus was used to improve the performance of an HMM part-of-speech tagger. Some initial experiments have already been conducted to this effect, however the observed performance of the TL model in choosing between different translations from an RBMT system gives an indication of the difficulty of improving over the ‘linguistic default’ baseline.

While the learning from parallel corpora is only a demonstration, we would like to look into methods to address the problem of filtering/pruning the generated rules to remove those which do not offer an improvement in translation quality, as it would also apply to learning rules without parallel corpora.

The system has also been built with the possibility of weighted rules, we would like to investigate the possibility of automatically assigning rule weights to more reliable rules.

Acknowledgements

We are thankful for the support of the Spanish Ministry of Science and Innovation through project TIN2009-14009-C02-01, and the Universitat d’Alacant through project GRE11-20. We also thank Sergio Ortiz Rojas for his constructive comments and ideas on the development of the system, and the anonymous reviewers for comments on the manuscript.

References

- Brandt, M. D., H. Loftsson, H. Sigurþórsson, and F. M. Tyers. 2011. Apertium-icenlp: A rule-based icelandic to english machine translation system. In *Proceedings of the 16th Annual Conference of the European Association of Machine Translation*, pages 217–224.
- Callison-Burch, C., P. Koehn, C. Monz, and O. F. Zaidan, editors. 2011. *Proceedings of the Sixth Workshop on Statistical Machine Translation*. Association for Computational Linguistics.
- Carpuat, M. and D. Wu. 2007. How phrase sense disambiguation outperforms word sense disambiguation for statistical machine translation. In *Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 43–52.
- Cortés-Vaíllo, S. and S. Ortiz-Rojas. 2011. Using apertium linguistic data for tokenization to improve mooses smt performance. In *Proceedings of the International Workshop on Using Linguistic Information for Hybrid Machine Translation, LIHMT-2011*, pages 29–35.

System	Total rules	Called	Error	BLEU	WER
freq	-	-	[42.8, 50.3]	[0.1687, 0.1794]	[0.712, 0.725]
ling	667	473	[25.4, 30.7]	[0.1772, 0.1879]	[0.710, 0.723]
alig	600	533	[19.3, 25.8]	[0.1786, 0.1892]	[0.709, 0.723]
t1m	-	-	[37.0, 44.9]	[0.1708, 0.1817]	[0.714, 0.727]
all	77,077	503	[21.3, 28.2]	[0.1779, 0.1885]	[0.710, 0.723]
filt1	9,978	503	[20.3, 26.9]	[0.1782, 0.1889]	[0.710, 0.723]
filt2	2,661	532	[17.9, 24.7]	[0.1789, 0.1896]	[0.709, 0.723]

Table 2: Evaluation results for the seven systems on the news commentary test corpus.

- Dagan, I. and A. Itai. 1994. Word sense disambiguation using a second language monolingual corpus. *Computational Linguistics*, 40(4):563–596.
- Efron, B. and R. J. Tibshirani. 1994. *An introduction to the Bootstrap*. CRC Press.
- Federico, M., N. Bertoldi, and M. Cettolo. 2008. Irlstm: an open source toolkit for handling large scale language models. In *Proceedings of Interspeech, Brisbane, Australia*.
- Forcada, M. L., M. Ginestí-Rosell, J. Nordfalk, J. O’Regan, S. Ortiz-Rojas, J. A. Pérez-Ortiz, F. Sánchez-Martínez, G. Ramírez-Sánchez, and F. M. Tyers. 2011. Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144.
- Ide, N. and J. Véronis. 1998. Word sense disambiguation: The state of the art. *Computational Linguistics*, 24(1):1–41.
- Karllsson, F., A. Voutilainen, J. Heikkilä, and A. Anttila. 1995. *Constraint Grammar: A language independent system for parsing unrestricted text*. Mouton de Gruyter.
- Koehn, P. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 388–395.
- Koehn, P. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the 10th MT Summit*, pages 79–86.
- Koehn, P. 2010. *Statistical Machine Translation*. Cambridge University Press, United Kingdom.
- Levenshtein, V. I. 1965. Binary codes capable of correcting deletions, insertions, and reversals. *Doklady Akademii Nauk SSSR*, 163(4):845–848. English translation in Soviet Physics Doklady, 10(8), 707–710.
- Melero, M., A. Oliver, T. Badia, and T. Suñol. 2007. Dealing with bilingual divergences in mt using target language n -gram models. In *Proceedings of the METIS-II Workshop: New Approaches to Machine Translation, CLIN17*, pages 19–26.
- Och, F. J. and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.
- Papineni, K., S. Roukos, T. Ward, and W-J. Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Assoc. Comp. Ling.*, pages 311–318.
- Sánchez-Martínez, F., J. A. Pérez-Ortiz, and M. L. Forcada. 2007. Integrating corpus-based and rule-based approaches in an open-source machine translation system. In *Proceedings of the METIS-II Workshop: New Approaches to Machine Translation, CLIN17*, pages 73–82.
- Sánchez-Martínez, F., J. A. Pérez-Ortiz, and M. L. Forcada. 2008. Using target-language information to train part-of-speech taggers for machine translation. *Machine Translation*, 22(1-2):29–66.
- Sánchez-Martínez, F., M. L. Forcada, and A. Way. 2009. Hybrid rule-based – example-based MT: Feeding apertium with sub-sentential translation units. In *Proceedings of the 3rd Workshop on EBMT*, pages 11–18.
- Venkatapathy, S. and S. Bangalore. 2007. Three models for discriminative machine translation using global lexical selection and sentence reconstruction. In *Proceedings of SSST, NAACL-HLT/AMTA Workshop on Syntax and Structure in Statistical Translation*, pages 152–159.
- Vickrey, D., L. Biewald, M. Teyssier, and D. Koller. 2005. Word-sense disambiguation for machine translation. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 771–778.
- Wiecheteck, L., F. M. Tyers, and T. Omma. 2010. Shooting at flies in the dark: Rule-based lexical selection for a minority language pair. *LNAI*, 6233:418–429.
- Zhang, Y. and S. Vogel. 2004. Measuring confidence intervals for the machine translation evaluation metrics. In *Proceedings of The 10th International Conference on Theoretical and Methodological Issues in Machine Translation*.

Statistical Post-Editing of Machine Translation for Domain Adaptation

Raphaël Rubino

Stéphane Huet

Fabrice Lefèvre

Georges Linarès

LIA-CERI

Université d'Avignon et des Pays de Vaucluse

Avignon, France

{firstname.lastname}@univ-avignon.fr

Abstract

This paper presents a statistical approach to adapt out-of-domain machine translation systems to the medical domain through an unsupervised post-editing step. A statistical post-editing model is built on statistical machine translation (SMT) outputs aligned with their translation references. Evaluations carried out to translate medical texts from French to English show that an out-of-domain machine translation system can be adapted *a posteriori* to a specific domain. Two SMT systems are studied: a state-of-the-art phrase-based implementation and an online publicly available system. Our experiments also indicate that selecting sentences for post-editing leads to significant improvements of translation quality and that more gains are still possible with respect to an oracle measure.

1 Introduction

Phrase-Based Machine Translation (PBMT) is a popular approach to Statistical Machine Translation (SMT) that leads to accurate translation results (Zens et al., 2002; Marcu and Wong, 2002; Koehn et al., 2003). The statistical models used in PBMT are based on the probabilities of bidirectional alignment of phrases between two sentences in the translation relation. The linguistic resources used to estimate such probabilities are parallel corpora and the main resulting statistical model is a translation table. Therefore, parallel corpora are the cornerstone for high quality translation. However, such resources are expensive to construct.

This lack of parallel data still remains an issue in PBMT. This phenomenon is accentuated by the diversity of texts to translate, in terms of origin and domain. As explained in (Sager et al., 1980), most of human activities involve a specific language or a *subject language*. A specific domain can be characterized by particular terminology or syntactic and discourse structures. As building domain specific translation systems for each domain is unreasonable, we assume that domain adaptation of out-of-domain translation systems can be one of the solutions to address the diversity of specific domains.

Although current machine translation systems can lead to impressive accuracy, translated texts require sometimes human post-processing to be usable. However, editing *a posteriori* can be costly depending on the amount of corrections required by machine translation outputs. Therefore, the automation of post-editing is an important task which can lead to higher quality machine translation without requiring human intervention.

In this paper, we propose a statistical post-editing (SPE) approach to adapt SMT systems to specific domains. We focus on translating texts in the medical domain from French to English. Several SMT systems are studied and we propose different methods to include the in-domain data into the translation process. We evaluate how translation quality can be improved with a post-editing step based on a phrase-based alignment approach. Two sets of experiments are presented in this paper: one applying SPE consistently on all the sentences and one resorting to SPE only on selected sentences.

The remainder of this paper is organized as follows. Section 2 presents the phrase-based post-editing approach. In Section 3, we propose an ex-

perimental setup and give details about the data, the language models, the translation and the post-editing systems used in our experiments. Section 4 evaluates each SMT system on a domain specific translation task, then Section 5 analyses the effect of a standard post-editing system on translated texts. Section 6 presents our approach to select sentences for post-editing. Finally, Section 7 concludes this paper.

2 Phrase-Based Statistical Post-Editing

2.1 SPE Principles

The post-editing of a machine translation output consists of the generation of a text T'' from a translation hypothesis T' of a source text S . When a PBMT system is built on bilingual parallel data, a phrase-based SPE system requires monolingual parallel texts. Recent approaches on SPE are based on three-part parallel corpora composed of a source language text, its translation by an MT system and this output manually post-edited (Knight and Chander, 1994; Allen and Hogan, 2000). If SPE can correct mistakes made by machine translation systems, it can also be used to adapt machine translation outputs to specific domains.

2.2 SPE for Adaptation

The research presented in this paper addresses the issue of adapting an out-of-domain machine translation system using a small in-domain bilingual parallel corpus. We study various uses of out and in-domain data to build Language Models (LMs) and Translation Models (TMs) inside the source-to-target language PBMT. Then, we evaluate the post-editing model using out and in-domain data to build LMs and in-domain data only for the SPE model. We also describe a new method to select sentences using classifiers built with the BLEU criterion (Papineni et al., 2002).

Figure 1 illustrates the general architecture of our experimental setup, described in the next section. The source language part of the in-domain parallel corpus is first translated into the target language by an SMT system. Then, the generated translation hypotheses are aligned with their translation references in order to form a monolingual parallel corpus and to build a SPE model. When a test corpus is translated and has to be post-edited, we propose two different approaches. The first one is a *naive* application of SPE which post-edits all the sentences of the test corpus. The second one

is based on a classification approach that aims to avoid a degradation of translation quality at the sentence level. For this last approach, we build a sentence classification model to predict whether or not the sentences from the test set can be improved with SPE.

2.3 Related Work

In (Simard et al., 2007a), the authors propose to post-edit translations from a Rule-Based Machine Translation (RBMT) system using the PBMT system PORTAGE (Sadat et al., 2005). A qualitative study of phrase-based SPE is presented by (Dugast et al., 2007; Dugast et al., 2009), where the Systran system outputs are post-edited with PORTAGE and MOSES. The authors report gains up to 10% absolute of BLEU.

In (Isabelle et al., 2007; Simard et al., 2007b), it is shown that a generic, or out-of-domain, RBMT system can be adapted to a specific domain through phrase-based SPE. Domain specific data are introduced at the post-editing level, which globally improves the translation quality. Besides, de Ilaraza et al. (2008) propose the same architecture, phrase-based SPE following a RBMT system, and introduce a small amount of in-domain data to train the SPE model, as well as morphological information in both systems.

More recently, Béchara et al. (2011) design a full PBMT pipeline that includes a translation step and a post-editing step. The authors report a significant improvement of 2 BLEU points for a French to English translation task, using a novel context-aware approach. This method takes into account the source sentences during the post-editing process through a word-to-word alignment between the source words and the target words generated by the translation system. This latter work is, to the best of our knowledge, the first attempt to combine two PBMT systems, one for translating from the source to the target language, and another one for post-editing the first system output.

This kind of PBMT pipeline had already been suggested by previous authors (Isabelle et al., 2007; Oflazer and El-Kahlout, 2007). Let us note that their work is not targeting to improve the outputs of an out-of-domain SMT system with adaptation data as in our approach. Another recent approach related to our work was presented in (Suzuki, 2011) to select sentences for post-editing. The authors present an architecture

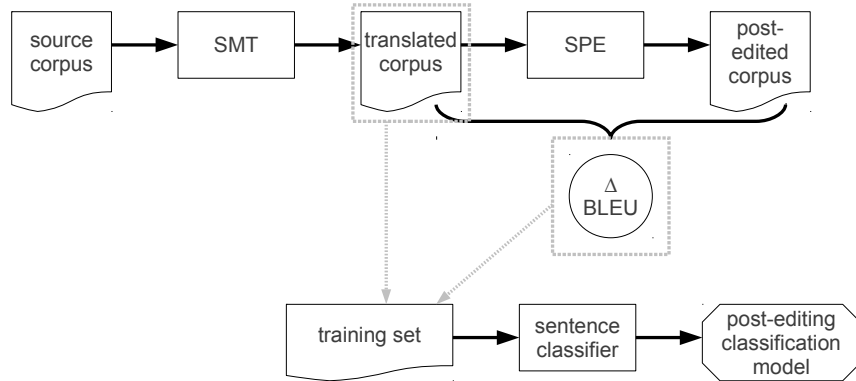


Figure 1: Training of a SVM classifier using a translated corpus where each sentence is associated with its $\Delta BLEU$ class.

composed of a phrase-based SPE system and a sentence-level automatic quality estimator based on Partial Least Squares.

3 Experimental Setup

In brief, the general idea of the work presented in this paper is to increase the quality of in-domain translations, generated by an out-of-domain SMT system, through a post-editing step. In order to thoroughly evaluate our approach, two SMT systems are considered to translate from the source language to the target language: the MOSES PBMT implementation (Koehn et al., 2007) and the GOOGLE TRANSLATE online system¹. The post-editing step is then performed using MOSES in both cases. The latter case (the online system) will help to justify our approach showing that a powerful yet fixed MT system can be profitably combined with a system trained on a small set of in-domain data. The approach is evaluated at two levels: first, we evaluate the accuracy of each translation system on a domain specific translation task. Second, we focus on the use of SPE systems to process each translation system output.

Section 3.1 introduces the out and in-domain data used in our experiments. These data can be combined in different ways inside LMs and TMs; the resulting translation systems are described in Section 3.2. Then, Section 3.3 provides information about our SPE models.

3.1 Resources

Out-of-domain data are presented in Table 1. The bilingual parallel corpora are the sixth version of the Europarl corpus (Koehn, 2005) and the United Nations corpus (Rafalovitch and Dale, 2009). The

¹<http://translate.google.com/>

monolingual corpora are composed of the target language part of the sixth version of the News Commentary corpus taken from the *Project Syndicate* website², and the Shuffled News Crawl corpus. All these corpora were made available for the 2011 Workshop on Machine Translation (WMT11)³. The bilingual data are used to build translation models, whereas the monolingual data are employed to train language models.

Corpus	Sentences	Words
<i>Bilingual Training Data</i>		
Europarl v6	1.8 M	50 M
United Nations	12 M	300 M
EMEA (Medical)	160k	4 M
<i>Monolingual Training Data</i>		
News Commentary v6	181 k	4 M
Shuffled News from 2007 to 2011	25 M	515 M

Table 1: Number of sentences and words for the out and the in-domain data used in our experiments.

The in-domain domain data used in our experiments are taken from the EMEA corpus (Tiedemann, 2009), made out of PDF documents from the European Medicines Agency⁴. The source documents are associated with three biomedical categories: general medical documents and public evaluation reports about human or veterinary treatments. This corpus is particularly interesting because it contains medical terminology and specific linguistic structures. Since the EMEA corpus contains lots of repeated expressions (on med-

²<http://www.project-syndicate.org/>

³<http://www.statmt.org/wmt11/>

⁴<http://www.emea.europa.eu/>

ical prescriptions for instance), we removed duplicates. Furthermore, short sentences of one word and long sentences exceeding 80 words were discarded. The resulting corpus is split separately for each category into three parts, which globally leads to three corpora: a 156k-sentence training set, a 2k-sentence development set and a 2k-sentence test set.

3.2 Initial SMT Systems

The online translation tool, noted *com* in the remainder of this paper, cannot be modified. It provides us with translation hypotheses which can be scored and post-edited in order to evaluate our approach. The MOSES PBMT implementation can be used to train a translation model from parallel corpora. Several PBMT systems are built, based on the bilingual and monolingual data used.

Three different 5-gram Kneser-Ney LMs are trained on the resources, using the SRILM toolkit (Stolcke, 2002). A first one (LM_g) is built on the monolingual out-of-domain data while a second one (LM_m) is built on the target language part of the medical (in-domain) corpus. These two models are combined through a linear interpolation (LM_{g+m}). For this last LM, weights were computed from the perplexity optimization on the EMEA development corpus, and vocabulary was fixed to 1 million words taking all the words of the in-domain corpus and the most frequent words from the out-of-domain corpora. Let us note that a high weight of 0.9 is associated with the medical LM despite its small size, which is explained by the great specificity of the medical domain.

Three Translation Models (TMs) incorporating a phrase table and a lexicalized reordering model are also built using MOSES: one (TM_g) from the out-of-domain data, one (TM_m) from the medical set and a last one (TM_{g+m}) from all the parallel corpora. For that purpose, bilingual data are aligned at the word level using the IBM 4 model (Och and Ney, 2003) with MGIZA++ (Gao and Vogel, 2008). The score weights of a given TM and a selected LM are finally computed in each tested configuration using the Minimum Error Rate Training (MERT) method (Och, 2003) to optimize BLEU on the EMEA development corpus. To mix the information from the out and in-domain in TM_{g+m} , we resorted to the multiple translation tables option implemented into MOSES. With this feature, we can provide two translation tables to

the decoder; the decoder first retrieves translation pairs from the in-domain phrase table, and resorts to the out-of-domain phrase-table as a fall-back.

3.3 SPE Systems

In order to build the SPE system for domain adaptation, we decide to translate the EMEA training corpus with each tested SMT system. Then, with the output of each system aligned with its translation reference, we build an SPE model using MOSES with default parameters. For the tuning process, we used the same in-domain development data as the SMT systems, this time with the SMT output aligned with its translation reference. Let us note that the weight optimization was repeated for each tested PBMT configuration.

4 Translating In-Domain Data

The first set of experiments deals with the translation of the domain specific, or in-domain, test corpus. The results are given in terms of BLEU scores in Table 2 with several uses of the previously described TMs and LMs. Pair-wise comparisons between systems is made using approximate randomization as implemented in the evaluation tool FASTMTEVAL (Stroppa et al., 2007). These results indicate that the best configuration is $TM_{g+m}LM_{g+m}$, with a BLEU score of 47.3%. This score is not significantly higher (p -value=0.75) than the one obtained by $TM_{g+m}LM_m$ with an in-domain language model. These observations show that the specificity of the medical domain, including terminology and syntactic structures, cannot be improved by the introduction of out-of-domain data into the LM. For the translation model, however, the combination of the two phrase tables is the best configuration in the presented system comparison.

SMT system	% BLEU	p -value
TM_g LM_g	29.9	0.002
TM_g LM_{g+m}	38.2	0.002
TM_g LM_m	39.2	0.002
<i>com</i>	44.9	0.007
TM_m LM_m	46.4	0.001
TM_{g+m} LM_m	47.2	0.75
TM_{g+m} LM_{g+m}	47.3	

Table 2: BLEU scores of the different initial SMT systems when translating the test corpus from the medical domain.

The same conclusion about the importance of in-domain data can be derived from the results obtained with TM_g built on the sole out-of-domain data. A 10 points BLEU improvement is indeed obtained using LM_m instead of LM_g . Interpolating the two LMs introduces noise and decreases by 1 BLEU point the result obtained with LM_m only. Finally, let us note that the online system GOOGLE TRANSLATE has a BLEU score only 1.5 points lower than a PBMT system built using small-sized but highly relevant data.

5 Post-Editing Translations

After the translation step, SMT outputs are post-edited. Several SPE models are built from the translations of the EMEA training corpus generated by each SMT system. We decide to compute two scores: a first one for which all the sentences from the test corpus are post-edited, and a second one for which only sentences are post-edited if their sentence-level BLEU is improved (*oracle*). The computation of this *oracle* score relies on the reference translation and is done to estimate the potential of SPE.

5.1 Online System

The online translation tool already leads to good results in terms of BLEU score. The in-domain test corpus translated by the online system is post-edited by its SPE system. The results are shown in Table 3. Computing *p*-values to compare results before and after SPE exhibits a significant difference ($p = 0.001$ for BLEU and $p = 0.05$ for the *oracle* score).

System	% BLEU (<i>oracle</i>)
<i>com</i>	44.9
+ $SPE_m LM_m$	46.8 (53.3)
+ $SPE_m LM_{g+m}$	47.9 (53.5)

Table 3: BLEU scores of SPE on the online system output.

Two SPE systems are built with a different LM. With the medical LM ($SPE_m LM_m$), the BLEU score of the post-edited translation reaches 46.8%, around 2 points above the SMT output BLEU score. The *oracle* score indicates that more than 6 BLEU points can still be gained if the post-editing is only applied to the improvable subset of sentences from the test corpus. Introducing the out-of-domain LM with $SPE_m LM_{g+m}$ leads to

a BLEU score of 47.9%. The highest BLEU score obtained by an initial SMT (47.2% with the system $TM_{g+m} LM_m$) is already overtaken by this last SPE system jointly used with the *com* SMT system. Since the *oracle* scores indicate that the highest gain can be reached by the SPE system with the interpolated LM, we will focus on this configuration for our experiments on sentence selection described in Section 6.

5.2 Out-of-Domain PBMT System

This section describes the post-editing of out-of-domain PBMT system outputs, for which medical data are only employed to build LMs. For each LM used during the translation step, we evaluate the impact of the proposed SPE approach.

5.2.1 Out-of-Domain LM

The first evaluation of SPE on the out-of-domain PBMT system is done with $TM_g LM_g$ relying only on out-of-domain data to build its statistical models. We introduce the in-domain data during the SPE step, in the SPE model, in the LM, or in both. The results are presented in Table 4. We can see

System	% BLEU (<i>oracle</i>)
$TM_g LM_g$	29.9
+ $SPE_m LM_m$	43.4 (44.2)
+ $SPE_m LM_{g+m}$	45.6 (47.0)

Table 4: BLEU scores of SPE on the out-of-domain PBMT system using an out-of-domain LM.

that introducing in-domain data during the post-editing step increases the BLEU score of the translated test corpus. From a baseline at 29.9% of BLEU, the SPE systems lead to an absolute improvement of 13.5 and 15.7 points depending on the SPE data configuration. Using the interpolated LMs for the SPE system shows the highest BLEU score, both with a *naive* application of SPE or for the *oracle* score. Let us note that the difference between $SPE_m LM_m$ and $SPE_m LM_{g+m}$ is statistically significant since it is associated with a *p*-value of 0.001. However, these results are lower than the BLEU score obtained by the specialized translation system ($TM_m LM_m$) presented in Table 2.

5.2.2 In-Domain LM

The second evaluation of SPE on the out-of-domain PBMT system concerns $TM_g LM_m$,

where in-domain data are introduced during the SMT process through the LM. The baseline is 39.2% of BLEU and the results presented in Table 5 show that 3.5 BLEU points are gained by the SPE step with a system built on medical data only. We performed the pairwise comparisons with BLEU and the *oracle* score and observed that SPE_mLM_m is statistically equivalent to SPE_mLM_{g+m} with $p > 0.1$ for both metrics. Again, these results are lower than the BLEU score obtained by the specialized translation system (TM_mLM_m) presented in Table 2.

System	% BLEU (<i>oracle</i>)
TM_gLM_m	39.2
+ SPE_mLM_m	42.7 (44.2)
+ SPE_mLM_{g+m}	42.5 (44.4)

Table 5: BLEU scores of SPE on the out-of-domain PBMT system using a medical LM.

5.3 In-Domain and Mixed PBMT Systems

After our experiments on the out-of-domain PBMT system using different LMs, we focus on the post-editing of in-domain PBMT system output. Two systems are studied here, one using only in-domain data (TM_mLM_m) and the other using both out and in-domain data ($TM_{g+m}LM_m$). For TM_mLM_m , the baseline BLEU score is 46.4% and none of the tested SPE configuration was able to increase this score. However, the *oracle* scores measured resp. at 47.4% and 47.5% with SPE_mLM_{g+m} and SPE_mLM_m show the potential improvement using SPE. This aspect motivates our sentence selection approach presented in Section 6.

As far as $TM_{g+m}LM_m$ is concerned, the use of the interpolated LM in the post-editing step (SPE_mLM_{g+m}) degrades the BLEU score by 0.8 point, while the use of the medical LM (SPE_mLM_m) does not statistically improve the baseline BLEU measured before SPE. For both configuration, the *oracle* score shows that a significant gain is still possible.

6 Selecting Sentences for Post-Editing

Post-editing selected sentences is motivated by the *oracle* scores measured in Section 5. We propose to build a classifier in order to partition sentences according to the possible BLEU gain with SPE. To

train such a classifier, we use the medical development corpus and compute for each sentence its associated $\Delta BLEU$ score comparing BLEU before and after SPE. It is a binary classification task: if the $\Delta BLEU$ score is positive, i.e. SPE improves the sentence, the sentence is labelled Class 1; otherwise, the sentence is tagged with Class 2. Figure 1 illustrates the general architecture of our system.

The classifier used in our experiments is a Support Vector Machine (SVM) (Boser et al., 1992) based on a linear kernel. We use the implementation of *libSVM* (Chang and Lin, 2011) in the WEKA (Hall et al., 2009) environment (El-Manzalawy and Honavar, 2005). The translated (by the MT system *com*) in-domain development set is used to build a sentence-level post-edition model. Each sentence of the training corpus is considered as a vector composed of n -grams ($n \in [1; 3]$).

We decided to apply the classification method to the highest *oracle* score observed in Section 5, i.e. the *com* translation system jointly used with a SPE_mLM_{g+m} post-editing step. The *oracle* score for this configuration reaches 53.5%, while the *naive* application of SPE leads to a BLEU score of 47.9%. The test set translated by *com* is classified using SVM, where each sentence is associated with a normalized score for each of the two classes. Using the translation reference, we evaluate the classifier in terms of recall and precision. The recall reaches 79.5% and the precision 40.1%. In order to evaluate the gain in terms of BLEU on the whole test set, we decide to post-edit sentences according to their Class 1 scores given by the SVM. This score is the probability to improve BLEU at the sentence level. The evaluation can be repeated individually for each 0.1 score span (*is*, only the sentences in this exact range are post-edited) and then cumulated over consecutive spans (*cs*, all sentences above the threshold are post-edited). The results are displayed in Figure 2.

The cumulated span evaluation shows that post-editing the sentences above a prediction score of 0.8 reaches the highest BLEU score. With this configuration, 1 BLEU point is gained compared to the *naive* application of SPE (from 47.9% to 48.9% of BLEU). The amount of sentences in each class is increasing between 0.5 and 0.8. Only 60 sentences remain in Class 1 with a prediction score above 0.9. The amount of training sentences

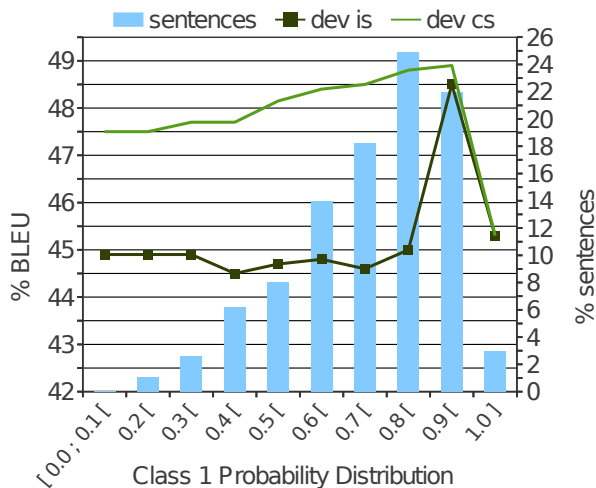


Figure 2: BLEU scores and amount of sentences classified in Class 1 for individual (*is*) and cumulated (*cs*) spans obtained on the test corpus.

in each class is an important aspect of the classifier accuracy. Figure 3 shows TER (Snover et al., 2006) and inverted BLEU scores of Class 1 sentences with a classification score over 0.8, before and after post-editing.

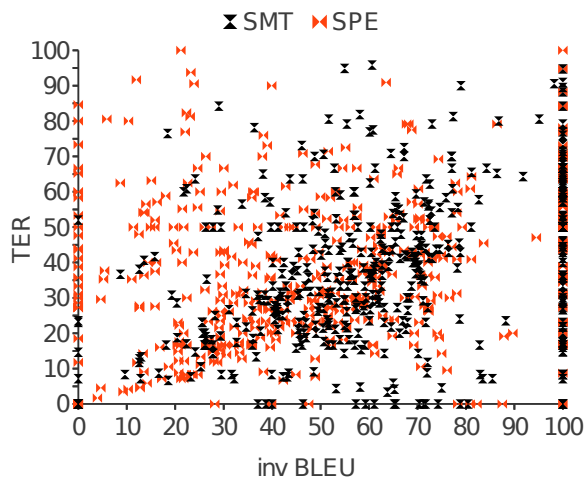


Figure 3: TER and *inverted* BLEU sentences distribution measured on the test corpus when Class 1 probability is over 0.8.

It clearly appears that there are more post-edited than translated sentences with a 100% BLEU score (0% inverted BLEU): resp. 47 and 11 sentences. Also among the 109 translated sentences with a 0% BLEU score, only half remains at this level after post-editing. The evaluation on the test set shows a general improvement using both metrics, as detailed in Table 6. These final results present the possible gain in terms of translation quality with

SPE and a classification approach. The comparison between the SPE systems with and without classification shows that the combination of SPE and SVM is better than the *naive* application of SPE with $p = 0.004$.

	SMT	+ SPE	+ SVM
TER	42.3	40.4	39.7
BLEU	44.9	47.9	48.9

Table 6: TER and BLEU scores on the test set after translation, post-editing and classification (with $p(\text{Class1}) \geq 0.8$).

7 Conclusion and Future Work

In this paper, we have presented a phrase-based post-editing approach for specific domain adaptation. Our experiments show that an out-of-domain translation system can be adapted *a posteriori* through a *naive* application of the proposed SPE approach. *Oracle* scores indicate that gains in terms of BLEU score are still possible, even with a PBMT system built on in-domain data and without introducing new data during the post-editing step. The highest BLEU score is obtained using GOOGLE TRANSLATE combined with an SPE system ($SPE_m LM_{g+m}$) and a classification step. Compared to the baseline, the BLEU score is increased by 4 BLEU points. Compared to the best PBMT system ($TM_{g+m} LM_{g+m}$) with 47.3% of BLEU, the score is increased by 1.6 BLEU points (with $p = 0.001$). In a future work, other metrics will be used to measure the translation quality at the sentence level. We also want to introduce more features into the classifier training set based on quality estimation techniques for our sentence selection approach, in order to better fill the gap between the current BLEU and the *oracle* score.

References

- Allen, J. and C. Hogan. 2000. Toward the development of a post editing module for raw machine translation output: A controlled language perspective. In *CLAW*, pages 62–71.
- Béchara, H., Y. Ma, and J. van Genabith. 2011. Statistical post-editing for a statistical MT system. In *MT Summit XIII*, pages 308–315.
- Boser, B.E., I.M. Guyon, and V.N. Vapnik. 1992. A training algorithm for optimal margin classifiers. In *5th annual workshop on Computational learning theory*, pages 144–152.

- Chang, Chih-Chung and Chih-Jen Lin. 2011. LIB-SVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- de Ilarraza, A.D., G. Labaka, and K. Sarasola. 2008. Statistical postediting: A valuable method in domain adaptation of RBMT systems for less-resourced languages. In *MATMT*, pages 35–40.
- Dugast, L., J. Senellart, and P. Koehn. 2007. Statistical post-editing on Systran’s rule-based translation system. In *WMT*, pages 220–223.
- Dugast, L., J. Senellart, and P. Koehn. 2009. Statistical post editing and dictionary extraction: Systran/Edinburgh submissions for ACL-WMT2009. In *WMT*, pages 110–114.
- EL-Manzalawy, Y. and V. Honavar, 2005. *WLSVM: Integrating LibSVM into Weka Environment*. Software available at <http://www.cs.iastate.edu/~yasser/wlsvm>.
- Gao, Q. and S. Vogel. 2008. Parallel implementations of word alignment tool. In *ACL Workshop: Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57.
- Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten. 2009. The WEKA data mining software: An update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.
- Isabelle, P., C. Goutte, and M. Simard. 2007. Domain adaptation of MT systems through automatic post-editing. In *MT Summit XI*, pages 255–261.
- Knight, K. and I. Chander. 1994. Automated postediting of documents. In *NCAI*, pages 779–779.
- Koehn, P., F.J. Och, and D. Marcu. 2003. Statistical phrase-based translation. In *NAACL-HLT*, volume 1, pages 48–54.
- Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL*, pages 177–180.
- Koehn, P. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit X*, volume 5, pages 79–86.
- Marcu, D. and W. Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *EMNLP*, volume 10, pages 133–139.
- Och, F.J. and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.
- Och, F.J. 2003. Minimum error rate training in statistical machine translation. In *ACL*, volume 1, pages 160–167.
- Ofizer, K. and I.D. El-Kahlout. 2007. Exploring different representational units in English-to-Turkish statistical machine translation. In *WMT*, pages 25–32.
- Papineni, K., S. Roukos, T. Ward, and W.J. Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *ACL*, pages 311–318.
- Rafalovitch, A. and R. Dale. 2009. United Nations general assembly resolutions: A six-language parallel corpus. *MT Summit XII*, pages 292–299.
- Sadat, F., J.H. Johnson, A. Agbago, G. Foster, R. Kuhn, J. Martin, and A. Tikuisis. 2005. PORTAGE: A phrase-based machine translation system. In *The ACL 2005 Workshop on Building and Using Parallel Texts: Data-Driven Machine Translation and Beyond*.
- Sager, J.C., D. Dungworth, and P.F. McDonald. 1980. English special languages: Principles and practice in science and technology.
- Simard, M., C. Goutte, and P. Isabelle. 2007a. Statistical phrase-based post-editing. In *NAACL-HLT*, pages 508,515.
- Simard, M., N. Ueffing, P. Isabelle, and R. Kuhn. 2007b. Rule-based translation with statistical phrase-based post-editing. In *WMT*, pages 203–206.
- Snover, M., B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *AMTA*, pages 223–231.
- Stolcke, A. 2002. SRILM—an extensible language modeling toolkit. In *InterSpeech*, volume 2, pages 901–904.
- Stroppa, N., K. OwczarBézak, and A. Way. 2007. A cluster-based representation for multi-system MT evaluation. In *TMI*, pages 221–230.
- Suzuki, H. 2011. Automatic post-editing based on SMT and its selective application by sentence-level automatic quality evaluation. In *MT Summit XIII*, pages 156–163.
- Tiedemann, J. 2009. News from OPUS—a collection of multilingual parallel corpora with tools and interfaces. In *RANLP*, volume V, pages 237–248.
- Zens, R., F. Och, and H. Ney. 2002. Phrase-based statistical machine translation. *KI 2002: Advances in Artificial Intelligence*, pages 35–56.

Crowd-based MT Evaluation for non-English Target Languages

Michael Paul and Eiichiro Sumita

NICT

Hikaridai 3-5

619-0289 Kyoto, Japan

<Firstname>.<Lastname>@nict.go.jp

Luisa Bentivogli and Marcello Federico

FBK-irst

Via Sommarive, 18

38123 Povo-Trento, Italy

{bentivo, federico}@fbk.eu

Abstract

This paper investigates the feasibility of using crowd-sourcing services for the human assessment of machine translation quality of translations into *non-English* target languages. Non-expert graders are hired through the CrowdFlower interface to Amazon’s Mechanical Turk in order to carry out a ranking-based MT evaluation of utterances taken from the travel conversation domain for 10 Indo-European and Asian languages. The collected human assessments are analyzed for their worker characteristics, evaluation costs, and quality of the evaluations in terms of the agreement between non-expert graders and expert/oracle judgments. Moreover, data quality control mechanisms including “locale qualification” “qualification testing”, and “on-the-fl verification are investigated in order to increase the reliability of the crowd-based evaluation results.

1 Introduction

This paper focuses on the evaluation of machine translation (MT) quality for target languages other than English. Although human evaluation of MT output provides the most direct and reliable assessment, it is time consuming, costly, and subjective. Various automatic evaluation measures were proposed to make the evaluation of MT outputs cheaper and faster (Przybocki et al., 2008), but automatic metrics have not yet proved able to consistently predict the usefulness of MT technologies. To counter the high costs in human assessment of MT outputs, the usage of crowdsourcing services such as Amazon’s Mechanical Turk¹ (MTurk) and CrowdFlower² (CF) were proposed recently (Callison-Burch, 2009; Callison-Burch et al., 2010; Denkowski and Lavie, 2010).

¹<http://www.mturk.com>

²<http://crowdflower.com>

The feasibility of crowd-based MT evaluations was investigated for shared tasks such as the WMT (Callison-Burch, 2009) and the IWSLT (Federico et al., 2011) evaluation campaigns. Their results showed that agreement rates for non-experts were comparable to those for experts, and that the crowd-based rankings correlated very strongly with the expert-based rankings. Most of the crowd-based evaluation experiments focused on English as the target language, with the exception of (Callison-Burch et al., 2010) evaluating Czech, French, German, and Spanish translation outputs and (Federico et al., 2011) evaluating translations into French.

This paper investigates the feasibility of using crowdsourcing services for the human assessment of translation quality of translation tasks where the target language is *not* English, with a focus on non-European languages. In order to identify non-English target languages for which we can expect to find qualified workers, we referred to existing surveys that analyze the demographics of MTurk workers (see Section 2). In total, we selected 7 non-European languages consisting of Arabic (ar), Chinese (zh), Hindi (hi), Japanese (ja), Korean (ko), Russian (ru), and Tagalog (tl), as well as 3 European languages covering English (en), French (fr), and Spanish (es) as the target languages for our translation experiments.

The MT evaluation was carried out using utterances taken from the domain of travel conversations. A description of the utilized language resources and the MT engines are summarized in Section 3. The translation quality of the MT engines was evaluated using (1) the automatic evaluation metric BLEU (Papineni et al., 2002) and (2) human assessment of MT quality based on the *Ranking* metric (Callison-Burch et al., 2007).

For the 10 investigated language pairs, non-expert graders were hired through the CF interface to MTurk in order to carry out the ranking-based MT evaluation as described in Section 4. In addition, expert graders were employed for four of

the target languages (en, ja, ko, zh) to carry out exactly the same evaluation task as the non-expert workers. For all target languages without expert graders, we used an oracle ranking metric based on the “Training Size Preference” assumption, i.e., *the larger the training size, the better the translation quality can be expected to be*, to evaluate the quality of the worker judgments.

Besides a thorough analysis of the obtained non-expert grading results, we also investigated different data quality control mechanisms in order to increase the reliability of crowd-based evaluation results (see Section 5). The experiments carried out in this paper revealed that the quality of the crowd-based MT evaluation is closely related to the demographics of the online work marketplace. Although high-quality evaluation results could be collected for the majority of the investigated non-English languages, the need for multi-layered data quality control mechanisms causes an increase in evaluation time. The finding of this paper confirms that crowdsourcing is an effective way of reducing the costs of MT evaluation without sacrificing quality even for non-English target languages given that control mechanisms carefully tailored to the evaluation task at hand are in place.

2 Mechanical Turk Demographics

Past surveys on the demographics of MTurk users indicated that most of the workers come from the US. (Ipeirotis, 2010) conducted a recent survey on the demographics of MTurk users which showed a shift in the “country of origin” of workers, i.e., a decrease in US workers to 47% and an increase of Indian workers to 34%, with the remaining 19% of workers coming from 66 different countries³. Based on the country information from MTurk workers taking part in the survey, we analyzed which languages are used by these workers.

The language distribution shows that the majority of workers speak English, followed by Hindi, Romanian, Tagalog, and Spanish. At least 5 workers were native speakers of Dutch, Arabic, Italian, German, and Chinese. However, taking into account official languages spoken in the respective countries, we can expect larger contributions of workers speaking Spanish, French, and Arabic.

3 MT Evaluation Task

The crowd-based MT evaluation is carried out using the translation results of phrase-based statis-

tical machine translation (SMT) systems that are trained on parallel corpora. The translation quality of SMT engines heavily depends on the amount of bilingual language resources available to train the statistical models. We exploited this characteristic of data-driven MT approaches to define an “oracle” ranking metric (ORACLE) according to the “Training Size Preference” assumption, in which an MT output of a system A wins (or ties in) a comparison with the MT output of a system B, where the training corpus of system B is a subset of the one of system A.

The language resources used to build MT engines are described in Section 3.1. We selected 10 Indo-European and Asian languages based on the following criteria:

- “*Worker Availability*” covering languages with ‘many’ (en, hi), ‘several’ (es, tl), ‘few’ (ar, fr, ja, ru, zh), ‘almost none’ (ko) MTurk workers available.
- “*Usage for MT Research*” covering ‘frequently’ (ar, fr, zh), ‘often’ (es, ru), ‘sporadically’ (ja, ko) used languages as well as under-resourced languages (tl, hi).
- “*Availability of Language Resources*” used for the training and evaluation of MT engines.

The training corpus consisting of 160k relatively short sentences was split into three subsets of 80k, 20k, and 10k sentence pairs, respectively. Each subset was used to train an MT engine whose translation quality significantly differed from the others, with the MT engine trained on the full corpus achieving the best translation quality.

This translation experiment setup renders the manual evaluation relatively reliable due to (1) a relatively easy translation task and (2) large differences in translation performance between the utilized MT engines. Moreover, the ORACLE metric can be exploited to judge the quality of crowd-based evaluation results for all languages where expert graders were not available.

3.1 Language Resources

The crowd-based MT evaluation experiments are carried out using the multilingual *Basic Travel Expressions Corpus* (BTEC), which is a collection of sentences that bilingual travel experts consider useful for people going to or coming from another country (Kikui et al., 2006). The sentence-aligned corpus consists of 160k sentences and covers all 10 languages investigated in this paper.

The parallel text corpus was randomly split into three subsets: for evaluating translation quality (*eval*, 300 sentences), for tuning the SMT model weights (*dev*, 1000 sentences) and for training the

³Details on the survey can be found at <http://hdl.handle.net/2451/29585>

statistical models (*train*, 160k sentences). Furthermore, three subsets of varying sizes (80k, 20k, and 10k sentences) were randomly extracted from the training corpus and used to train four SMT engines on the respective training data sets for each of the investigated language pairs.

3.2 Translation Engines

The translation results evaluated in this paper were obtained using fairly typical phrase-based SMT engines built within the framework of a feature-based exponential model. For the training of the SMT models, standard word alignment (Och, 2003) and language modeling (Stolcke, 2002) tools were used. Minimum error rate training (MERT) was used to tune the decoder’s parameters and was performed on the *dev* set using the technique proposed in (Och, 2003). For the translation, an in-house multi-stack phrase-based decoder was used.

In order to maximize the gains⁴ from an increased training data size and therefore allow for reliable ORACLE judgments, we selected English as the source language for the translations into Arabic, Japanese, Korean, and Russian. For all other translation experiments, Japanese source sentences were used as the input for the SMT decoder.

3.3 Automatic Evaluation

For the automatic evaluation of translation quality, we applied the BLEU metric (Papineni et al., 2002). Scores range between 0 (worst) and 1 (best).

The results of the translation engines described in Section 3.2 are summarized in Table 1, where the BLEU scores are given as percent figures (%BLEU). The obtained scores confirm the “Training Size Preference” assumption (160k>80k>20k>10k) of the ORACLE metric. Concerning the target languages, the highest BLEU scores were achieved for Korean and Japanese, followed by English, Chinese, Spanish and French. Arabic and Hindi seem to be the most difficult target languages for the given translation and evaluation tasks obtaining the lowest automatic evaluation scores for each of the investigated tasks.

3.4 Subjective Evaluation

Human assessments of translation quality were carried out using the *Ranking* metrics where human graders were asked to “rank each whole sentence translation from Best to Worst relative to the

⁴For relatively simple translation tasks, the amount of training data affects the translation quality of closely related languages far less than for more distinct languages.

Table 1: Translation Quality (%BLEU)

Language		MT Engine			
Source	Target	160k	80k	20k	10k
en	ar	12.90	12.45	10.89	9.97
	ja	28.58	25.38	21.00	19.41
	ko	29.53	26.42	21.43	18.66
	ru	16.15	15.84	13.90	12.36
ja	en	24.47	19.95	15.35	12.57
	es	19.52	17.43	13.30	11.73
	fr	19.35	18.84	14.67	14.43
	hi	14.17	12.57	9.97	8.24
	tl	18.93	17.81	15.78	13.58
	zh	21.22	17.08	13.03	12.64

other choices (ties are allowed)” (Callison-Burch et al., 2007).

The unit of evaluation was the *ranking set*, which is composed of a source sentence, the main reference provided as an acceptable translation, and the MT outputs of all four MT engines to be judged. The order of the MT outputs was changed randomly for each ranking set to avoid bias. The *Ranking* evaluation was carried out using a web-browser interface and graders had to order four system outputs by assigning a grade between 1 (*best*) and 4 (*worse*).

4 Crowd-based MT Evaluation

To counter the high costs in human assessment of MT outputs, crowdsourcing services such as MTurk and CF have attracted a lot of attention both from industry and academia as a means for collecting data for human language technologies at low cost. MTurk is an on-line work marketplace, where people are paid small sums of money to work on Human Intelligence Tasks (HITs), i.e. tasks that machines have hard time doing. The CF platform works across multiple crowdsourcing services, including MTurk. CF gives unrestricted access, making it possible for non US-based requesters to place HITs on MTurk.

4.1 Data Quality Control Mechanism

One of the most crucial issues to consider when collecting crowdsourced data is how to ensure their quality. MTurk and CF provide requesters with quality control mechanisms including the “locale qualification” option to restrict workers by country. Preliminary qualification for workers can be set by requiring workers to complete a qualification test using training ranking sets. Only workers passing the test are allowed to accept a HIT for the evaluation task at hand. Moreover, CF provides a mechanism to verify the workers’ reliability on-the-fly. The HIT design interface provided by CF allows including so called “gold units”, i.e. items

with known labels, along with the other units composing the requested HIT. Gold units are randomly mixed with the other units by CF when it creates the worker assignments. These control units⁵ allows distinguishment between trusted workers (those who correctly replicate the gold units) and untrusted workers (those who fail the gold units). Untrusted workers are automatically blocked and not paid, and their labels are filtered out from the final data set. CF uses the workers' history to apply confidence scores (the "trust level" feature) to their annotations. In order to be considered trusted in a job, workers are required to judge a minimum of four gold units and to be above an accuracy threshold of 70%. As a further control, CF pauses a job (the "auto-takedown" feature), if workers are failing too many gold units.

In this paper, we investigated the dependency of the quality of the evaluation results for the following quality control features:

- *locale qualification* (LOC): restriction to official language countries; the most important control mechanism to prevent workers from tainting the evaluation results.
- *qualification testing* (PRI): training phase assessment of worker's eligibility prior to the evaluation task.
- *on-the-fly verification* (GOLD): identification of trusted workers using control units with a known answer.

4.2 Control Units

Control units have to be unambiguous, not too trivial, and also not too difficult. For the translation task at hand, we selected the original corpus sentence as the main reference translation. From paraphrased reference translations⁶, we selected a single reference as the *gold translation* to be included in the control units. A paraphrased reference to be selected as a gold translation should have the following characteristics: (1) it should be similar to the main reference and (2) its translation quality should be better than the best MT output for all translation hypotheses of the same input. If native speakers are available, the gold translation quality should be checked manually. However, for most of the investigated target languages, native speakers were not available. Thus, we automatically selected a gold translation based on the edit distance of each paraphrased reference to (a) the main reference and (b) the ORACLE-best (=160k) MT output for all sentence IDs of the *eval* set. We selected the most appropriate paraphrased reference according

⁵The suggested amount of gold units to be provided is around 10% of the requested units.

⁶Up to 15 paraphrased reference translations are available for the data sets described in Section 3.1.

to its minimal distance to the main reference and its maximal distance to the MT output. The top-30 sentence IDs with the best gold translation distance scores were selected as control units for the respective translation task.

For each control unit sentence ID, a random MT output was replaced in the ranking set with the gold translation. For our experiments, we distinguished two GOLD annotation schemes:

- "best-only" (GOLD^b): check only the best translation, i.e., force rank '1' assignment for the gold translation.
- "best+worse" (GOLD^{bw}): check the best and the worst translation, i.e., allow rank '1' or '2' for the gold and rank '3' or '4' for the ORACLE-worst (10k) translation.

4.3 Evaluation Interface

CF provides two interfaces: (1) an *external* one for MTurk workers and (2) an *internal* one for which you have to prepare your own work force. The internal interface is (currently) free of charge and was used to collect judgments from in-house expert graders using exactly the same HITs and the same online interface as the MTurk workers.

4.4 Experiment Setup

For each target language (TRG), we repeated the same MT evaluation experiment using the following data quality control settings⁷:

1. *NONE*: no quality control (all TRGs)
2. *GOLD*: on-the-fly only (all TRGs)
3. *LOC+GOLD*: locale+on-the-fly (all TRGs)
4. *LOC+GOLD+PRI*: locale+testing+on-the-fly (hi, ko)

All experiments using the same control settings were carried out simultaneously, i.e., a single worker might take part in more than one evaluation experiment. A HIT consisted of 3 ranking sets per page and is paid 6 cents for all experiments. In total, the evaluation costs⁸ for all the experiments added up to \$390 for 30 experiments, resulting in an average of \$13 for the crowd-based evaluation of 4 MT outputs for 300 input sentences.

5 Evaluation Results

In order to investigate the effects of the data quality control mechanisms, the analysis of the evaluation results is conducted experiment-wise. i.e., we do not differentiate between single workers, but treat all the collected judgments of the respective experiment as a "single" grader result. This enables a

⁷India was excluded by default for all experiments besides the ones having Hindi as the target language.

⁸The requester's payment includes a fee to MTurk of 10% of the amount paid to the workers. In addition, CF takes a 33% share of the payments by the requester.

comparison of non-expert vs. expert/oracle grading results and the impact of each control setting on the quality of the collected judgments. The details of the experiment results for each target language are listed in Appendix A.

5.1 Worker Characteristics

Table A.1. summarizes the amount of participating workers. For each control setting, we list the amount of workers (*total*) and the percentage of workers coming from a country where the language is the official language (*native*). The worker demographics are summarized in Table A.2.

Without any control mechanism in place, the judgments mainly originated from non-native workers. 53% of the workers submitted HITs for at least two tasks, with the largest overlap being five tasks. Although some workers might be able to speak and evaluate more than two languages, the results indicate that *the larger the overlap, the less reliable the judgments are expected to be*.

The on-the-fly verification based on gold translations only (*GOLD^b*) resulted in a high percentage of judgments obtained from trusted workers (65~100%) for the majority of tasks, but achieved worse figures with respect to native worker contributions. These findings indicate that *single gold translations are not sufficient to identify workers assigning grades based on fixed patterns*.

As a counter-measure, we limited the worker origin to the official language countries and the US, and annotated both the best and worst translation of the control units. As a result, 47% of the *LOC+GOLD^{bw}* gradings were collected from native speakers. These results show that the locale and on-the-fly control enable the collection of less tainted judgments and the identification of untrusted workers, respectively. Table A.3. summarizes the amount of judgments collected for each task. The total count depends on the number of non-trusted workers accepting HITs for the respective language.

Although high-quality control units positively affect the quality of the evaluations as shown in Section 5.2, the average time needed to collect the data increased by a factor of 8. The evaluation period, i.e., the number of days needed to collect all the data, the grading time, i.e., the hours spent on actually grading the translations, and the average grading time per assignment are summarized in Table A.4. The grading time for each task ranged from 2.5h to 6.5h for the *LOC+GOLD^{bw}* experiments. However, the evaluation period largely

depends on the language, ranging from 2 days (hi, tl, es) to over 2 weeks (ru, zh, ko). The analysis of the average time needed to judge a single HIT indicates that *the shorter the evaluation time, the less reliable the judgments are expected to be*.

The most problematic languages are Korean and Hindi. For Korean, the evaluation experiments lasted 3 months due to the lack of trusted workers. Moreover, the Hindi *LOC+GOLD^{bw}* task could not be finished because the large amount of untrusted workers triggered CF’s *auto-takedown* feature. In order to prevent an auto-takedown for jobs where low trust levels of workers are to be expected, a training phase assessing the worker’s eligibility prior to the evaluation task needs to be included. Only workers passing the qualification test were allowed to accept HITs for the respective task. The Korean and Hindi results given in Appendix A were therefore obtained using the *LOC+GOLD^{bw}+PRI* data quality control setting.

5.2 Ranking Results

The *Ranking* scores were obtained as the average number of times that a system was judged better than any other system. The results summarized in Table A.5. differ largely for the investigated data quality settings. System ranking scores resulting in an MT system ordering other than the expert rankings are marked in boldface. For most of the uncontrolled tasks, worker rankings are different from expert rankings. The *GOLD^b* setting tasks achieved a higher correlation with expert rankings, but still differ for 3 out of the 10 languages. The *LOC+GOLD^{bw}* tasks ranked all the MT systems identically to the experts. Interestingly, the ranking scores obtained for the better controlled evaluation experiments are much higher, indicating the collected evaluation data is of good quality.

5.3 Grading Consistency

The most informative indicator of the quality of a dataset is given by the agreement rate, or grading consistency, both between different judges and the same judge. To this purpose, the agreement between non-expert graders of experiments using different data quality control mechanisms was calculated for the MTurk data and compared to the results obtained by expert/oracle judgments. Agreement rates are calculated using the *Fleiss’ kappa coefficient* κ (Fleiss, 1971):

$$\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)},$$

where $\Pr(a)$ is the observed agreement among graders, and $\Pr(e)$ is the hypothetical probability of

chance agreement. In our task, $\text{Pr}(a)$ is given by the proportion of times that two judges assessing the same pair of systems on the same source sentence agree that $A > B$, $A = B$, or $A < B$. Grader agreement scores can be interpreted as follows: “none” $\kappa < 0$, “slight” $\kappa \leq 0.2$, “fair” $\kappa \leq 0.4$, “moderate” $\kappa \leq 0.6$, “substantial” $\kappa \leq 0.8$, and “almost perfect” $\kappa \leq 1.0$ (Landis and Koch, 1977).

The quality of the judgment is confirmed by the ranking agreement scores listed in Table A.6. Comparing the worker vs. the expert judgments, only *slight* agreement was obtained for the less controlled settings, but the proposed data quality control mechanisms achieved levels of up to *substantial* agreement. The comparison of agreement scores for oracle and expert judgments indicates that at least *fair* agreement is to be expected for languages where expert graders are not available.

6 Conclusions

In this paper, we investigated the use of the data quality control mechanisms of online work marketplaces for the collection of high-quality MT evaluation data for non-English target languages. The analysis of the worker characteristics revealed that *locale qualification* control settings enable the collection of less tainted judgments and that bad workers can be identified by short HIT grading times, large overlaps of evaluation tasks run simultaneously, and low trust levels measured either prior to or during the evaluation task.

Due to the lack of expert graders for 6 out of 10 languages, the creation of control units was carried out automatically, where the proposed similarity-based gold translation selection method proved to be a practical alternative to manual selection by native speakers. The improved setting of control units to verify not only the best but also the worst translation helped to identify untrusted workers using fixed gradings schemes. Finally, the combination of multiple control mechanism proved to be essential for collecting high-quality data for all the investigated non-English languages.

Based on the obtained findings we recommend carrying out crowd-based MT evaluations by (1) limiting the access to workers in countries where the target language is the official language, although for languages lacking workers, the US might be included if evaluation time is a crucial factor and (2) defining control units so that expected rankings for the best and the worst systems are preserved and grading variations of non-expert graders are taken into account.

As future work, we are planning to investigate the effectiveness of other control mechanisms such as *payment* and the applicability of the proposed crowd-based MT evaluation method to more complex translation tasks, ranking more MT systems, as well as covering other domains such as the translation of public speeches.

References

- Callison-Burch, C., C. Forgyce, P. Koehn, C. Monz, and J. Schroeder. 2007. (Meta-) Evaluation of Machine Translation. In *Proc. of the Second Workshop on SMT*, pages 136–158.
- Callison-Burch, C., P. Koehn, C. Monz, K. Peterson, M. Przybocki, and O. Zaidan. 2010. Findings of the 2010 Joint Workshop on SMT and Metrics for MT. In *Proc. of the Joint Fifth Workshop on SMT and Metrics/MATR*, pages 17–53.
- Callison-Burch, C. 2009. Fast, Cheap, and Creative: Evaluating MT Quality Using Amazon’s Mechanical Turk. In *Proc. of the EMNLP*, pages 286–295.
- Denkowski, M. and A. Lavie. 2010. Exploring Normalization Techniques for Human Judgments of Machine Translation Adequacy Collected Using Amazon Mechanical Turk. In *Proc. of the NAACL HLT Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk*, pages 57–61.
- Federico, M., L. Bentivogli, M. Paul, and S. Stücker. 2011. Overview of the IWSLT 2011 Evaluation Campaign. In *Proc. of IWSLT*, pages 11–27.
- Fleiss, J. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76 (5):378–382.
- Ipeirotis, P. 2010. New demographics of Mechanical Turk. <http://hdl.handle.net/2451/29585>.
- Kikui, G., S. Yamamoto, T. Takezawa, and E. Sumita. 2006. Comparative study on corpora for speech translation. *IEEE Transactions on Audio, Speech and Language Processing*, 14(5):1674–1682.
- Landis, J. and G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33 (1):159–174.
- Och, F.J. 2003. Minimum Error Rate Training in SMT. In *Proc. of the 41st ACL*, pages 160–167.
- Papineni, K., S. Roukos, T. Ward, and W. Zhu. 2002. BLEU: a Method for Automatic Evaluation of MT. In *Proc. of the 40th ACL*, pages 311–318.
- Przybocki, M., K. Peterson, and S. Bronsart. 2008. Metrics for MACHINE TRANSLATION Challenge. <http://nist.gov/speech/tests/metricsmatr/2008/results>.
- Stolcke, A. 2002. SRILM: an extensible language modeling toolkit. In *Proc. of the ICSLP*.

Appendix A. Crowd-based MT Evaluation

A.1. Amount of Workers

The total number of participating workers, as well as the number and the percentage of trusted/native workers for each evaluation task.

TRG	Data Quality Control Mechanism								
	total count	LOC+GOLD ^{bw}		total count	GOLD ^b		total count	NONE	
		trusted [% of total]	native [% of total]		trusted [% of total]	[native] [% of total]		trusted [% of total]	native [% of total]
en	23	18 (78.3%)	13 [56.5%]	38	30 (78.9%)	10 [26.3%]	8	–	4 [50.0%]
ar	41	26 (63.4%)	23 [56.0%]	29	19 (65.5%)	6 [20.6%]	14	–	0 [0.0%]
es	19	19 (100.0%)	15 [78.9%]	12	11 (91.6%)	2 [16.6%]	8	–	0 [0.0%]
fr	10	9 (90.0%)	4 [40.0%]	10	9 (90.0%)	0 [0.0%]	14	–	2 [14.2%]
hi	31*	28* (90.3%)	27* [87.0%]	85	37 (43.5%)	34 [40.0%]	47	–	33 [70.2%]
ja	14	11 (78.5%)	3 [21.4%]	15	13 (86.6%)	0 [0.0%]	10	–	1 [10.0%]
ko	45*	43* (95.5%)	2* [4.4%]	24	17 (70.8%)	0 [0.0%]	5	–	0 [0.0%]
ru	30	20 (66.6%)	4 [13.3%]	7	7 (100.0%)	0 [0.0%]	14	–	0 [0.0%]
tl	10	9 (90.0%)	5 [50.0%]	6	6 (100.0%)	0 [0.0%]	2	–	1 [50.0%]
zh	18	11 (61.1%)	3 [16.6%]	16	12 (75.0%)	0 [0.0%]	7	–	0 [0.0%]

* marked results are obtained using the LOC+GOLD^{bw}+PRI data quality control setting.

A.2. Country of Origin

The total number of countries and workers per country participating in each evaluation task.

TRG	Data Quality Control Mechanism		
	LOC+GOLD ^{bw} country: workers	GOLD ^b country: workers	NONE country: workers
en	9 countries USA:15, AUS:1, CAN:1, GBR:1, MYS:1, PHL:1, BGD:1, CMR:1, SGP:1	11 countries USA:15, MKD:9, CHN:2, NLD:2, JPN:2, PAK:2, AUS:1, BGD:1, CMR:1, MDV:1	4 countries USA:5, AUS:1, JPN:1, MKD:1
ar	11 countries JOR:12, EGY:8, USA:7, TUN:3, LBN:3, SAU:2, MAR:2, DZA:1, KWT:1, ARE:1, OMN:1	15 countries MKD:6, TUN:3, JOR:3, EGY:2, USA:2, BGD:2, ARE:2, GBR:2, DZA:1, CHN:1, ESP:1, MDV:1, ROU:1, OMN:1, SAU:1	10 countries MKD:3, EGY:2, PAK:2, CHN:1, DZA:1, GBR:1, LBN:1, TUN:1, ARE:1, USA:1
es	8 countries ESP:5, MEX:4, USA:4, COL:2, ARG:1, GTM:1, URY:1, VEN:1	5 countries MKD:7, ESP:2, USA:1, BGD:1, ROU:1	7 countries USA:2, BHS:1, ESP:1, PRT:1, MKD:1, PAK:1, ROU:1
fr	4 countries USA:5, FRA:3, CAN:1, CMR:1	5 countries MKD:6, USA:1, CMR:1, NLD:1, ROU:1	8 countries MKD:3, PAK:3, FRA:2, ROU:2, CAN:1, CMR:1, NLD:1, USA:1
hi	2 countries IND:30*, USA:1*	4 countries IND:80, PAK:3, USA:1, ROU:1	8 countries IND:33, MKD:6, CHN:2, PAK:2, SGP:1, ARE:1, ROU:1, USA:1
ja	2 countries USA:10, JPN:4	8 countries MKD:6, ROU:2, PAK:2, BGD:1, CHN:1, JPN:1, MDV:1, NLD:1	5 countries USA:4, JPN:2, MKD:2, PAK:1, PHL:1
ko	2 countries USA:41, KOR:2	10 countries MKD:9, ROU:3, PHL:3, USA:2, CHN:2, POL:1, BGD:1, MDV:1, PAK:1, ESP:1	3 countries CHN:2, USA:2, MKD:1
ru	2 countries USA:25, RUS:5	5 countries PAK:2, ROU:2, GBR:1, SRB:1, MKD:1	7 countries MKD:8, MDA:1, POL:1, SRB:1, UKR:1, CHN:1, PAK:1
tl	2 countries PHL:7, USA:3	3 countries MKD:3, ROU:2, PAK:1	1 country PHL:2
zh	4 countries USA:12, CHN:3, SGP:2, HKG:1	6 countries MKD:9, USA:3, ROU:1, NLD:1, CHN:1, BGD:1	4 countries USA:3, CHN:2, SGP:1, MKD:1

* marked results are obtained using the LOC+GOLD^{bw}+PRI data quality control setting.

A.3. Judgments

The total number of rankings sets judged by all/trusted/native workers for each evaluation task.

TRG	Data Quality Control Mechanism								
	total count	LOC+GOLD ^{bw}		total count	GOLD ^b		total count	NONE	
		trusted [% of total]	native [% of total]		trusted [% of total]	native [% of total]		trusted [% of total]	native [% of total]
en	564	495 (87.8%)	168 [29.8%]	664	568 (85.5%)	128 [19.3%]	442	–	78 [17.6%]
ar	693	543 (78.4%)	432 [62.3%]	559	463 (82.8%)	117 [20.9%]	465	–	0 [0.0%]
es	581	581 (100.0%)	542 [93.3%]	428	416 (97.2%)	86 [20.1%]	421	–	0 [0.0%]
fr	463	409 (88.3%)	178 [38.4%]	416	404 (97.1%)	0 [0.0%]	495	–	18 [3.6%]
hi	580*	505* (87.1%)	496* [85.5%]	1013	531 (52.4%)	477 [47.1%]	723	–	314 [43.5%]
ja	386	356 (92.2%)	60 [15.5%]	472	448 (94.9%)	0 [0.0%]	447	–	0 [0.0%]
ko	642*	603* (93.9%)	66* [10.3%]	583	523 (89.7%)	0 [0.0%]	408	–	0 [0.0%]
ru	657	555 (84.5%)	96 [14.6%]	370	370 (100.0%)	0 [0.0%]	504	–	0 [0.0%]
tl	437	428 (97.9%)	91 [20.8%]	344	344 (100.0%)	0 [0.0%]	371	–	36 [9.7%]
zh	575	481 (83.6%)	354 [61.6%]	462	429 (92.9%)	0 [0.0%]	476	–	0 [0.0%]

* marked results are obtained using the LOC+GOLD^{bw}+PRI data quality control setting.

A.4. Evaluation Time

The evaluation period (given in days), the total grading time (given in hours, “(hh:mm:ss)”), and the average time per HIT (given in seconds, “[mm:ss]”) of the trusted gradings obtained for each evaluation task.

TRG	Data Quality Control Mechanism											
	EXPERT			LOC+GOLD ^{bw}			GOLD ^b			NONE		
	evaluation period	(grading time)	[avg. time per assignment]	evaluation period	(grading time)	[avg. time per assignment]	evaluation period	(grading time)	[avg. time per assignment]	evaluation period	(grading time)	[avg. time per assignment]
en	6.9 days	(06:41:09)	[00:13]	4.8 days	(04:30:13)	[00:39]	0.9 days	(03:24:45)	[00:25]	0.4 days	(01:12:32)	[00:17]
ar	–			4.7 days	(06:29:32)	[00:47]	0.7 days	(02:48:34)	[00:24]	0.1 days	(00:45:50)	[00:07]
es	–			2.2 days	(06:06:55)	[00:47]	0.3 days	(01:49:34)	[00:16]	0.1 days	(00:48:22)	[00:07]
fr	–			3.9 days	(04:19:36)	[00:40]	0.2 days	(03:13:52)	[00:29]	0.2 days	(00:55:04)	[00:14]
hi	–			1.2 days*	(03:27:34)*	[00:35]*	0.2 days	(02:44:42)	[00:19]	0.1 days	(00:52:55)	[00:08]
ja	1.1 days	(05:48:35)	[01:07]	12.8 days	(02:22:28)	[00:27]	0.7 days	(01:39:58)	[00:14]	0.1 days	(01:07:17)	[00:10]
ko	7.1 days	(11:29:41)	[00:16]	88.9 days*	(04:45:05)*	[00:41]*	3.1 days	(01:10:46)	[00:10]	0.1 days	(01:07:52)	[00:11]
ru	–			17.0 days	(06:48:44)	[00:52]	0.1 days	(01:55:05)	[00:18]	0.2 days	(01:12:47)	[00:11]
tl	–			2.1 days	(03:03:16)	[00:26]	0.1 days	(00:43:59)	[00:07]	0.1 days	(01:07:17)	[00:10]
zh	1.1 days	(07:32:56)	[01:26]	23.7 days	(05:09:30)	[00:43]	2.1 days	(01:29:36)	[00:13]	0.1 days	(01:52:16)	[00:16]

* marked results are obtained using the *LOC+GOLD^{bw}+PRI* data quality control setting.

A.5. Ranking Results (%_{better})

The subjective evaluation of translation quality of 4 MT engines trained on different training data sizes (160k, 80k, 20k, 10k). The *Ranking* scores were obtained as the average number of times that a system was judged better than any other system.

TRG	Data Quality Control Mechanism															
	EXPERT				LOC+GOLD ^{bw}				GOLD ^b				NONE			
	160k	80k	20k	10k	160k	80k	20k	10k	160k	80k	20k	10k	160k	80k	20k	10k
en	0.5245	0.4755	0.3272	0.1453	0.4766	0.3481	0.2343	0.1138	0.2853	0.2620	0.1673	0.0750	0.1605	0.1714	0.1020	0.0680
ar	–	–	–	–	0.4319	0.3038	0.1943	0.1497	0.1816	0.1135	0.0837	0.0723	0.0008	0.0009	0.0019	0.0081
es	–	–	–	–	0.4899	0.4062	0.2342	0.1176	0.1983	0.1474	0.0758	0.0620	0.0000	0.0000	0.0000	0.0000
fr	–	–	–	–	0.4823	0.4020	0.1652	0.0908	0.1929	0.1631	0.0879	0.1035	0.0400	0.0326	0.0370	0.0370
hi	–	–	–	–	0.2837*	0.2068*	0.1094*	0.0889*	0.1872	0.1587	0.0868	0.0947	0.0201	0.0111	0.0191	0.0040
ja	0.5735	0.4803	0.2528	0.1027	0.4811	0.3695	0.1461	0.0755	0.2355	0.1639	0.1281	0.0675	0.0724	0.0678	0.0470	0.0165
ko	0.4690	0.3746	0.2625	0.1136	0.3809*	0.3185*	0.1740*	0.0919*	0.0862	0.0689	0.0532	0.0517	0.0000	0.0000	0.0000	0.0000
ru	–	–	–	–	0.3459	0.2957	0.1830	0.1078	0.2588	0.2390	0.1887	0.1613	0.0606	0.0552	0.0433	0.0400
tl	–	–	–	–	0.3914	0.2679	0.1428	0.1027	0.0679	0.0648	0.0340	0.0340	0.0022	0.0011	0.0022	0.0044
zh	0.5482	0.4313	0.3318	0.2133	0.6367	0.5128	0.4110	0.2811	0.1371	0.1331	0.1223	0.1035	0.0802	0.0552	0.0542	0.0427

* marked results are obtained using the *LOC+GOLD^{bw}+PRI* data quality control setting.

A.6. Ranking Agreement

Fleiss’ kappa correlation coefficient comparing the obtained crowd-based evaluation results to the oracle and expert judgments for each translation task. The κ scores are interpreted in (Landis and Koch, 1977) as follows:

$\kappa < 0$: “none” $\kappa \leq 0.6$: “moderate”
 $\kappa \leq 0.2$: “slight” $\kappa \leq 0.8$: “substantial”
 $\kappa \leq 0.4$: “fair” $\kappa \leq 1.0$: “almost perfect”

Worker vs. Oracle/Expert Agreement

TRG	Data Quality Control Mechanism					
	LOC+GOLD ^{bw}		GOLD ^b		NONE	
	oracle	expert	oracle	expert	oracle	expert
en	0.45	0.62	0.19	0.30	0.39	0.43
ar	0.22	–	0.09	–	0.11	–
es	0.35	–	0.08	–	1.00	–
fr	0.26	–	0.04	–	0.53	–
hi	0.05*	–	0.00	–	-0.02	–
ja	0.38	0.66	0.10	0.22	0.01	0.23
ko	0.56	0.50	0.79	0.14	-0.01	0.17
ru	0.32	–	0.08	–	0.15	–
tl	0.21	–	0.04	–	-0.02	–
zh	0.62	0.56	0.07	0.09	0.17	0.20

* marked results are obtained using the *LOC+GOLD^{bw}+PRI* data quality control setting.

Readability and Translatability Judgments for ‘Controlled Japanese’

Anthony Hartley

Toyohashi University of Technology

a.hartley@imc.tut.ac.jp

Midori Tatsumi

Toyohashi University of Technology

midori.tatsumi2@mail.dcu.ie

Hitoshi Isahara

Toyohashi U of Technology

isahara@tut.jp

Kyo Kageura

University of Tokyo

kyo@p.u-tokyo.ac.jp

Rei Miyata

University of Tokyo

rei@p.u-tokyo.ac.jp

Abstract

We report on an experiment to test the efficacy of ‘controlled language’ authoring of technical documents in Japanese, with respect both to the readability of the Japanese source and the quality of the English machine-translated output. Using four MT systems, we tested two sets of writing rules designed for two document types written by authors with contrasting professional profiles. We elicited judgments from native speakers to establish the positive or negative impact of each rule on readability and translation quality.

1 Introduction

It is widely acknowledged that the typological ‘distance’ between Japanese and English (the most common European target language for MT from Japanese) hampers the achievement of high-quality translation. We seek to address this challenge by investigating the feasibility of developing a ‘controlled Japanese’ with explicit restrictions on vocabulary, syntax and style adequate for authoring technical documentation.

Our starting point is sentences extracted from two types of document: consumer user manuals (UM) and company-internal documents articulating the know-how of key employees (KH). UM are produced by professional technical authors, while KH are written as ‘one-offs’ by the employees themselves, capturing their own know-how. Thus, there is a sharp difference in the ef-

fort the two groups of writers can be expected to invest and the linguistic knowledge they bring to a controlled authoring task.

In outline, our experiment entailed formulating a set of writing rules (‘authoring guidelines’) for each document type. Sentences violating the rules were extracted from the original data and rewritten (‘pre-edited’ in this experimental setting) in accordance with the respective rule. The original and rewritten sentences were then translated by different MT systems; finally, the inputs and outputs were submitted to human evaluation.

Since the readers of the original Japanese and the readers of the translated English are equally important, we devised protocols to assess what we termed the ‘readability’ of the Japanese source sentences and their ‘translatability’ as gauged by the perceived quality of the English target sentences.

In interpreting the results, we try to identify the most promising avenues for further development.

2 Controlled Language and MT

The general principles of controlled language (CL) and the challenges posed by its deployment are clearly summarised by (Kittredge, 2003; Nyberg et al., 2003). Evidence of the effectiveness of CL in cutting translation costs has been in the public domain for some 30 years, from (Pym, 1990) in the automotive domain to (Roturier, 2009) in the software domain.

More specific studies have been undertaken to identify those rules which have the greatest impact on the usability of MT output (e.g., O’Brien and Roturier, 2007).

Overwhelmingly, controlled language studies have focused on English as source language. This is not to say that CL varieties do not exist for languages other than English. Among recent work, Barthe (1998) relates the process of developing GIFAS, the ‘rationalised’ French counterpart of the AECMA documentation standard for the aerospace industry, while Lieske et al. (2002) describe a controlled German.

In the case of Japanese, the application of the CL notion dates back to (Nagao and Tanaka, 1984), who describe a framework for assisting authors in producing what they termed ‘machine-readable’ Japanese. Yoshida (1987) outlines a framework for designing a ‘standardised’ Japanese for MT. Kaji (1999) offers a few Japanese examples.

More recent computational work has focused on automatic re-writing of what we can term ‘MT-intractable’ Japanese (e.g., Shirai, 1998; Matsuyoshi et al., 2004). Since such re-writing is a machine-internal process, these studies are not necessarily directly applicable to guiding the authoring of human-readable texts.

Morita and Ishida (2011) provide protocols to enable monolingual users to converge on a correct Japanese/English machine translation, but no a priori writing or editing rules are proposed.

The proposals in (Sato et al., 2003) are motivated by personal rather than technical communication. Matsui and Magnusson (2011) require language learners using online Japanese-to-English MT to apply six ‘revision’ rules to their input, including insertion of pronominal subjects (Japanese is a pro-drop language) and of the determiner *その* before nouns. However, to generalise such insertions is unnatural and potentially misleading for human readers.

Finally, the rules proposed in (Ogura et al., 2010) are intended for technical writers, but no empirical evidence of their efficacy is presented.

3 Formulation of Authoring Guidelines

As we noted, we are dealing in this case with authoring in two very different settings, distinguished by the professional background of the authors themselves, the purpose of the documents they write and the characteristics of their readerships. Accordingly, we adopted different rationales for selecting what can be formally described as *rules*, which are presented to the writers as *guidelines*. Nyberg et al. (2003) identify prior writing expertise as a key factor in the successful deployment of CL.

3.1 Settings and selection process

In the case of UM, we are dealing with professional authors producing instructions for consumer-users whose perception of the appliances will depend in part on the quality of the documentation. As for KH, the authors have no prior training in technical documentation. Their task is to write down the conceptual and procedural know-how underlying their own job in order to share it with other staff both in Japan and in overseas operations. Their readers are ‘insiders’ with experience of the corporate culture and can be expected to tolerate some infelicity of expression provided the content is understandable.

The purpose and motivation in selecting the guidelines differ somewhat between the two settings. While the training of the UM authors allows some guidelines that require sophisticated linguistic knowledge, the guidelines for the KH setting need to prioritise ease of implementation by non-professional writers unaccustomed to writing with translation in mind. The trade-off for this gentle learning curve is incomplete coverage of problematic linguistic features by the guidelines.

The UM guidelines were developed through a combination of bottom-up and top-down approaches. From a corpus of 38,527 Japanese-English translation units we selected all Japanese segments of length greater than 150 bytes. We translated the resulting 10,026 segments with Google Translate¹ and Systran 7 Premium Translator². Given that the data was judged to be typical of user manuals in Japanese, we emphasised improving MT quality. We manually identified segments with flagrant translation errors induced by structural features of the source text. This search was guided by the categories identified in (Ogura et al., 2010). The outcome was a set of 20 problem features, described in section 3.2.

For the KH guidelines, we proceeded top-down. Our corpus consisted of three documents comprising 33, 20, and 53 pages, or 177,742, 10,433, and 32,366 characters respectively. Unlike the UM corpus, the KH showed little homogeneity in wording and style. General technical and business writing guidebooks³ provided suggestions for some of the guidelines we formulated. Others were chosen to remedy known problems of Japa-

¹ <http://translate.google.com/>

² <http://systransoft.com/>

³ 日本語スタイルガイド 第2版 (一般財団法人テクニカルコミュニケーション協会編著), 説得できる文章・表現 200の鉄則 (日経BP社出版局)

nese to English MT. An initial set of some 40 candidate guidelines was filtered according to two criteria. First, some of the problem features occurred either not at all in the corpus or with very low frequency. Second, some guidelines were judged to require meta-linguistic knowledge which could not safely be assumed on the part of non-professional writers or imparted in a necessarily brief training session. The outcome was a set of 10 guidelines, described in section 3.3.

3.2 Authoring guidelines: UM

Table 1 lists the 20 problem features from the UM corpus which we experimented with. These gave rise to 28 pre-editing rules formulated as ‘Omit ...’, ‘Replace with ...’ or ‘Add ...’.

F1	Long sentences (> 50 characters)
F2	Sentences of 3 or more clauses
F3	Negative expressions
F4	Verb + nominaliser こと
F5	Nominaliser もの
F6	Verb + ように (‘it is suggested that’)
F7	Topicalizing particle は
F8	Coordinating conjunction または (‘or’)
F9	Modal れる・られる (‘can’)
F10	Verb 見える (‘can be seen’)
F11	Compound noun strings
F12	Particle など (‘and so on’)
F13	Single use of conjunction たり (‘either’)
F14	Katakana verbs
F15	Suffix 感 (‘sense of’)
F16	Verb かかる (‘start’)
F17	Verb 成る (‘become’)
F18	Verb 行う (‘perform’)
F19	Case-marking particle で (‘with’, ‘by’)
F20	Verb ある・あります (‘exist’)

Table1. ‘Avoid’ features of UM guidelines

Table 2 shows examples of (a) original and (b) re-written sentences for three of the features.

F7	1a	ソングは「メロディー」と自動伴奏の組み合わせでできています
	1b	「メロディー」と自動伴奏の組み合わせでソングができています
F9	2a	1つのウェブフォームに割り当てられるキーバンクは最大128個までです
	2b	1つのウェブフォームに割り当てる

		ことができるキーバンクは最大128個までです
F20	3a	コードの詳細は64ページにあります
	3b	コードの詳細は64ページに記載されています

Table 2. Pre-edited UM sentences

3.3 Authoring guidelines: KH

These guidelines fall into three categories: notation (*a, b, c, d*); word/phrase structure (*e, f*); sentence structure (*g, h, i, j*).

a. Do not use single-byte Katakana characters

Katakana, used mainly for writing foreign words, is the only one of the three Japanese scripts that can also be written in single byte. Single-byte Katakana writes a voiced consonant with an unvoiced base character followed by a diacritic (underlined) that indicates voicing:

プロシ^ャェク^タ・スクリー^ン・ケー^フル

This can perturb tokenisation by MT systems.

b. Do not use symbols in sentences

MT systems can fail to identify the terms of the relationship (underlined) represented by symbols such as a minus sign signalling ‘the difference between A and B’.

実際の投入工数 - 基準時間との比較による能率管理

c. Do not use nakaguro (bullet) as a delimiter

MT systems can fail to distinguish parallel items delimited by nakaguro (underlined> from the surrounding text.

会社のステージ・業績に応じた賃金、賞与の水準

d. Avoid using inappropriate Kanji characters

This equates to spelling mistakes in English.

e. Avoid creating long noun strings

Nouns and stems of adjectives, adverbs, and verbs can be combined to form a compound noun.

f. Do not use ‘perform’ to create a *sa*-verb

Sa-verbs formed by adding a ‘do’ verb to a noun are widely used. Adding ‘perform’ or ‘execute’ (行う/実行する) instead of the simple する creates verbose texts and awkward output.

g. Avoid topicalisation

Japanese is ‘topic-prominent’, i.e., the topic is often given the particle は, which makes it look like the subject of the sentence, even if it is not.

- h. Do not connect sentences to make a long sentence
- i. Do not interrupt a sentence with a bulleted list

The first line of the example reads ‘When setting the standard unit price,’ and the last ‘must be specified.’ Kohl (2008) recommends combining these into a complete sentence, such as ‘The following items must be specified when setting ...’

基準単価設定は

- ・セット品の単価 (品番 : S1)
 - ・単品部品の単価 (品番 : A、B)
- の設定をしなければならない

- j. Avoid listing numerous parallel items in a sentence; use a bulleted list instead

4 Experimental Set-Up

For both the UM and KH settings our aim was to assess, using human judges, any gain or loss in (a) the readability of the Japanese sentences after pre-editing, and (b) their translatability as gauged by the perceived quality of the English translations produced by MT. Table 3 shows the size of the data sets and the numbers of judges.

	UM	KH
Rules tested	20	10
Sentences per rule	5	6
JAO sentences	100	60
JAR sentences	100	60
JA judges	31	20
MT systems	3	2
ENO sentences	300	120
ENR sentences	300	120
Human reference	100	0
EN judges	28	8

Table 3. Data sets and judges

4.1 Japanese test corpus selection

In the case of UM, we selected five sentences for each of the 20 features of Table 1 by randomly ordering our 38,527 segments and choosing the first five sentences to satisfy the condition and for which the human reference translation contained no added information. With KH, for each guideline described in Section 3.3 we selected six sentences that violated it. In both cases, when a sentence contained more than one type of problem, we changed only the part that was subject to the applicable guideline. Thus, we had sets of original sentences UM-JAO and KH-JAO and re-written sentences UM-JAR and KH-JAR.

4.2 English test corpus generation

Using two online MT systems, namely, Excite⁴ and Google Translate, we translated all the JAO and JAR sentences into English (ENO and ENR, respectively). We used them ‘off the shelf’ via the internet, with no user dictionary or any sort of customisation. The UM Japanese sentences were also translated using Systran 7 Premium, with the benefit of user dictionaries extracted from the available English human (reference) translation of the manual.

Note that it was not the MT systems themselves that were the focus of our evaluation, but the writing guidelines. We wanted to see whether rules had a positive impact irrespective of system.

4.3 Judges

Ideally, the quality of a text should be evaluated by readers who are similar in profile to the actual target readership. Since the UM data related to consumer electronic audio and music equipment, we recruited as plausible readers Japanese native-speaker university students and English native-speaker graduate students with no knowledge of Japanese (which might allow them to compensate for mis-translations).

In the case of the KH judges, both the Japanese native-speakers and the English native-speakers were recruited from within the company. They were, therefore, ideally suited to the task.

4.4 Questionnaire design

The quality of the Japanese source text written according to the guidelines must be as good as or better than that of the text written without guidelines. With the UM-JA data, we presented the judges with pairs of sentences A and B in which the ordering of JAO and JAR was randomised. They had five options: A much more readable than B; A more readable than B; A and B equally readable; B more readable than A; B much more readable than A.

However, such a pairwise comparison does not tell whether one or both are acceptable or not. In order to overcome this shortcoming, with KH-JA the judges were again shown a pair of ‘before’ and ‘after’ sentences at a time; but were asked to evaluate each of them on the four-point scale in Figure 1 (English gloss of the questions, which were written in Japanese).

⁴ <http://www.excite.co.jp/world/> Note that Excite changed their MT engine (from Fujitsu to Toshiba) between the time of the UM experiment and that of the KH experiment.

The following two sentences convey the same content but are written using different words. Please evaluate the readability of each sentence.

A 欠勤・早退・遅刻・離業など、業務に従事していないときの賃金は、原則として支払いません。

B 欠勤・早退・遅刻・離業など、業務に従事していないときは、原則として賃金を支払いません。

How readable is A? Tick the closest option:

Easy Fairly easy Fairly difficult Difficult

How readable is B? Tick the closest option:

Easy Fairly easy Fairly difficult Difficult

Figure 1. Question to judges of KH-JA

We surmised that showing two sentences at a time would lead the judges to focus on readability in terms of expression rather than content. Therefore, we used the word ‘readable’ (読みやすい) rather than ‘understandable’ (わかりやすい), to avoid the results being affected by any gaps in the judges’ content knowledge. Moreover, although the judges were not explicitly asked to compare the two and decide which was better, we thought that, if they perceived a difference in readability between the two texts, they might differentiate between them in their judgment.

In evaluating the English translations we adopted the same approach for KH-EN as for UM-JA, that is, judges were asked to say whether they thought sentence A more readable than B, B more readable than A, or A and B equally readable. This decision was dictated by the small number of judges available (eight).

For UM-EN we were able to employ more judges. Given that we were dealing with (mostly ill-formed) MT output, we preferred to elicit judgments of sentences independently rather than pairwise, in case judges were more ‘forgiving’ of a better or less bad member of a pair. Thus, judges were shown a single sentence at a time and asked the question in Figure 2.

In addition, setting can be cancelled even by the fact that another song is chosen.

How well did you understand the sentence?

Fully Mostly Partly Not at all

Figure 2. Question to judges of UM-EN

4.5 Administration of questionnaires

The questionnaires were posted online and each judge was given a unique password. Only one question was presented at a time and the judges

were asked not to return to a question after ticking their preference. Unanswered questions were flagged. The judges clicked the submit button once the whole questionnaire was completed.

With UM-JA, each of the 31 judges saw all 100 sentence pairs, with a 10-minute break in the middle. The ordering of presentation of the pairs was randomised and the ordering of JAO and JAR within the pair was equally distributed. With KH-JA, each of the 20 judges saw 30 of the 60 sentence pairs (again, randomly presented, such that each question set was unique), which yielded 10 judgments for each pair.

With UM-EN, as Table 3 shows, we had 700 sentences, including the human reference, which did not always receive the best score (see Section 5.3). Each of the 28 judges saw 100 sentences in random order, with a 10-minute break in the middle. No judge saw two translated versions of the same JAO source sentence or translations of both members of a JAO/JAR pair. Again, the presentation order was randomised.

Unfortunately with KH-EN, only eight English native-speakers were available, so, in order to obtain four judgments for each pair of sentences ENO/ENR-Excite and ENO/ENR-Google, each judge saw 60 pairs. Again, the ordering of ENO and ENR was randomised, each judge saw (in random order) an equal number of outputs from each MT system, and no judge saw translations of the same JAO/JAR pair by both systems.

5 Results and Interpretation

5.1 Japanese readability by guideline: UM

Figure 3 shows, in percentages, the judgments of improvement or deterioration caused by each of the 20 guidelines. The labeling of a rewritten text as ‘Better’, ‘Same’ or ‘Worse’ than the original derives directly from the relative rating given by the judges.

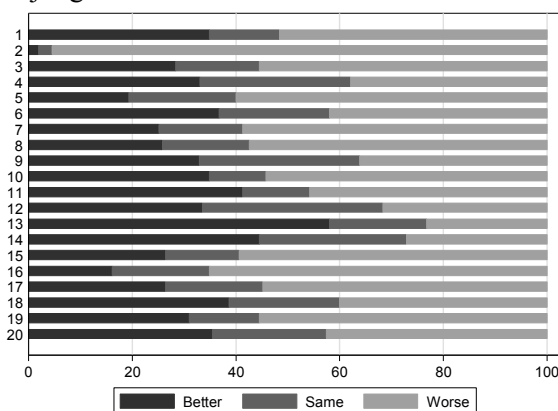
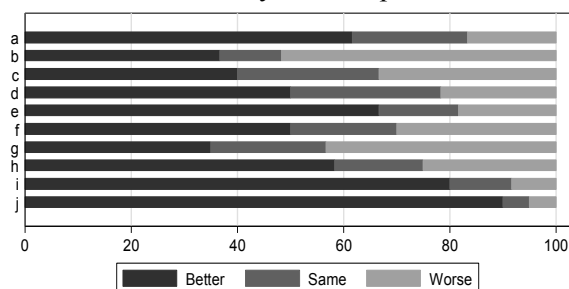


Figure 3. UM Japanese readability gains

Most rules were judged to make the Japanese less readable, (2, 3, 5, 7, 10, 16, 17) having a particularly severe effect. The exceptions were 13 and 14, although neither of these improves translatability (See Figure 5).

5.2 Japanese readability by guideline: KH

The questionnaire design (see Section 4.4) enables us to draw conclusions on both the relative and absolute readability of the Japanese text.



In relative terms, Figure 4 shows that most of the guidelines achieved the objective of improving or at least maintaining the quality of the text, in so far as they were valued as Better or Same by at least two thirds of the judges.

The exceptions were *b* (Do not use symbols in sentences) and *g* (Avoid topicalisation). Guideline *c* (Do not use nakaguro (bullet) as a delimiter) also received a rather low evaluation. These results for *b* and *c* suggest that the use of non-linguistic devices to relate meaningful parts of a sentence promotes concision. The result for *g* was somewhat expected, since topicalisation does not usually compromise readability for humans and editing sentences to eliminate topicalisation can result in wordiness.

The greatest positive impact on readability was registered by guidelines *i* (Do not interrupt the introductory sentence before bulleted lists) and *j* (Avoid listing parallel items in a sentence). While the use of bulleted list is regularly recommended in a number of writing guides, ‘avoiding interrupted sentences’ does not usually make a topic for guides targeting Japanese. Talking about English, Kohl (2008) recommends this practice to help MT systems, but regards it as ‘low priority’ for human translators and non-native readers. In the case of Japanese, our experiment shows that it also helps human readers.

To ground the absolute readability of the text, we converted the rating options to numbers as follows: ‘Easy to read’ = 4, ‘Fairly easy’ = 3, ‘Fairly difficult’ = 2, ‘Difficult’ = 1. Table 4

compares the median values of the evaluation results for JAO and JAR, and ENO and ENR.

	JAO	JAR	EXC	GOO
a	3	4	0	1
b	3	3	1	-4
c	3	3	0	-1
d	2.5	3	1	1
e	3	4	-3	3
f	3	3	3	-1
g	3	3	-1	3
h	3	3	-5	2
i	2	4	5	2
j	2	4	2	-2

Table 4. KH readability and translatability

The table highlights several results. First, overall readability for both JAO and JAR is rather good; there is no category whose median value is lower than 2. This is not surprising, however, since all sentences have been written by a human.

Second, there are no categories for which JAO received a lower score. This suggests that the set of guidelines we used for this experiment was generally successful in maintaining and even raising the quality of Japanese sentences.

Third, among the categories for which JAR received higher scores than JAO, namely, *a*, *d*, *e*, *i*, and *j*, the levels of improvement vary. While there is only a 0.5 point increase for *d*, there is 2 point increase for *i* and *j*, which demonstrates that these two guidelines have the highest impact on improving the readability of the Japanese text.

5.3 Translatability into English: UM

Table 5 gives examples of improvements induced by the guidelines (for input see Table 2).

F7 GOO	1a	Song is “melody” is made with a combination of auto-accompaniment.
	1b	Melody is a song made by a combination of automatic accompaniment.
F9 EXC	2a	They are a maximum of 128 key banks assigned to one wave form.
	2b	They are a maximum of 128 key banks which can be assigned to one wave form.
F20 SYS	3a	As for details of the cord/code there is page 64.
	3b	Details of the cord/code are stated in page 64.

Table 5. UM output: ENO and ENR sentences

Figure 5 shows that the rules for features 2 (three or more clauses), 7 (topicalisation), 18 (‘perform’) and 20 (‘exist’) are highly effective. Their ‘Better’ to ‘Worse’ ratio is greater than 3:1.

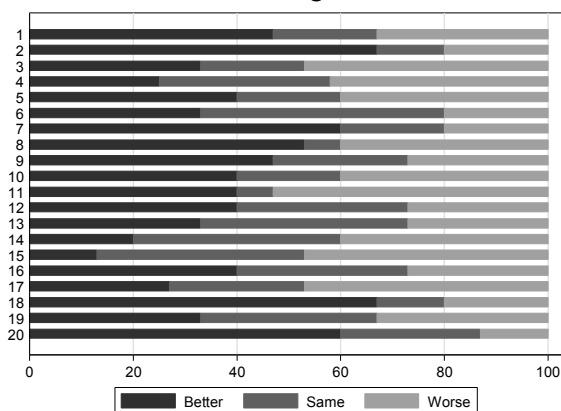


Figure 5. UM translatability gains

We interpreted responses to the question shown in Figure 2 by considering ‘fully’ or ‘mostly’ understandable as ‘acceptable’ and ‘partly’ or ‘not at all’ understandable as ‘inacceptable’. Figure 6 corroborates the impact of rules (2, 7, 18, 20). Taking the 47% acceptability of 2 and 7 as a lower threshold, we can add rules 4 and 12 to the set promoting translatability.

There is, however, no intersection between those rules that boost translatability and those that enhance readability. Indeed, 2 and 7 inhibit readability considerably.

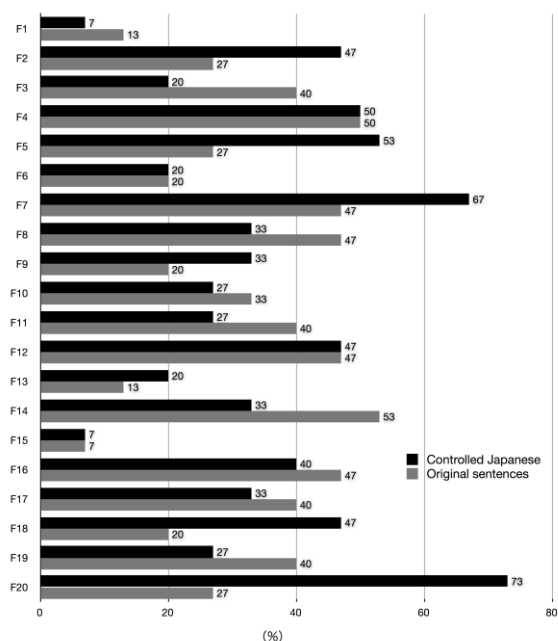


Figure 6. UM translation acceptability

The human reference was judged worse than both ENO and ENR MT outputs in 2% of cases, and no better in 10% of cases (median values). However, the cases varied between ENO and ENR, as Section 5.4 shows.

5.4 Translatability into English: KH

Since the impact of the rules on translation quality diverged markedly between Excite (RBMT) and Google Translate (SMT), we present them separately, in Figures 7 and 8.

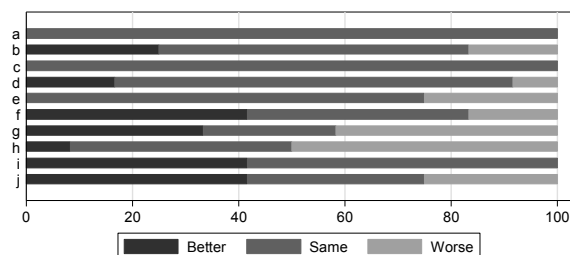


Figure 7. KH translatability gains – Excite

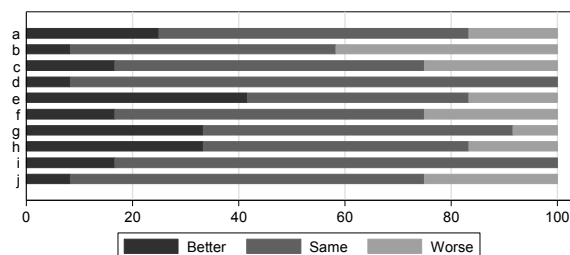


Figure 8. KH translatability gains – Google

Some differences are easy to explain: Excite handles single-byte Katakana while Google does not (rule *a*); scrutiny shows Excite to better handle long sentences (rule *h*), whose naturalness may then be impaired by unnecessary splitting.

Considering the relation between improved readability and improved translation quality, the last two columns of Table 4 give the net sum of the judgments comparing ENO/ENR-Excite and ENO/ENR-Google, respectively, in the range +12 to -12. Although there is no statistically significant correlation, it appears that Google Translate may track readability somewhat more closely than does Excite. Only with rules *d* and *i* do all three indicators improve.

In the case of rules that maintain readability without improving it, the effects are noticeably contrasting: rules *b* and *f* boost Excite but depress Google, while rules *g* and *h* have precisely the reverse effect.

Note that these are relative changes in the performance of the same system given modified inputs. We did not set out to compare MT sys-

tems. Limitations on the availability of competent judges prevented us from trying to ground the judgments in terms of the acceptability of the sentences, as we did with the UM-EN data.

6 Conclusions

We developed two sets of writing rules for use in two contrasting settings. The simple rules applied to the KH texts written by non-professional authors consistently maintained or improved readability, arguably from a relatively low baseline. This may motivate future writers to use them, even if only two rules also raise MT quality. The negative impact on readability of the great majority of UM rules may be due to their departure from de facto technical writing standards for Japanese already judged ‘good’. However, four UM rules boosted translatability to a point where post-editing costs might be considerably reduced. This is a trade-off to explore further. The intersections of the sets ($1,2\approx h$, $7\approx g$, $11\approx e$, $18\approx f$), show little common promise.

Although the MT systems as such were not under investigation, the results overall suggest that their ‘reactions’ are quite idiosyncratic, even if Excite and Systran (both RBMT) behave similarly to each other (and differently to Google Translate). This suggests in turn the need to mutually tune MT system and writing guidelines. The obvious path is to create an authoring environment fully integrated with the MT resources.

Our future work will adopt a functional rather than surface-syntactic perspective on the goal of creating translation-ready documents.

Acknowledgments

This work was supported by the Strategic Information and Communication R&D Promotion Programme of the Ministry of Internal Affairs and Communications, Japan, and Japan Society for the Promotion of Science grant (A) 21240021.

References

- Kohl, John R. 2008. *The Global English Style Guide*. SAS Institute Inc, Cary, NC.
- Barthe, Kathy. 1998. GIFAS Rationalised French: Designing One Controlled Language to Match Another. *2nd International Workshop on Controlled Language Applications*. Pittsburgh, PA. 87-102.
- Kaji, Hiroyuki. 1999. Controlled Languages for Machine Translation: State of the art. *Machine Translation Summit VII*. Singapore. 37-39.
- Kittredge, Richard. 2003. Sublanguages and Controlled Language. R. Mitkov (ed.). *The Oxford Handbook of Computational Linguistics*. OUP, Oxford, UK. 430-447.
- Lieske, Christian, Christine Thielen and Melanie Wells. 2002. Controlled Authoring at SAP. *Translating and the Computer 22*. Aslib, London, UK.
- Matsui, June-ko and David Magnusson. 2011. Six Pre-edit Techniques for Enhancing Japanese to English Machine Translations. *Interpreting and Translation Studies*, 11:173-184.
- Matsuyoshi, Suguru, Satoshi Sato and Takehito Utsuro. 2004. Paraphrasing a Functional Word ‘nara’ for Machine Translation. *Information Processing Society of Japan*, NL-159:201-208.
- Morita, Daisuke and Toru Ishida. 2011. Collaborative Translation Protocols. T. Ishida (ed.). *The Language Grid*. Springer, New York, NY. 215-230.
- Nagao, Makoto and Nobuyoshi Tanaka. 1984. Support System for Writing Texts Based on Controlled Grammar. *Information Processing Society of Japan*, NL-44:33-40.
- Nyberg, Eric, Teruko Mitamura and Willem-Olaf Huijsen. 2003. Controlled Language for Authoring and Translation. H. Somers (ed.). *Computers and Translation*. Benjamins, Amsterdam, NL. 245-281.
- O’Brien, Sharon and Johann Roturier. 2007. How Portable are Controlled Language Rules? *MT Summit*, Copenhagen, DK.
- Ogura, Hidesato, Mayo Kudo and Hideo Yanagi. 2010. Simplified Technical Japanese: Writing Translation-Ready Japanese Documents. *Information Processing Society of Japan*, DD-5:1-8.
- Pym, Peter. 1990. Pre-Editing and the Use of Simplified Writing for MT. *Translating and the Computer 10*. Aslib, London, UK.
- Roturier, Johann. 2009. Controlled Language for MT in Action. *Translingual Europe*, Prague, CZ.
- Sato, Satoshi, Masatoshi Tsuchiya, Masahiro Asaoka, Masahiro Asaoka and Qingqing Wang. 2003. Standardizing Japanese Sentences. *Information Processing Society of Japan*, NL-4:133-140.
- Shirai, Satoshi, Satoru Ikehara, Akio Yokoo and Yoshifumi Ooyama. 1998. Automatic Rewriting Method for Internal Expressions in Japanese to English MT and its Effects. *2nd International Workshop on Controlled Language Applications*. Pittsburgh, PA. 62-75.
- Yoshida, Sho. 1987. Standardizing Japanese and Design of Controlled Japanese. Organizing Committee of 1st University Science Public Symposium (ed.). *Characteristics of Japanese and Machine Translation*. Tokyo, Japan. 132-142.

A Phrase Table without Phrases: Rank Encoding for Better Phrase Table Compression

Marcin Junczys-Dowmunt

Faculty of Mathematics and Computer Science

Adam Mickiewicz University, Poznań, Poland

junczys@amu.edu.pl

Abstract

This paper describes the first steps towards a minimum-size phrase table implementation to be used for phrase-based statistical machine translation. The focus lies on the size reduction of target language data in a phrase table. Rank Encoding (R-Enc), a novel method for the compression of word-aligned target language in phrase tables is presented. Combined with Huffman coding a relative size reduction of 56 percent for target phrase words and alignment data is achieved when compared to bare Huffman coding without R-Enc. In the context of the complete phrase table the size reduction is 22 percent.

1 Introduction

As the size of available parallel corpora increases, the size of translation phrase tables used for statistical machine translation extracted from these corpora increases even faster. Although phrase table filtering methods (Johnson et al., 2007) have been described and physical memory as well as disk space are cheap, even current high-end systems can be pushed to their limits. The current in-memory representation of a phrase-table in Moses (Koehn et al., 2007), a widely used open-source statistical machine toolkit, is unusable for anything else but toy-size translation models or prefiltered test set data. A binary on-disk implementation of a phrase table is generally used, but its on-disk size requirements are significant.¹

The goal of this paper is to describe the first steps towards a compact phrase table implementa-

tion that can be used as a drop-in replacement for both, the binary phrase table implementation and the in-memory phrase table available in Moses. An important requirement is the faithful production of translations identical to translations generated from the original phrase table implementation if the same settings are provided.

The general idea is to trade in processor time for disk and memory space. Instead of keeping fully constructed target phrases in the phrase table, they are stored as Huffman compressed sequences of bytes. On demand, they are decompressed, decoded, and constructed as objects during run-time. As we show later, this does not necessarily mean that performance is negatively affected.

Even better compression can be achieved with a dedicated encoding method of target words developed for translation phrase tables. Rank Encoding (R-Enc) exploits the fact that target phrase words can be reduced to abstract symbols that describe properties of source phrase words rather than target words. The statistical distribution of these abstract symbols in the phrase table allows for a much better choice of Huffman codes.

2 Related Work

Zens and Ney (2007) describe a phrase table architecture on which the binary phrase table of Moses is based. The source phrase index consists of a prefix tree. Memory requirements are low due to on-demand loading. Disk space requirements however are substantial.

Promising alternatives to the concept of fixed phrase tables are suffix-array based implementation of phrase tables (Callison-burch and Bannard, 2005; Zhang and Vogel, 2005; Lopez, 2008; Levenberg et al., 2010) that can create phrase pairs on-demand more or less directly from a parallel corpus. However, we do not compare this approach with ours, as we are not concerned with on-demand phrase table creation.

© 2012 European Association for Machine Translation.

¹The need for a more compact phrase-table implementation arose during the author's collaboration with the MT team at WIPO. The space requirements of the binary representations of the phrase table and the reordering table for a single language pair exceeded the space available on a single SSD hard drive.

Other approaches, based on phrase table filtering (Johnson et al., 2007) can be seen as a type of compression. They reduce the number of phrases in the phrase table by significance filtering and thus reduce space usage and improve translation quality at one stroke. An important advantage of this approach is that can be easily combined with any fixed phrase table, including ours.

The architecture of the source phrase index of the discussed phrase table has been inspired by the efforts concerned with language model compression and randomized language models (Talbot and Brants, 2008; Guthrie et al., 2010). Guthrie et. al (2010) who describe a language model implementation based on a minimal perfected hash function and fingerprints generated with a random hash function is the greatest influence. The idea to use the CMPH² library (Belazzougui et al., 2009) and MurmurHash³ for our phrase table implementation originates from that paper.

The problem of parallel text compression has been addressed by only few works (Nevill-Manning and Bell, 1992; Conley and Klein, 2008; Sanchez-Martinez et al., 2012), most other works are earlier variants of Sanchez-Martinez et al. (2012). Conley and Klein (2008) propose to use an encoding scheme based on word alignment and source words. They require the existence of lemmatizers and other knowledge-heavy language related data. Also, compression results are reported without taking into account the additionally needed data. Conley and Klein claim to use phrase pairs for compression, but in our opinion their method is essentially word based, since pointers to all inflected words of a phrase need to be stored. The most recent work in the field is Sanchez-Martinez et al. (2012) who propose to use *generalized biwords* to compress running parallel data. A generalized biword consists of a source word, a sequence of target words aligned with the source word and a corresponding sequence of offsets. Their *Translation Relationship-based Encoder* (TRE) encodes a biword as a pair consisting of a source language word and a position information in a dictionary of generalized biwords. Rank-Encoding, though developed independently, is a combination of the methods presented by Conley and Klein and the TRE introduced by Sanchez-Martinez et al.

²<http://cmph.sourceforge.net/>

³<http://code.google.com/p/smhasher/wiki/MurmurHash3>

Phrase pairs:	3.36×10^8
Distinct source phrases:	2.15×10^8
Distinct target language words:	550,446
Distinct phrase scores:	1.36×10^7
Distinct alignment points:	49
Running source language words:	1.06×10^9
Running target language words:	1.62×10^9
Running phrase scores:	1.68×10^9
Running alignment points:	1.52×10^9
Total running target symbols:	4.84×10^9
Total running symbols:	5.89×10^9

Table 1: Coppa phrase table statistics

3 Experimental Data

The presegmented version of Coppa, the Corpus Of Parallel Patent Applications (Pouliquen and Mazenc, 2011), a parallel English-French corpus of WIPO’s patent applications published between 1990 and 2010, is chosen for phrase table generation. It comprises more than 8.7 million parallel segments with 198.8 million English tokens and 232.3 million French tokens. The Coppa phrase table that is used throughout this paper has been created using the standard training procedure of Moses with included word alignment information. Table 1 gives a set of figures for the phrase table.

The file size of the Moses binary phrase table is given in Table 3 (Section 5.3) along with the space and memory requirements of the variants of our phrase table implementation.

4 Compact phrase table implementation

Figure 1 illustrates the architecture of the discussed phrase table implementation. Its main modules are described in more detail in the following subsections.

4.1 Source Phrase Index

The structure of the source phrase index is inspired by Guthrie et al. (2010) who use a similar implementation for huge n-gram language models. The most important part of the index is a minimal perfect hash function (MPH) that maps a set S of n source phrases to n consecutive integers. This hash function has been generated with the CHD algorithm included in the CMPH library (Belazzougui et al., 2009). The CHD algorithm generates very small MPH (in this case 109 Mbytes) in linear time.

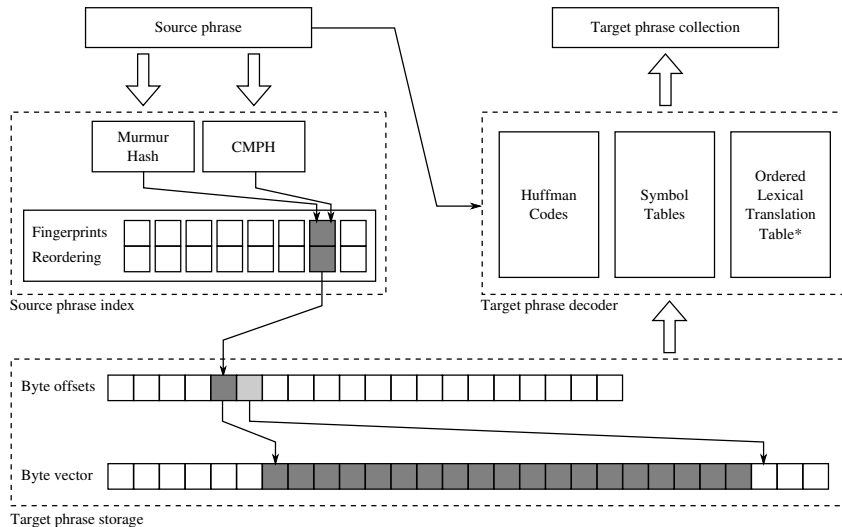


Figure 1: Simplified phrase table implementation schema

The MPH is only guaranteed to map known elements from S to their correct integer identifier. If a source phrase is given that has not been seen in S during the construction of the MPH, a random integer will be assigned to it. This can lead to false assignments of target phrase collections to unseen source phrases. Guthrie et. al (2010) propose to use a random hash algorithm (MurmurHash3) during construction and store its values as fingerprints for each phrase from S . For querying, it suffices to generate the fingerprint for the input phrase and compare it with the fingerprint stored at the position returned by the MPH function. If it matches, the phrase has been seen and can be further processed. For 32 bit fingerprints there is a probability of 2^{-32} of an unseen source phrase slipping through. During our experiments it never happened for such false assignments to surface to a translation.

The MPH generated by the CHD algorithm is not order-preserving, hence the original position of the source phrase in an ordered set S is stored together with each fingerprint. Order-preservation is crucial if any kind of disk IO is involved. In Moses, source phrases are queried by moving the start point of a phrase to each word of a sentence and increasing the phrase length until the length limit or the end of the sentence is reached. Therefore for each start word the querying order is a lexicographical order. In a phrase table representation that preserves the lexicographical source phrase order for corresponding target phrase collections, the results for lexicographically ordered queries will lie close or next to each other. If the data is

stored on disk, this translates directly to physical proximity of the data chunks on the drive and less movement of the magnetic head. Without order-preservation the positions assigned by the MPH are random which can render a memory-mapped version of the phrase-table near unusable.

This phrase table does not contain any representation of source phrases besides the MPH function. Source phrases can be checked for inclusion, but not recovered.

4.2 Target Phrase Storage

The target phrase storage consists of a byte vector that stores target phrase collections consecutively according to the order of their corresponding source phrases. A target phrase collection consists of one or more target phrases, again stored consecutively. A target phrase is a sequence of target word symbols followed by a special stop symbol, a fixed-length sequence of scores, and a sequence of alignment points followed again by a special stop symbol.

Random access capability is added by the byte offset vector. For every target phrase collection, it stores the byte offset at which this collection starts. By inspecting the next offset the end position of a target phrase collection can be determined.

Size reduction is achieved by compressing the symbol sequence of a target phrase collection using symbol-wise Huffman coding (Huffman, 1952; Moffat, 1989). Target phrase words, scores, and alignment points are encoded with different sets of Huffman codes which are switched during coding and decoding.

(a) Example phrase pair with alignment

Source	Target	Rank	Step	Result	Alignment
a	un	0	0.	une souche de bacille	0-0 2-1 1-3
a	une	1	1.	une[0] souche[2] de bacille[1]	∅
a	de	2	2.	une[0,1] souche[2,0] de bacille[1,1]	∅
a	la	3	3.	[0,1] [2,0] de [1,1]	∅
bacillus	bacillus	0	4.	[1] [2,0] de [1,1]	∅
bacillus	bacille	1	(c) Target phrase encoding procedure		
bacillus	bacilles	2	Step	Result	Alignment
strain	souche	0	0.	[1] [2,0] de [1,1]	∅
strain	contrainte	1	1.	[0,1] [2,0] de [1,1]	∅
strain	déformation	2	2.	(a)[1] (strain)[0] de (bacillus)[1]	0-0 2-1 1-3
of	de	0	3.	une souche de bacille	0-0 2-1 1-3
of	d'	1	(d) Target phrase decoding procedure		
of	du	2			

(b) Bilingual dictionary

(d) Target phrase decoding procedure

Figure 2: An encoding/decoding example

While the byte vector is just a large array of bytes, the byte offset vector is a more sophisticated structure. Instead of keeping offsets as 8-byte integers⁴ differences between the offsets are stored. A synchronization point with the full offset value is inserted and tracked for every 32 values.⁵

This turns the byte offset vector into a list of rather small numbers, even more so when the byte array is compressed. Techniques from inverted list compression for search engine indexes are used to reduce the size further, Simple-9 encoding (Anh and Moffat, 2004) for offset differences and Variable Byte Length encoding (Scholer et al., 2002) for synchronization points. As both techniques use less space if smaller numbers are compressed, the size of the structure keeps decreasing with decreasing offset differences. Therefore a better compression method for the byte array results automatically in a smaller byte offset vector. For the baseline phrase table, the roughly 215 million offsets use 260 Mbytes, but only 220 Mbytes for the rank-encoded variant.

4.3 The Phrase Decoder

The target phrase decoder contains the data that is needed to decode the compressed byte streams. It includes source and target word lists with indexes, the Huffman code sets, and if Rank Encoding is used, a sorted lexical translation table. The word lists and the translation table do not account for

⁴4-byte integers could hold byte offsets up to 4GB only.

⁵This an arbitrarily set step size.

more than 30 Mbytes. Huffman codes are stored as canonical Huffman codes, a memory efficient representation. The size of the target phrase decoder is treated as part of the size required to represent target phrases.

4.4 Baseline implementation

In the baseline implementation three different sets of Huffman codes are used to encode target words, scores, and alignments; encoding and decoding relies on switching between the three types of Huffman codes. For target phrase words and alignment points one special stop symbol has to be added. Scores and alignment points are encoded directly, target phrase words have an intermediate representation as integer identifiers which are looked up in a target word table. This implementation is referred to as “Baseline”. See Table 3 for the size characteristics of this implementation.

5 Rank Encoding

In this section Rank Encoding (R-Enc), a method for the compression of parallel texts, is presented. Strictly speaking, it is not a compression method by itself, but prepares the data in such a way that traditional compression is more efficient.

5.1 Outline of the Method

The main idea is to modify the probability distribution of symbols in the target data in such a way that the average length of the Huffman codes decreases. Bilingual data (a phrase table is nothing

else) has a property that helps with this problem.

Given a source phrase, a target phrase, and a bilingual dictionary of source and target words, it can be told for most target words which source words they are translations of. This information can be encoded into the target phrase and the surface forms of the target words can be dropped.

Figure 2 illustrates the procedure in more detail for an example phrase pair (2a) consisting of a source phrase, a target phrase, and alignment data. The available alignment information simplifies the process of finding correspondences between source and target words. Also a sample bilingual dictionary (2b) is included. The encoding procedure (2c) can be performed in four steps:

1. Alignments are moved to target words.
2. The source word is looked up in the dictionary and the position (rank) of the target word among the translations is recorded.
3. Aligned target words are dropped.
4. If positions of target and source word are equal, the alignment is dropped.

The target phrase consists now of three different types of symbols: ranks, ranks with alignment information, and target words.

The decoding procedure (2d) is as simple:

1. Alignment information is added to rank-only symbols based on their position in the pattern.
2. The original alignment data is reconstructed and source words are determined by their position in the source phrase.
3. Based on source words and ranks of their translation the target words are re-inserted.

In this example the alignment been reduced to the empty set, as it happens in most cases.

The counterparts of a source word in the dictionary are ordered by their decreasing translation probability $p(t|s)$. The lexical translation table generated during Moses training can be used for this. The lower the rank of a translation the higher is its probability. Symbols with low ranks are therefore more likely to occur, for a probability-based compression algorithm as Huffman coding this is highly desirable.

The described encoding scheme removes the actual target phrases from the phrase table, similarly

as the MPH in the source index removes source phrases, hence, the title of the paper. Source and target phrases can only be recovered during querying. Compression is thus achieved by moving information to the query.

5.2 Formal Description of Algorithms

Two functions for dictionary querying are defined: the rank $r(s, t)$ of a target word t relative to a source word s is the position of t within the list of translations of s . Conversely, given a source word s and a rank r the target word $t = t(s, r)$ is obtained from the lexical table.

For each of the three symbols types an encoding function is defined. Plain target words are encoded with e_1 , symbols that contain implicit alignments with e_2 , and e_3 encodes pairs of source position and target rank. The inverse functions e_1^{-1} , e_2^{-1} , and e_3^{-1} decode numerical values to symbols.

The codomains of e_1 , e_2 , and e_3 are required to be pair-wise distinct. Then the decision function d can determine the type of a given symbol based on its encoded numerical value. Based on that, the correct decoding function is applied.⁶

Algorithm 1 formalizes the encoding procedure. First, source word positions are partitioned into n sets J_i of positions of source words aligned with t_i . That way, worst-time complexity is $O(mn)$ if the given alignment contains all possible $m \times n$ alignments points. Average time complexity is $O(n)$ for alignments with about n alignment points.⁷

The algorithm processes the target phrase \mathbf{t} of length n from left to right. For each target word t_i all source words that are aligned with t_i are examined. If t_i is not aligned with any source word, it is encoded as a plain symbol of type 1.

For an aligned target word t_i , the minimal rank r of t_i relative to any of these source words is determined. If for the minimal rank there is more than one source word aligned with t_i , the left-most source word position k is chosen. This two-fold selection of minimal values is crucial for the size-reduction effect of the later compression. Lower values of rank and position appear more often and are assigned shorter Huffman codes. If $k = i$, i.e. source and target word occupy the same position,

⁶The implementation of the decision, encoding, decoding functions relies on bitwise operations on integer values, but in fact any representation can be used if it fulfills the previous requirements.

⁷According to the statistics for the Coppa phrase table there are actually less alignment points than target words.

```

Function EncodePhrase( $s, t, A$ )
begin
   $\hat{t} \leftarrow \langle \rangle$ 
   $\hat{A} \leftarrow A$ 
  foreach  $\langle j, i \rangle \in A$  do
     $J_i \leftarrow J_i \cup \{j\}$ 
  end
  foreach  $i \in \{1, \dots, |t|\}$  do
    if  $J_i = \emptyset$  then
       $\hat{t} \leftarrow \hat{t} \cdot \langle e_1(t_i) \rangle$ 
    else
       $r \leftarrow \min\{r(s_j, t_i) : j \in J_i\}$ 
       $k \leftarrow \min\{j : j \in J_i \wedge r(s_j, t_i) = r\}$ 
      if  $k = i$  then
         $\hat{t} \leftarrow \hat{t} \cdot \langle e_2(t_i) \rangle$ 
      else
         $\hat{t} \leftarrow \hat{t} \cdot \langle e_3(t_i) \rangle$ 
      end
       $\hat{A} \leftarrow \hat{A} \setminus \{\langle k, i \rangle\}$ 
    end
  end
  return  $\langle \hat{t}, \hat{A} \rangle$ 
end

```

Algorithm 1: Rank encoding

only the rank r is encoded (symbol type 2). Otherwise, k and r are encoded together as one symbol (symbol type 3). Alignment points used during encoding are dropped from the input alignment. Only the unused alignment points in \hat{A} are saved in the alignment of the encoded target phrase.

Decoding (algorithm 2) is straightforward. The encoded target phrase pattern \hat{t} is processed from left to right. Each symbol is decoded using the appropriate decoding function based on the symbol type. For symbols of type 1 no alignment point is restored. Symbols of type 2 are decoded to a rank value and looked up using the source phrase word that is located at same position i as the current target phrase word t_i , an alignment point $\langle i, i \rangle$ is restored. For symbols of type 3, the source word position j is recovered from the symbol and the target word is looked-up, a point $\langle j, i \rangle$ is added to the alignment. Average and worst-case time complexity is equal to $O(n)$. Phrase tables without explicit alignment data can be encoded by providing a Cartesian product of source and target word positions instead. Average complexity for encoding is then equal to the worst case complexity $O(nm)$. Decoding time complexity is unchanged.

```

Function DecodePhrase( $s, \hat{t}, \hat{A}$ )
begin
   $t \leftarrow \langle \rangle$ 
   $A \leftarrow \hat{A}$ 
  foreach  $i \in \{1, \dots, |\hat{t}|\}$  do
    switch  $d(\hat{t}_i)$  do
      case 1
         $t \leftarrow t \cdot \langle e_1^{-1}(\hat{t}_i) \rangle$ 
      case 2
         $r \leftarrow e_2^{-1}(\hat{t}_i)$ 
         $t \leftarrow t \cdot \langle t(s_i, r) \rangle$ 
         $A \leftarrow A \cup \{\langle i, i \rangle\}$ 
      case 3
         $\langle j, r \rangle \leftarrow e_3^{-1}(\hat{t}_i)$ 
         $t \leftarrow t \cdot \langle t(s_j, r) \rangle$ 
         $A \leftarrow A \cup \{\langle j, i \rangle\}$ 
    endsw
  end
  return  $\langle t, A \rangle$ 
end

```

Algorithm 2: Rank decoding

5.3 Results

Table 2 summarizes the results for Rank Encoding applied to target phrases of the Coppa phrase table. Here, only figures for target words and alignment points are compiled as we want to evaluate the performance of Rank Encoding alone.

Rank Encoding reduces the number of distinct target words from 550,446 to 86,367. In the baseline phrase table the first 100 most frequent symbols account for 52 percent of the running target words, but for 91 percent if Rank Encoding is used. These different distributions of symbols and frequencies affect the later applied Huffman coding significantly. The number of bits per running target words decreases from 10.8 to 6.5. The size reduction is even more substantial for alignment points as the number of bits per running alignment point drops from 5.4 to 0.5. This is the effect of the majority of alignment points being encoded into target words symbols. Of ca. 1.5 billion alignment points only 55 million (3.7 percent) are compressed explicitly. Bit numbers include the overhead introduced by stop symbols, for R-Enc the bilingual dictionary is added as well. The total size of target phrases with alignments is reduced by 56 percent from 3,096 Mbytes to 1,351 Mbytes. In the context of the complete phrase table (in Table 3) the size reduction is 22 percent.

	Baseline	R-Enc
Distinct target words:	550,446	86,367
Bytes per target phrase (without scores):	9.7	4.2
Bits per target word:	10.8	6.5
Bits per alignment point:	5.4	0.5
Bits per symbol (words & alignment):	8.2	3.6
Total space (Mbytes):	3,096	1,351

Table 2: Results for rank-encoded target words and alignments

	Moses	Baseline	R-Enc
Total size in Mbytes (ordered) :	29,418	7,681	5,967
Ordered source phrase index (Mbytes):	5,953	1,750	
Target phrase storage (Mbytes):	23,441	5,873	4,127
Target phrase decoder (Mbytes):	—	59	90
Bytes per target phrase:	73.1	18.5	13.2
Bits per symbol (words & score & alignment):	40.6	10.3	7.2
Translation time (1st run):	1606 s	1322 s	1450 s
Translation time (2nd run):	1051 s	940 s	957 s
Memory usage peak:	1.6 G	2.7 G	2.8 G

Table 3: Comparison of phrase table implementations

We measured the speed of our phrase table variants and the Moses phrase table on the first unique thousand sentence pairs from test set provided by WIPO⁸. Two scenarios are considered: During the “1st run” operation system IO caches are dropped before translation. During the “2nd run” the translation process is started with the same parameters, but IO caches of the previous run are available. Caching as provided by Moses is enabled.

Concerning speed, our phrase table implementations outperforms the Moses binary phrase table. The difference is more noticeable for first runs. One has to keep in mind, that the search for translation options occupies only a small percentage of time during the translation. The decoding process itself is much more time consuming. Improved performance for first runs can be explained by greatly reduced disk access which levels out increased processing requirements due to decompression. Speed is more similar for second runs, where all phrase table variants can take advantage of the IO caching mechanism of the operation system. The Moses phrase table fares well when peak

memory consumption is compared. The Baseline and the rank encoded variant use over 1 GB more memory than the binary Moses phrase table. This due to the source phrase index which at the moment is fully kept in memory.⁹

6 Conclusions and Future Work

Rank Encoding if combined with Huffman coding reduces the size of a phrase tables substantially when compared to bare Huffman coding. Translation speed is faster or comparable to the binary phrase table in Moses. Memory requirements are currently higher. In the presented phrase table implementation compression has been achieved by removing actual representation of source and target phrases from the phrase table. Both can only be recovered when the phrase table is being queried with potential source phrases.

There is still much potential for further size reductions. In this work the focus lay mainly on tar-

⁸<http://www.wipo.int/patentscope/translate/coppa/testset2011.tmx.tgz>

⁹After submission of this paper, we managed to reduce the space requirements of the index and to implement a lazy loading procedure. Instead of 1.7 GBytes only 300 MBytes are consumed for the translation of the test set. The methods used to achieve this reduction will be described in a forthcoming paper.

get words and alignment information. Now scores take up a majority of space in a target phrase and can surely be reduced by mathematically grounded smoothing methods without a noticeable impact on translation quality. Also, since a translation table is already used for R-Enc on-line calculation of lexical probabilities can be considered an option. The source phrase index needs to be optimized. Other order preserving or monotonous hash functions or indexing methods are to be reviewed and tested. The impact of fingerprint bit length on translation quality should be examined.

Concerning R-Enc, other similar encoding techniques need to be investigated, especially extending the described approach to full bilingual phrase pairs instead of word pairs. Due to the highly repetitive nature of phrase tables this might be a promising course of research.

Acknowledgments

This research is funded by the Polish Ministry of Science and Higher Education (grant no. N N516 480540). The idea for this work was conceived during a stay of the author's at WIPO in Geneva, while working on the in-house MT system.

References

- Anh, Vo Ngoc and Alistair Moffat. 2004. Index compression using fixed binary codewords. In *Proceedings of the 15th Australasian database conference*, pages 61–67.
- Belazzougui, Djamel, Fabiano C. Botelho, and Martin Dietzfelbinger. 2009. Hash, displace, and compress. In *Proceedings of the 17th European Symposium on Algorithms*. Springer LNCS.
- Callison-burch, Chris and Colin Bannard. 2005. Scaling phrase-based statistical machine translation to larger corpora and longer phrases. In *In Proceedings of ACL*, pages 255–262.
- Conley, Ehud S. and Shmuel T. Klein. 2008. Using Alignment for Multilingual Text Compression. *Int. J. Found. Comput. Sci.*, 19(1):89–101.
- Guthrie, David, Mark Hepple, and Wei Liu. 2010. Efficient Minimal Perfect Hash Language Models. In *Proceedings of the Seventh Language Resources and Evaluation Conference*.
- Huffman, David. 1952. A Method for the Construction of Minimum-Redundancy Codes. *Proceedings of the IRE*, 40(9):1098–1101.
- Johnson, J. Howard, Joel Martin, George Fost, and Roland Kuhn. 2007. Improving translation quality by discarding most of the phrasetable. In *In Proceedings of EMNLP-CoNLL07*, pages 967–975.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Annual Meeting of the ACL, Prague*.
- Levenberg, Abby, Chris Callison-Burch, and Miles Osborne. 2010. Stream-based translation models for statistical machine translation. In *Proceedings of NAACL-HLT 2010, Los Angeles*, pages 394–402.
- Lopez, Adam. 2008. Tera-scale translation models via pattern matching. In *Proceedings of the 22nd International Conference on Computational Linguistics, COLING '08*, pages 505–512, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Moffat, A. 1989. Word-based Text Compression. *Softw. Pract. Exper.*, 19(2):185–198.
- Nevill-Manning, Craig G. and Timothy C. Bell. 1992. Compression of Parallel Texts. *Inf. Process. Manage.*, 28(6):781–794.
- Pouliquen, Bruno and Christophe Mazenc. 2011. COPPA, CLIR and TAPTA: three tools to assist in overcoming the language barrier at WIPO. In *MT-Summit 2011*.
- Sanchez-Martinez, Felipe, Rafael C. Carrasco, Miguel A. Martinez-Prieto, and Joaquin Adiego. 2012. Generalized Biwords for Bitext Compression and Translation Spotting. *Journal of Artificial Intelligence Research*, 43:389–418.
- Scholer, Falk, Hugh E. Williams, John Yiannis, and Justin Zobel. 2002. Compression of inverted indexes for fast query evaluation. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '02*, pages 222–229.
- Talbot, David and Thorsten Brants. 2008. Randomized Language Models via Perfect Hash Functions. In *Proceedings of ACL-08: HLT*, pages 505–513, Columbus, Ohio, June. Association for Computational Linguistics.
- Zens, Richard and Hermann Ney. 2007. Efficient Phrase-table Representation for Machine Translation with Applications to Online MT and Speech Translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, Rochester, NY.
- Zhang, Ying and Stephan Vogel. 2005. An efficient phrase-to-phrase alignment model for arbitrarily long phrase and large corpora. In *Proceedings of the 10th Conference of the European Association for Machine Translation (EAMT-05)*, pages 30–31.

Creating Term and Lexicon Entries from Phrase Tables

Gregor Thurmair

Linguattec

g.thurmair@linguatec.de

Vera Aleksić

Linguattec

v.aleksic@linguatec.de

Abstract

This document describes a tool which extracts term and lexicon entries from SMT phrase tables, without further reference to monolingual data. It applies filters to such tables, and builds lexicon entries from the ‘good’ candidates. Error rates of the tool can be as low as 7.3%, accumulated from source, target, and transfer errors.¹

1 Introduction

It is a common understanding that machine translation systems need to be adapted to the domain and text type they are supposed to translate. For knowledge-driven systems, such adaptation is done by means of lexicon update: The domain terminology is identified, and coded as a special additional lexicon repository, loaded at runtime. In the age of data-driven technology, terminology is extracted from corpus data, and so are translation equivalents for the found terms.

1.1 Task

The task of the P2G (phrasetable2glossary) tool is to create proper bilingual lexicon entries from comparable corpus data; the technique should be usable for special domains, and should create output which can be imported into a backend (rule-based) MT system.

The question what the target of a bilingual extraction component is, is difficult to define. Real term banks, even in the same domain, contain very different material, depending on the subdomain and focus, and the skills of the translators involved. As a result, the term extraction process

will always contain a step whereby humans investigate a term list and decide which entry candidates they want to keep for term bank import.

The task of a term extraction tool is to prepare this candidate list. The quality of the extraction tool is determined by the effort it makes to go through this list.

The approach of P2G consists of the following steps:

- Step 1: Extract phrases with a good chance of being translations of each other. This means to apply word and phrase alignment to the input. Tools exist which do this.
- Step 2: Not all phrases are well-formed terms. Therefore, the term candidates are filtered on several levels:

Frequency filter: only phrases with a frequency and translation probability above a given threshold are considered as candidates.

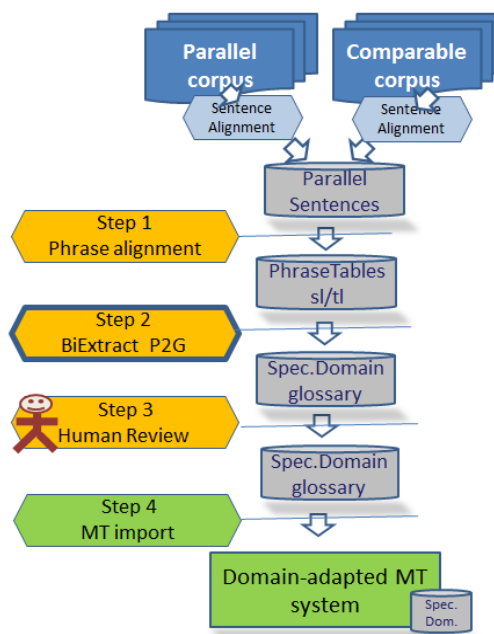


Fig. 1: Lexicon extraction workflow

© 2012 European Association for Machine Translation.

¹ This work was done in the context of the FP7-ICT projects ACCURAT (248347) (system core) and the PANACEA (248064) (language extensions).

Linguistic filter: have an internal linguistic structure. Only candidates that match this structure are legal term candidates.

Lexicon filter: This is an optional user-defined component which allows eliminating non-terms.

- Step 3: The resulting list will be given to human post-editing, to correct erroneous system decisions. The quality of the tool is a key factor for the efficiency of this step.
- Step 4: The correction result will be imported into a rule-based MT system (e.g. Linguatrec's *Personal Translator*). Care must be taken that all the annotations required by the backend MT system are available.

The focus of this paper is on step 2, term and lexicon extraction.

1.2 Related Work

There is an abundance of literature on bilingual term extraction. In the present context, we focus on papers which use phrase alignment for term extraction.

Macken et al. 2008 use linguistic pre-processing on the SL and TL side, and try to identify chunks from which they can conclude phrase similarity. They report an error rate between 15% and 33%, for the automotive domain. Our approach has a much smaller error rate, and does not need any corpus pre-processing.

Ideue et al. 2011 first extract term candidates from SL and TL texts, and then try to find matches in bilingual phrase tables, which they score according to different measures. They have a very small evaluation set (only 100 terms); however, the argument would be that

- a. if a string is a term then it *must* show up in the aligned phrases somehow,
- b. if it shows up in the phrase tables then it *must* be able to be extracted from there, and no reference to any source and target sentences is required
- c. as a consequence, no comparison / distance between sentence-based and phrasetable-based terms needs to be computed.

In turn, our approach needs *only* aligned phrases as input, and tries to find the good terms in them.

Wolf et al. 2011 have a similar objective than the present report, namely using phrase tables for RBMT lexicon improvement; they use a full RBMT analysis (and generation) component to identify translation candidates in the phrase tables, by exploring if a phrase table entry matches constraints imposed by the MT tree. They do not

report evaluation results for term extraction but only for overall MT quality improvements; however they share a lot of aspects (like the need to create MT-compatible entries) with the present work.

Our approach is more robust, as it does not need a full MT system for term identification, and does not require 'phrase-table-external' term candidates; it applies linguistic patterns which are usable by most RBMT systems, and provides annotations which should enable a straightforward lexicon import.

All these approaches follow the standard approach towards bilingual term extraction, which is a two-step procedure: First *identification* of term candidates in the source language, and then *mapping* of source to target term candidates. Usually the corpus data need to be preprocessed, from the level of lemmatization / POS tagging (Caseli/Nunez 2006) to the level of logical form creation (Menezes/Richardson 2001); this is always a source of error.

1.3 Approach

The system presented here takes the opposite approach: It does mapping *first* (using state-of-the-art phrase aligners), and *then* it does extraction from the aligned phrases, by applying filters to the phrases. This approach follows the following considerations:

1. If a (monolingual) source language term candidate does not have a correspondence in the target language, it is unlikely that it is really a term. In turn, this means that if something is a term (i.e. a relevant concept) in a bilingual setup, then it *must* show up in the alignment results, and the alignment can be used as a filter for term candidates.
2. The best available alignment tools produce translation tables which contain all possible term mappings (and beyond that many phrases which would not be considered as proper terms). So most of the correct term candidates *will* be represented in such translation tables.
3. As a result, the task consists in identifying 'good' term candidates from phrase table input. This is achieved by applying different *filters* to such input to extract the good terms.

Therefore, the approach reverses the identification and mapping steps, and identifies term candidates only from alignment results. The *only* source of input therefore is a set of aligned

phrases, as produced by standard aligners. No monolingual extraction is needed.

2 Mode of Operation

As mentioned earlier, the approach is to apply filters on input records of aligned phrases, whereby formats of different alignment tools are supported as input.

Three filters are applied, as shown in Fig. 2.

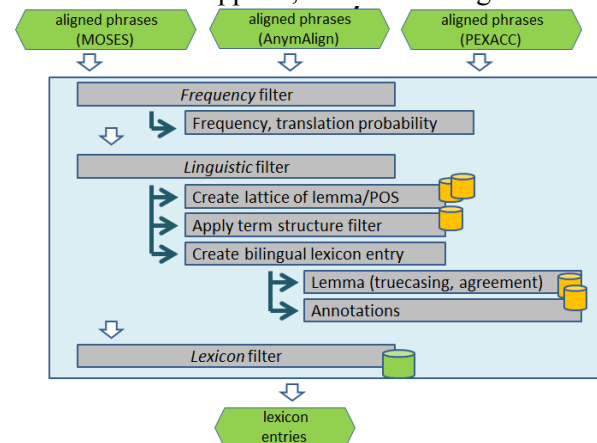


Fig. 2: Operation flow of the P2G system

The filters are:

- A **Frequency filter**: Only phrases with a given frequency and / or translation probability are accepted as term candidates
- A **Linguistic filter**: Only phrases which have certain linguistic properties are acceptable.

If a candidate passes the linguistic filter, it is brought into the right lexicon form, in terms of lemma creation, assignment of annotations, etc.

- The **Lexicon filter** compares the lexicon entries just produced with a filter resource. This way, candidate entries can be removed which are already known, or are not wanted, or should not be part of the output for some other reason.

Details are given in the following sections.

2.1 Frequency Filter

As the system does not itself create alignments (i.e. translation candidates), it must rely on the efficiency of the statistical alignment tools from which it receives the aligned candidates. The first step is therefore to identify the best translation proposals, in terms of recall (as many terms as possible) and precision (as good translations as possible).

Two factors influence the translation quality of the P2G tool: the selection of the alignment tool,

and the selection of the thresholds for frequency and translation probability.

For the alignment tool, it can easily be seen that GIZA++ only is insufficient, as no multi-word entries are found, which form close to 50% of a lexicon / term list, esp. in narrow domains. So the focus was on phrase alignment tools, which also give superior quality in translation (Och/Ney 2004). To create phrase alignment, two alignment methods were tried out²:

- Giza++ and **MOSES** (cf. Koehn 2010), creating Phrase Tables. From the *LT_automotive* input data (cf. below), a phrase table with about 7.97 mio entries was built.
- Phrases as produced with **Anymalign** (Lardilleux/Lepage 2009). Anymalign created about 3.14 mio word/phrase pairs from the same input data.

It soon turned out that if **frequency** is not considered, too much noise would be in the output. Therefore, frequency (on source and target side) is used and set to > 1 .

For the **translation probability**, tests were done to find the optimal recall / precision combination.

The two alignment systems were compared, using different values for the translation probability. For evaluation, a random set of term candidates manually inspected³, and the errors in alignment / translation were counted⁴. The results are given in Table 1.

Tool	transl. prob.	no entries	errors
MOSES	$p > 0.8$	12.000	5.54%
MOSES	$0.6 < p < 0.8$	3.900	5.42%
MOSES	$0.4 < p < 0.6$	20.000	55.11%
AnymAlign	$p > 0.7$	12.600	46.91%
AnymAlign	$p > 0.8$	10.900	47.56%

Table 1: Translation errors for different alignment methods and probabilities

It can be seen that the MOSES Alignment has a much better quality, and is in the reach of being usable; AnymAlign error rates are ten times higher. For AnymAlign, taking a higher threshold (0.8 instead of 0.7) does not improve alignment quality. Overall MOSES input with a

² Input from PEXACC (Ion et al. 2011) for comparable corpora is also supported.

³ Entries starting with the letters C, F, and S.

⁴ There are always unclear cases among translations (e.g. transfers usable only in certain cases); they were not counted as errors. Errors are only clearly wrong translations; however a range of subjectivity remains.

threshold of 0.6 for P(f|e) seems to give best results for term extraction, for this size of phrase tables⁵, with an overall error rate of about 5.5%: It increases recall without reducing precision.

It should be noted that alignment errors result from external phrase alignment components, and are just ‘inherited’ by the current extraction system. However, they count in the overall workflow evaluation: Incorrect translation proposals lead to significantly higher human reviewing effort.

2.2 Linguistic Filter

Not all phrase aligned candidates which pass the frequency filter are linguistically meaningful. So only the ones which can be terms, or lexicon entries, are extracted⁶. Most such terms have an internal linguistic structure, described by a part-of-speech tag sequence. So the internal structure of the linguistic filter is:

- Create a word lattice for the input string, providing the different readings for each of the input words
- Match the input lattice to the legal term patterns, on source and target side;
- Create a lexicon entry for candidates with a successful match on both source and target side, with proper lemma and its annotations.

a. Word lattice

First, each candidate input phrase is tokenized and normalized in spelling and casing⁷.

Next, each token is lemmatized to find its base form and part-of-speech tag. Lemmatization is basically done by lexicon lookup. Unknown words are handled by a POS-defaulting component; for German unknown words, a decomposer component is called to find a known head word. This procedure is documented in (Thurmair et al. 2012).

As tokens can have multiple readings, the result of this procedure is a word lattice consisting of the respective readings of each of the single words of a candidate. This procedure is lan-

guage-specific, and is done on both source and target side.

b. Term Pattern matching

From the word lattice, all possible POS sequences are created, and compared to the legal term structure patterns.

The patterns go significantly beyond the ‘usual suspects’; they were collected as the result on an inspection of a large terminological database. For German, patterns for the structures are provided⁸ as shown in Fig. 3.

```
Term ::= AdP? NoC (NoC | NP | PP)?
AdP ::= Ad | VbP
NP ::= Dt (AdP)? NoC
PP ::= (Ap Dt? AdP? NoC) | (ApPD AdP? NoC)
```

Fig. 3: Term structure for German

The maximum length of such patterns is set to 6 members; longer terms are hardly ever found in term banks, and are even rarer in running texts.

The pattern filters are of course language-specific; e.g. in German and Greek, patterns must be foreseen which cover post-head NP’s in genitive case, French and Spanish patterns cover both prenominal and postnominal adjectives, etc.

The matching strategy is a simple best-first approach, i.e. it returns the first match. It could be improved by sorting the multiword patterns according to frequency, and/or giving weights to the different POS readings of an input word. However such extensions would only marginally affect the results, and would not avoid the most frequent errors of this filter (cf. the evaluation below, section 3).

The pattern filter is applied to the candidates on both the source and target side, independently of each other, to be able to map a source language single word (e.g. a German compound) to a target language multiword expression. If both side candidates pass the filter, then the sequence of readings corresponding to the matching patterns is given to the entry creation module.

c. Term and Lexicon Entry Creation

All entries which have passed the filter so far must be brought into a proper canonical form. The creation of lexicon entries for source and target consists of two parts:

⁵ However, this changes with the size of the phrase table, cf. section 5.5 below.

⁶ As a consequence, there are phrases in the phrase table which are perfectly valid translations, however would never be found in a term bank.

⁷ Normalization in casing is problematic as it also lowercases proper names. However, *not* doing it would lead to significant errors due to the fact that phrase tables contain many capitalized non-propername words. The output would contain pseudo-doublents from capitalized and non-capitalized term proposals. Example: ‘*Financial debt*’ where lowercased ‘*financial debt*’ can also be found.

⁸ Not covered: Proper nouns (*Lufthansa Service Center*), and terms containing conjunctions (*Facts and Figures*), as the backend MT system cannot cope with some of such structures.

- Creating proper **lemmata**. This is required for both term and lexicon use.
- Creating proper **lexicon entries**. This is relevant if the extracted terms are to be integrated into MT systems; such systems usually require certain annotations (at least part of speech information).

Lemma creation implies the creation of a canonical form for the entry. This has two aspects:

- **Truecasing** of all lemma parts: Proper names and German common nouns should be capitalized, the other forms lowercased.
- Production of the **canonical form** of the lemma.

The *head* (or the term if it is a single word) is lemmatized, and the lemma is given as canonical form. In multiword entries, the head position is given in the pattern.

The *modifiers* in a multiword entry are treated as follows:

Head-modifying adjectives must be set into gender-number-agreement with their head (it ‘*cardiopatía coronárica*’, es ‘*cuestión política*’)⁹. Therefore the production of the lemma of multiword entries requires knowledge about the gender of the head. To provide this, a special component (gender defaulter) has been added to the system which consults an appropriate resource; depending on the gender of the noun, the adjective is inflected¹⁰.

The *post-head modifiers* of the multiword stay in their inflected form: de ‘*Oberfläche mit speziellen Farbpigmenten*’, en ‘*surface with special color pigments*’ would leave the PP untouched.

Based on these two principles, the multiword lemma is composed¹¹.

It should be noted that the step of creating canonical forms can create duplicates (e.g. if a phrase table contains one entry for a singular and another

one for a plural noun). Such duplicates must be eliminated before the final list is output.

Lexica go beyond term lists as their entries need **annotations**. The lexicon entries in P2G show the following annotations:

All of them have a lemma, a part of speech, and a reading number, as these elements constitute an entry. In addition, they have annotations which depend on a feature called ‘*entrytype*’, with values ‘*singleword*’, ‘*compound*’, ‘*multiword*’.

Single word entries are annotated with gender (in German) and inflection; this information is either taken from the lexicon, or defaulted.

Multiword entries and compounds (i.e. the agglutinated German compounds) share the same entry structure; they provide: the head position, the sequence of lemmata, and the sequence of parts of speech of which the multiword consists. These annotations allow for a successful identification of multiword terms in texts.

Of course, the lexicon must contain much more information; however this goes beyond what the term extraction can contribute. In turn, the use which can be made of the provided annotations depends on the single backup MT systems and their import possibilities: Most systems can use (or even require) POS information, but e.g. not all multiword term patterns are supported (e.g. terms containing conjunctions). Tests on transfers, like in (Caseli/Nunez 2006), are not created, however.

The final output of the linguistic filter consists either of complete *lexical* entries (for MT import), or of *term* entries (for human lookup), depending on an output format parameter.

2.3 Lexicon Filter

Before human post-editors select the entries which they really want to keep, a possibility has been created to remove unwanted term candidates. Such entries could be:

- Candidates which are already known; they need not be reviewed a second time
 - Candidates which do not belong to a specific domain (e.g. automotive); the filter then would be a general-domain lexicon, letting pass only narrow-domain words
 - Candidates which contain certain stopwords (like en ‘*large*’)
 - Candidates which are known to be irrelevant.
- The system offers the option to apply a filter which blocks this kind of entries. Users would

⁹ In German, there are even two options, the weak inflection (<das> ‘*niedrige Zinsniveau*’) or the strong one (<ein> ‘*niedriges Zinsniveau*’). Both can be found in dictionaries; the strong inflection is more difficult as it requires knowledge of the head noun gender; unfortunately this is the form expected by the backend MT system.

¹⁰ The system uses a static inflection resource for this.

¹¹ These heuristics for truecasing and for lemma creation leave room for errors, e.g. in cases where the prenominal adjective is in comparative form (de ‘*der frühere Präsident*’ -> *‘*der frühe Präsident*’), or in cases where the head should be in plural (en ‘*facts & figures*’ -> *‘*fact & figure*’). However, they show the best performance overall.

provide the filter data themselves; only non-matching entries pass the lexicon filter.

3 Evaluation

3.1 Methodology

As explained above, it is difficult to evaluate a term extraction tool vis-à-vis a gold standard; term extraction always depends on the knowledge and interest of the users. Despite of a research focus on the selection of relevant entries ('termhood', cf. Vu et al. 2008, Wong et al. 2007, Kit 2002), there will always be a step where users review the list of candidates produced by the extraction tool, and select the entries they want to keep.

While there is no clear view which entries *should* be in the term list, on the other side, there is agreement on which candidates should *not* be presented, and be considered as noise: wrong translations, the same entry in singular and plural form, or in capitalized and lowercased spelling, etc. It is *this* type of entry, which a term extract evaluation should focus on: Creation of only 'good' term candidates. This is what the following evaluation does.

3.2 Data

Several corpora were used for testing, related to several projects:

- The PANACEA corpora for environment, prepared by DCU: (*DCU_ENV*) and labour legislation (*DCU_LAB*)¹²
- Corpora in the Health and Safety domain, collected by Linguattec (*LT_H&S*) in different languages
- A corpus on automotive texts, collected by Linguattec (*LT_autom.*)
- The ACCURAT corpora for automotive, in two versions, prepared by DFKI: *DFKI_adapt* and *DFKI_lexacc*¹³.

The size, languages treated, size of phrase tables created, and number of glossary entries extracted is given in Table 2.

3.3 Evaluation Procedure

From all corpus data sets, term candidates were extracted by the P2G system. From these candidates, term candidates were selected randomly. These candidates were evaluated manually by two evaluators.

Corpus	lang.	No. sentences	Phr.Tab. size
DCU_ENV	en-fr	29 K	0.4 M
DCU_LAB	en-fr	21 K	0.8 M
LT_H&S	en-fr	52 K	2.9 M
LT_H&S	en-es	48 K	2.6 M
LT_H&S	en-it	40 K	2.1 M
LT_H&S	en-pt	14 K	0.6 M
LT_autom.	en-de	155 K	7.97 M
DFKI_adapt	en-de	1483 K	85.0 M
DFKI_lexacc	en-de	1595 K	83.9 M

Table 2: Test corpora (no. sentences, phrasetable size, size of extracted glossaries)

Overall, 99 K bilingual term candidates were extracted of which 17.2 K (17%) were manually evaluated; details are given in Table 2 below.

3.4 Results

First, speed was measured for the corpora. Depending on the frequency filter, the system processes between 45K (no filter) and 170K (0.8 filter) entries per second on a standard PC. This would be fast enough for practical use.

As for quality and errors, two kinds of errors are distinguished in the evaluation:

- *Translation* errors, i.e. the candidates are not translations of each other. These errors are produced by the aligners, as explained above. For the final tests, MOSES was selected as alignment method, with a translation probability threshold set to 0.6 and a frequency threshold set to >1.
- *Lemma and annotation* errors; these errors are created by the P2G tool. They are obviously language-specific; an error analysis is given below.

Table 3 shows the evaluation results. The average error rate of the complete P2G system is 9.26%, varying from 7.3 to 14.4%.

Translation errors: Translation errors vary from 1.5% to 12.7%, with 5.1% on average.

Translation errors seem to correlate with the size of the phrase tables¹⁴: Larger phrase tables show a lower translation error rate for the extracted terms. This is not particularly surprising, as more data usually lead to better performance.

Translation errors are produced by MOSES alignment, and are not accessible to the P2G tool; however, they increase the total error rate.

¹² cf. Mastropavlos / Papavassiliou. 2011.

¹³ cf. ACCURAT Deliverable D4.2: Improved baseline SMT systems adjusted for narrow domain. 2012

¹⁴ DCU_ENV and DCU_LAB need to be considered in more detail.

	PhrTab size	Gloss. size	Transl. error	P2G error	Total error
DCU_ENV	400	2.8	5.2%	1.3%	7.8%
DCU_LAB	800	4.5	4.9%	1.2%	7.3%
LT_H&S fr	2.900	10.7	11.3%	1.3%	13.9%
LT_H&S es	2.600	13.2	10.9%	0.4%	11.6%
LT_H&S it	2.100	9.9	9.8%	2.3%	14.4%
LT_H&S pt	600	4.4	12.7%	0.4%	13.5%
LT_autom.	7.970	15.7	5.7%	2.8%	10.3%
DFKI_adapt	85.000	23.2	1.5%	3.3%	8.0%
DFKI_lexacc	83.900	23.3	1.7%	3.1%	7.9%

Tab. 3: Evaluation results: Phrase Table size (K entries), size of extracted glossaries (K entries), error rates of translation, of P2D, and combined error rates

P2G errors: P2G errors vary from 0.4% to 3.3%, depending on the languages involved¹⁵, with an average error rate of 2.1%. Main of errors are:

- errors in linguistic filtering: either homograph words pass the filter (en **are permanent* as *are* etc. has also a noun reading; similar it *sono* in **sono piccolo*, etc.). Or patterns pass the filter which are no terms but happen to have the ‘right’ structure: en **strategy for example*, it **formazione a favore*, de **Flüchtlings-fonds für den Zeitraum*.
- errors in lemma creation: either errors in casing (en **fujitsu*, **flemish port*), mostly due to lexicon gaps, or errors in agreement, (de **freundlicher Wort*, fr **force élevées*, es **animal infectados*).

Many of these errors can be corrected by improvements of the backend components (dictionary, gender defaulters etc.), which would bring the P2G error rate down by an estimated 1%.

The P2G errors do not depend on the size of the data; they are language-dependent of course: Errors in German result from more complicated gender agreement; in Italian, homograph problems, in English casing problems are the main error source. Variations of error rates within one language in the different test sets do not seem to be significant.

Total errors: As the output of the system is a bilingual lexicon, i.e. description of two source terms plus their translation, the error rates accumulate, so the overall error rate of the tool is two P2G errors plus translation errors; the total error

¹⁵ P2G supports the languages en de fr es it pt

rate is somewhat linear to the translation error rate. In total it is between 7.3% and 14.4%, which means that 8 entries out of 100 need to be corrected by human reviewers. This can be considered a reasonable result of a term extraction component.

3.5 Recall Issues

Another observation is that the number of phrase table entries containing good terms decreases with the size of the phrase table: As Table 2 shows, the extraction factor for smaller tables is about 150 phrases per ‘good’ term, while for the large tables it is about 3600, producing only 23.000 terms. So, either these tables contain more irrelevant entries, or the translation probability factors need to be adjusted in relation to the size of the phrase table.

A comparison between the terms of *DFKI_lexacc* and *DFKI_adapted* showed that there was a difference of about 15% in the output entries, meaning that there are at least 15% undetected ‘good’ terms in the data.

As a consequence, the translation probability threshold for the frequency filter should be set depending on the size of the phrase table. To test this, the *DFKI_lexacc* data were split into packages depending on the translation probabilities. In each package, about 1000 entries were manually evaluated. The result is shown in Table 4.

Translation . probability.	no entries found	error rate
p > 0.8	5.900	2.11%
0.6 < p < 0.8	20.500	0.58%
0.4 < p < 0.6	54.900	2.33%
0.2 < p < 0.4	58.100	4.03%
0.0 < p < 0.2	1.001.900	59.69%

Tab. 4: Error rates and probabilities in large phrase tables (*DFKI_lexacc*)

The results show that the entry sets with a probability > 0.4 have basically the same error rate (the 0.58% may be due to some data idiosyncrasies); entry sets from 0.2 to 0.4 have a slightly increased error rate, and entries < 0.2 cannot be used.

This means that recall can be improved dramatically by lowering the probability threshold, with no or just minimal loss in precision, cf. Table 5.

translation probability	no. entries retrieved	expected translation error rate
P (fe) > 0.4	67.664	2.25 %
P (fe) > 0.2	109.418	3.53 %

Tab. 5: Recall improvement for large phrase tables

As a result, the P2G term extraction tool can produce a 110 K bilingual glossary from phrase tables where 92 out of 100 entries are correct (7.7% total error rate¹⁶).

Schalterkontakt	No	switch contact	No
Schalterraum	No	switch room	No
Schaltfinger	No	shift finger	No
Schaltgabel	No	shift fork	No
Schaltgetriebe eines Fahrzeugs	No	gearbox of a vehicle	No
Schalthebel	No	shift lever	No
Schalthebelanordnung	No	shift lever assembly	No
Schalthebel-Positionssensor	No	shift lever position sensor	No
Schalthebelsystem	No	shift lever system	No
Schalthebelvorrichtung	No	shift lever device	No
Schaltintervall	No	shift interval	No
Schaltkabel	No	shift cable	No
Schaltkanal	No	shifting channel	No
Schaltknopf	No	button	No
Schaltkolben	No	shifting piston	No
Schaltkraft	No	switching force	No
Schaltkraftsensor	No	gear shift sensor	No

Fig. 4: Example term output (automotive domain)

References

- Caseli, H., Nunes, M., 2006: Automatic induction of bilingual resources for machine translation: the ReTraTos project. *Machine Translation* 20,4
- Daille, B., Morin, E., 2005: French-English Terminology Extraction from Comparable Corpora. *Proc. IJCNLP 2005*
- Fung, P., McKeown, K., 1997: Finding Terminology Translations from Non-parallel Corpora. *Proc. 5th Annual Workshop on Very Large Corpora (VCL 97)*, Hong Kong
- Gamallo Otero, P., 2007: Learning Bilingual Lexicons from Comparable English and Spanish Corpora. *Proc MT Translation Summit Copenhagen*
- Gamallo Otero, P., 2008: Evaluating Two Different Methods for the Task of Extracting Bilingual Lexicons from Comparable Corpora. *Proc. LREC Workshop on Comparable Corpora, Marrakech*
- Ideue, M., Yamamoto, K., Utiyama, M., Sumita, E., 2011: A Comparison of Unsupervised Bilingual Term Extraction methods Using Phrase Tables. *Proc. MT Summit XIII, Xiamen*
- Ion, R., Ceașu, A., Irimia, E., 2011: An Expectation Maximization Algorithm for Textual Unit Alignment. *Proc. 4th Workshop on Building and Using Comparable Corpora (BUCC)*, Portland, USA
- Kit, Ch., 2002: Corpus Tools for Retrieving and Deriving Termhood Evidence, *Proc. 5th East Asia Forum of Terminology*
- Koehn, P., 2010: *Statistical Machine Translation*. Cambridge University Press.
- Lardilleux, A., Lepage, Y., 2009: Sampling-based multilingual alignment. *Intern. Conf. on Recent Advances in Natural Language Processing (RANLP 2009)*, Borovets, Bulgaria
- Macken, L., Lefever, E., Hoste, V., 2008: Linguistically-based sub-sentential alignment for terminology extraction from a bilingual automotive corpus. *Proc. 22nd COLING*, Manchester
- Mastropavlos, Nikos; Papavassiliou, Vassilis. (2011). Automatic Acquisition of Bilingual Language Resources. *Proceedings of the 10th International Conference on Greek Linguistics*. Komotini, Greece
- Menezes, A., Richardson, St.D., 2001: A best-first alignment algorithm for automatic extraction of transfer mappings from bilingual corpora. *Proc. ACL / DMMT*
- Morin, E., Prochasson, E., 2011: Bilingual Lexicon Extraction from Comparable Corpora Enhanced with Parallel Corpora. *Proc. BUCC*, Portland, Oregon, USA
- Och, F., Ney, H., 2004: The Alignment Template Approach to Statistical Machine Translation. *Computational Linguistics* 30,4
- Rapp, R., 1999: Automatic identification of word translations from unrelated English and German corpora. *Proc. 37th ACL*, College Park, Maryland
- Robitaille, X., Sasaki, X., Tonoike, M., Sato, S., Utsuro, S., 2006: Compiling French-Japanese Terminologies from the Web. *Proc. 11th EACL*, Trento
- Thurmair, Gr., 2003: Making Term Extraction Tools Usable. *Proc. CLT*, Dublin
- Thurmair, Gr., Aleksić, V., Schwarz, Chr., 2012: Large-scale lexical analysis. *Proc. LREC Istanbul*
- Vu, Th, Aw, A.T., Zhang M., 2008: Term Extraction Through Unithood And Termhood Unification. *Proc. IJCNLP 2008*, Hyderabad, India
- Weller, M., Gojun, A., Heid, U., Daille, B., Harastani, R., 2011: Simple methods for dealing with term variation and term alignment. *Proc TIA 2011: 9th International Conference on Terminology and Artificial Intelligence*, Paris, France
- Wolf, P., Bernardi, U., Federmann, Chr., Hunsicker, S., 2011: From Statistical Term Extraction to Hybrid Machine Translation. *Proc EAMT Leuven*
- Wong, W., Liu, W., Bennamoun, M., 2007: Determining Termhood for Learning Domain Ontologies using Domain Prevalence and Tendency, *Proc. Sixth AusDM 2007*, Gold Coast, Australia. CRPIT 70

¹⁶ Two times the average P2G of 2.1% plus the translation error rate of 3.53%

WIT³: Web Inventory of Transcribed and Translated Talks

Mauro Cettolo

Christian Girardi

Marcello Federico

FBK – Fondazione Bruno Kessler
Trento, Italy
{surname}@fbk.eu

Abstract

We describe here a Web inventory named WIT³ that offers access to a collection of transcribed and translated talks. The core of WIT³ is the TED Talks corpus, that basically redistributes the original content published by the TED Conference website (<http://www.ted.com>). Since 2007, the TED Conference, based in California, has been posting all video recordings of its talks together with subtitles in English and their translations in more than 80 languages. Aside from its cultural and social relevance, this content, which is published under the Creative Commons BY-NC-ND license, also represents a precious language resource for the machine translation research community, thanks to its size, variety of topics, and covered languages. This effort repurposes the original content in a way which is more convenient for machine translation researchers.

1 Introduction

Data play a key role in machine learning as they are the main source of information to infer parameter values of the employed mathematical model.

In statistical machine translation (SMT), learning is performed on parallel texts, i.e. documents, sentences or even fragments of sentences with their translation(s). Large amounts of in-domain parallel data are usually required to properly train translation and reordering models.

Unfortunately, parallel data are a scarce resource, which are freely available only for some language pairs and for few, very specific domains.

For example, MultiUN (Eisele and Chen, 2010) provides large parallel texts (300 million words) but for only 6 languages; Europarl (Koehn, 2005) consists of the translation into most European languages of the proceedings of the European Parliament (at most 50 million words); JRC-Acquis¹ comprises the total body of European Union law applicable to the Member States, written in 22 European languages (35 million words); other smaller parallel corpora in specific domains are included in OPUS (Tiedemann, 2009) for various languages.

On the other hand, it is unfeasible for research laboratories to cover all possible needs in terms of parallel texts by resorting to professional translators, given their high cost.

The data available at the TED website² is therefore particularly valuable for the MT community. TED is a nonprofit organization that invites “the world’s most fascinating thinkers and doers [...] to give the talk of their lives”. The site makes available under the Creative Commons BY-NC-ND license the video recordings of the best TED talks, all subtitled in English and translated in many other languages by volunteers worldwide. The set of subtitles represents a precious multilingual parallel corpus since its size continuously increases (more than 900 TED talks had been collected at the end of 2011), subtitles are available in a significant number of languages (82 now, to be extended to 90 in the near future) and topics covered span the whole of human knowledge, making such data useful for any possible application domain.

In order to make this collection of talks more effectively usable by the research community, we

¹<http://optima.jrc.it/Acquis> (accessed April 16, 2012).

²<http://www.ted.com> (accessed April 16, 2012).

have developed WIT³ – an acronym for Web Inventory of Transcribed and Translated Talks –, a website hosting a ready-to-use version of this multilingual corpus, benchmarks for MT based on these data, as well as software tools to process them.

The paper is organized as follows: The TED Talks corpus is presented in Section 2, where specific subsections are devoted to the format of the files and to the sentence-level alignment; corpus statistics and an objective analysis of the difficulty of translating TED talks are also given. Section 3 describes the use of the TED Talks Corpus in the MT evaluations campaigns of the International Workshop on Spoken Language Translation (IWSLT). Finally, experimental results on baseline systems developed on several language pairs are provided in Section 4. The paper ends with the description of the WIT³ website (Section 5) and a summary (Section 6).

2 TED Talks Corpus

TED talks are mostly held in English and their videos are available through the TED website together with subtitles provided in many languages. Almost all of the talks have been translated, by volunteers, into Arabic, Bulgarian, Chinese (simplified), French, Italian, Korean, Portuguese (Brazil) and Spanish. For about 70 other languages, the number of translated talks ranges from several hundreds (e.g. such as other Dutch, German, Hebrew, Romanian) to one (e.g. Hausa, Hupa, Bislama, Ingush, Maltese). Notice that original subtitles and their translations are segmented on the basis of sound, hence the correspondence between captions and sentences is weak. It may therefore happen both that sentences are split into more consecutive captions, and that captions include sentences fragments.

For preparing parallel corpora, the raw data were first crawled, translations of the same talks were paired, captions were aligned and sentences were re-built. Each single step is described in some detail in the following subsections.

2.1 Crawling

TED talk subtitles are crawled by means of HLTWebManager (Girardi, 2011), an in-house crawler written in Java for downloading pages published on the Web in different languages. From the original HTML downloaded documents, only

subtitles and useful metadata concerning talks are kept and stored in a XML format defined by the DTD available at the WIT³ website (Section 5). For each language, a single XML file is generated which includes all talks subtitled in that language. Each talk is enclosed in tags `<file id="int">` and `</file>` and includes, among other tags:

<code><url></code>	the address of the original HTML document of the talk
<code><speaker></code>	the name of the talk speaker
<code><talkid></code>	the numeric talk identifier
<code><transcript></code>	talk subtitles split in captions
<code><date></code>	the issue date of the talk
<code><content></code>	talk subtitles

The `transcript` and `content` fields only differ in the presence of timestamps indicating splits introduced to make subtitles readable during video playing.

The `talkid` field is an integer uniquely identifying the original transcript of a talk and all its translations. Therefore, it can be used to pair translations of the same talk.

There are other tags (e.g. `description`, `keywords`, `title`, whose meaning is self-explanatory) that, providing further metadata of the talks, could be exploited for purposes like clustering, information retrieval, categorization and adaptation.

2.2 Alignment

Given a pair of languages, it is straightforward to select the talks for which subtitles are available in both languages, exploiting the `talkid` mentioned in Section 2.1. For each of such talks, the captions in the two languages are extracted from the `transcript` tags and paired in the order of appearance. A number of heuristic checks are performed in order to assess the parallelism. A whole talk is discarded if either the number of captions in the two documents differs, or the sequences of timestamps differ. Moreover, pairs of aligned captions within a talk are marked as unreliable and removed if their length ratio is an outlier, assuming a normal distribution and a 95% confidence interval.

To get an idea of the impact of filtering data with these heuristics, for the English–French collection it eliminates about 3% of the words.

Once captions are aligned, sentences are re-generated by concatenating on both sides consecutive captions until a strong punctuation mark is

	ar	bg	zh	en	fr	it	ko	pt-BR	es	de	he	nl	pl	ro	ru	tr	cs	el	hu	ja	fa	pt	vi
ar	-	1.29	1.28	1.36	1.31	1.19	1.21	1.31	1.34	0.79	0.95	0.65	0.86	1.02	0.75	0.95	0.41	0.48	0.63	0.69	0.34	0.41	0.45
bg	1.39	-	1.63	1.72	1.61	1.47	1.46	1.67	1.70	0.91	1.13	0.75	1.02	1.22	0.86	1.12	0.49	0.55	0.71	0.81	0.37	0.46	0.51
zh	0.21	0.24	-	0.25	0.24	0.21	0.22	0.24	0.25	0.13	0.17	0.11	0.15	0.18	0.13	0.17	0.07	0.08	0.11	0.12	0.06	0.07	0.08
en	1.63	1.91	1.89	-	1.92	1.70	1.70	1.93	1.98	1.08	1.35	0.92	1.20	1.43	1.01	1.35	0.59	0.65	0.85	0.94	0.48	0.56	0.62
fr	1.64	1.87	1.86	2.00	-	1.69	1.70	1.91	1.96	1.06	1.31	0.87	1.17	1.40	0.99	1.31	0.55	0.64	0.83	0.93	0.46	0.54	0.60
it	1.34	1.53	1.48	1.60	1.52	-	1.37	1.54	1.57	0.93	1.13	0.78	1.00	1.23	0.90	1.11	0.50	0.57	0.74	0.82	0.41	0.48	0.52
ko	1.01	1.12	1.11	1.17	1.12	1.01	-	1.14	1.14	0.66	0.80	0.53	0.73	0.87	0.63	0.81	0.33	0.40	0.51	0.59	0.28	0.34	0.37
pt-BR	1.51	1.78	1.76	1.85	1.76	1.58	1.60	-	1.84	0.98	1.23	0.80	1.06	1.31	0.93	1.20	0.52	0.60	0.76	0.87	0.41	0.50	0.55
es	1.56	1.84	1.80	1.92	1.82	1.63	1.62	1.86	-	1.00	1.26	0.83	1.09	1.35	0.95	1.24	0.53	0.60	0.79	0.88	0.42	0.50	0.57
de	0.91	0.96	0.94	1.03	0.97	0.95	0.91	0.97	0.98	-	0.84	0.62	0.75	0.87	0.71	0.79	0.41	0.47	0.58	0.63	0.35	0.39	0.42
he	0.85	0.93	0.91	1.00	0.93	0.89	0.86	0.95	0.96	0.65	-	0.54	0.67	0.80	0.63	0.73	0.36	0.41	0.49	0.55	0.29	0.34	0.36
nl	0.75	0.81	0.79	0.89	0.81	0.81	0.74	0.81	0.83	0.63	0.71	-	0.64	0.74	0.58	0.69	0.40	0.40	0.51	0.50	0.32	0.35	0.39
pl	0.78	0.86	0.84	0.90	0.85	0.80	0.80	0.83	0.84	0.59	0.68	0.50	-	0.70	0.55	0.67	0.30	0.35	0.47	0.51	0.26	0.30	0.34
ro	1.19	1.31	1.28	1.38	1.30	1.27	1.22	1.32	1.35	0.88	1.05	0.74	0.90	-	0.82	1.04	0.50	0.55	0.69	0.77	0.40	0.45	0.51
ru	0.75	0.80	0.80	0.85	0.80	0.80	0.77	0.81	0.82	0.62	0.71	0.50	0.61	0.71	-	0.67	0.35	0.39	0.49	0.52	0.30	0.31	0.36
tr	0.82	0.89	0.88	0.97	0.90	0.85	0.85	0.90	0.92	0.60	0.71	0.51	0.65	0.77	0.57	-	0.32	0.38	0.48	0.52	0.27	0.31	0.37
cs	0.40	0.45	0.43	0.49	0.44	0.43	0.40	0.45	0.45	0.35	0.40	0.33	0.33	0.42	0.35	0.36	-	0.25	0.32	0.29	0.21	0.22	0.23
el	0.55	0.58	0.55	0.62	0.59	0.58	0.54	0.59	0.59	0.46	0.52	0.40	0.44	0.54	0.44	0.49	0.29	-	0.39	0.38	0.26	0.27	0.28
hu	0.59	0.61	0.60	0.66	0.62	0.61	0.57	0.62	0.64	0.47	0.51	0.41	0.48	0.56	0.45	0.51	0.30	0.32	-	0.41	0.24	0.26	0.27
ja	0.13	0.14	0.14	0.15	0.14	0.14	0.13	0.14	0.14	0.10	0.12	0.08	0.11	0.12	0.10	0.11	0.05	0.06	0.08	-	0.05	0.06	0.06
fa	0.47	0.47	0.50	0.57	0.51	0.50	0.47	0.49	0.50	0.43	0.46	0.39	0.38	0.47	0.41	0.44	0.30	0.32	0.36	0.38	-	0.27	0.24
pt	0.48	0.50	0.49	0.54	0.50	0.49	0.48	0.49	0.50	0.40	0.44	0.35	0.38	0.45	0.36	0.41	0.26	0.27	0.33	0.34	0.22	-	0.24
vi	0.69	0.71	0.71	0.80	0.73	0.71	0.68	0.72	0.75	0.55	0.61	0.51	0.57	0.67	0.55	0.65	0.35	0.38	0.45	0.50	0.27	0.33	-

Table 1: The names of languages are represented by ISO 639-1 codes. Numbers refer to millions of units (untokenized words). (row, col) entries of bottom-left triangle provide the size of parallel text available for the row language side, those of upper-right triangle, for the col language side.

detected on the target side. This means that the provided parallel corpus could have: (i) lines including more sentences, as sentences can end inside captions; (ii) source lines that do not end with a strong punctuation mark.

2.3 Statistics

As of October 2011, we have collected almost 17 thousand transcripts, corresponding to translations of around 1000 English talks into 80 languages. Crawled text in all languages is left in its original format. In particular, no tokenization is applied and no word segmentation is performed for languages such as Chinese and Japanese. Hence, the reported size of corpora refer to the number of tokens, or string units, where words are possibly joined to punctuation marks and not segmented.

The distribution of translations over the 80 languages is very uneven, and consequently even more sparse among the possible 3160 language pairs.

For the three pairs from {English, French, Spanish}, parallel data reach about 2 million units. At least 1 million units can be collected for all pairs from a set of 9 languages (36 possible pairs), while at least 500K for any pair from a set of 16 languages (120 possible pairs) and at least 200 thousand for any pair from a set of 23 languages (253 possible pairs). Table 1 collects the size of paral-

lel corpus available for each pair from the 9/16/23 sublists.

2.4 Insights

How difficult is to translate TED talks? One hint comes from the scores obtained by participants at the recent evaluation campaign of IWSLT 2011 (Federico et al., 2011), which organized MT tracks based on the TED Talks data. The best reported automatic scores, computed on a single reference (see Table 7), are in fact comparable to those obtained by the best systems in the 2011 WMT evaluation (Callison-Burch et al., 2011) for the English-to-French direction on the generic news domain. This comparison is particularly significant given the similarity of experimental conditions: equivalent amount of in-domain training data and same out-of-domain training corpora. On the other hand, IWSLT scores for the translation of TED Talks from Arabic and Chinese into English are definitely lower than those obtained on news by the best systems in the last NIST evaluation,³ for the same translation directions; however, in this case the comparison is made difficult by the very different training conditions and by the use of multiple references in score computation.

³<http://www.itl.nist.gov/iad/mig/tests/mt/2009/ResultsRelease/progress.html> (accessed April 16, 2012).

Beyond using MT performance scores, the difficulty of a translation task can be weakly related to the target language model perplexity (PP) and out-of-vocabulary word rate (OOV). If such figures are computed on in-domain data, they provide hints on how intrinsically hard the task is; if they are computed on out-of-domain texts, they provide a cue on how close and potentially useful they are to improve in-domain models.

Hence, as a case study, we analyzed the English-to-French translation track of the 2011 IWSLT evaluation campaign. First, 5-gram language models (LMs) have been estimated on a number of French texts made available for training purposes, namely:

- TED: the monolingual French corpus consisting of TED talks; it is the only in-domain text
- NC: the French side of the parallel English-French News Commentary corpus
- EPPS: the French side of the parallel English-French Europarl corpus
- MultiUN: the French side of the parallel English-French MultiUN corpus.

The PP/OOV of the target side of the 2011 English-to-French test set have then been computed using each LM and collected in Table 2, which reports also the number of tokens used for training the LMs.

data	corpus size	PP	%OOV
TED	2.35M	103.8	1.67
NC	3.36M	266.8	2.83
EPPS	56.2M	200.3	1.79
MultiUN	402.8M	288.2	1.21
all	464.7M	150.8	0.72

Table 2: PP and %OOV of the IWSLT 2011 test set with respect to four 5-gram LMs estimated on in- and out-of-domain different sized corpora. Values are also reported for the LM built on the union of all corpora.

The following considerations can be drawn:

- the in-domain corpus always gets the lowest PP, even if it is the smallest one; this shows that even if the topics covered by the TED

talks are rather different, the common situation induces speakers to use a somehow similar language

- the TED talks are quite far from all the other genres of text considered here: news, proceedings of the European Parliament and resolutions of the General Assembly of the United Nations. It is quite unexpected that EPPS is closer to talks than news, but the difference in PP could be due to the size of the two corpora rather than their nature
- the OOV with respect to out-of-domain corpora seems to be mainly related to their size; it is worth noticing that the OOV can be more than halved if out-of-domain corpora are added to the in-domain one (see entry `all`), showing that the proper exploitation of available data can be anyway beneficial.

The figures just analyzed regard the evaluation set as a whole, but one could wonder if they hide large fluctuations across different talks. Table 3 provides some figures computed at talk-level both on the test set and on the TED training corpus; specifically: the mean μ of PP and OOV, their standard deviations σ , minimum and maximum values. Concerning the test set, the scores were computed on the LM estimated on text available for training. For talks in the training set, figures were computed using a 1-fold cross validation scheme.

		μ	σ	[min,max]
tst2011	PP	103.7	19.7	[68.9,132.0]
	%OOV	1.55	0.46	[0.91,2.37]
training	PP	130.2	49.3	[38.8,505.7]
	%OOV	1.76	1.04	[0.00,15.79]

Table 3: Mean, standard deviation and minimum and maximum values of PP and %OOV of talks in the test and training sets.

It results what follows:

- on average, the values of PP and OOV of talks selected for evaluation are lower than those of talks included in training data; likely, this is due to the presence of very hard talks in the training data or of very easy talks in the testing data
- the ([min,max]) ranges of observed PP and OOV values are rather large; this means

that talks can linguistically differ significantly among each others and consequently MT performance on them too.

3 WIT³ for IWSLT evaluations

The International Workshop on Spoken Language Translation is a yearly event associated with an open evaluation campaign on spoken language translation. IWSLT proposes every year challenging research tasks and an open experimental infrastructure for the scientific community working on spoken and written language translation.

In 2010 edition (Paul et al., 2010), alongside the tasks on traveling domain built on the BTEC corpus (Takezawa et al., 2007), a new challenge was introduced, that is the translation of TED talks. This became the only MT task proposed to participants in edition 2011 (Federico et al., 2011) and will remain the main task in 2012 as well.

From a translation point of view, TALK is basically a subtitling translation task, in which the ideal translation unit is a single caption as defined by the original transcript.

Concerning training data, in the 2011 edition, in addition to the roughly 2-million word parallel corpora of TED talks for each considered language pair, several out-of-domain large parallel corpora have been provided, including texts from the United Nations, European Parliament, news commentaries and the Web.

From 2012, TED Talks training data for the IWSLT evaluations will be distributed through the WIT³ website. In addition to the official tasks, the site will also release unofficial benchmarks for many other language pairs.

4 Baselines

In this section, we present results on some benchmarks that we obtained by training MT baseline systems on the available TED Talks data. The aim is to provide MT scientists with reference results that can help them in assessing their experimental outcomes. In addition to language pairs for which results were already published at IWSLT 2011, we have considered several new translation directions. The scores reported for the former will allow the assessment of the quality of our baselines with respect to state-of-the-art systems; the scores reported for the new languages can help either to understand the degree of difficulty of the task or simply to set a reference .

4.1 IWSLT 2011 MT Track Language Pairs

4.1.1 Data

Experiments were performed on data supplied by the organizers of the IWSLT 2011 evaluation campaign for the MT track,⁴ who asked participants to automatically translate talks from Arabic to English, from Chinese to English and from English to French. For developing the baselines, only texts from the TED domain were employed, i.e. no additional out-of-domain resources were used. Different preprocessings were performed depending to the language: Arabic and Chinese were segmented by means of AMIRA (Diab et al., 2004) and the Stanford Chinese Segmenter (Tseng et al., 2005), respectively; the tokenizer script released together with the Europarl corpus (Koehn, 2005) was applied to other languages.

The same partitioning of the evaluation campaign in terms of parallel training data, development (dev2010, tst2010) and test (tst2011) sets has been adopted: Tables 4 and 5 report some statistics of such texts.

text	#sent.	Arabic		English	
		W	V	W	V
parallel	90.6k	1.71M	71.1k	1.74M	42.5k
dev2010	934	19.3k	4.6k	20.1k	3.4k
tst2010	1664	30.9k	6.0k	32.0k	3.9k
tst2011	1450	26.7k	5.8k	27.0k	3.7k
text	#sent.	Chinese		English	
		W	V	W	V
parallel	107.1k	1.95M	51.9k	2.07k	46.9k
dev2010	934	21.6k	3.7k	20.1k	3.4k
tst2010	1664	33.3k	4.4k	32.0k	3.9k
tst2011	1450	24.8k	3.9k	27.0k	3.7k
text	#sent.	English		French	
		W	V	W	V
parallel	107.3k	2.07M	46.6k	2.22M	58.2k
dev2010	934	20.1k	3.4k	20.3k	3.9k
tst2010	1664	32.0k	3.9k	33.8k	4.8k
tst2011	818	14.5k	2.5k	15.6k	3.0k

Table 4: Statistics on parallel data used for setting up the baselines of IWSLT 2011 language pairs. “#sent.” stands for “number of sentences”, $|W|$ for “running words”, $|V|$ for “vocabulary size”, k and M for 10^3 and 10^6 , respectively. Counts refer to tokenized texts.

⁴http://www.iwslt2011.org/doku.php?id=06_evaluation (accessed April 16, 2012).

monolingual	#sent.	W	V
English	123.9k	2.41M	51.3k
French	111.4k	2.32M	60.3k

Table 5: Statistics on monolingual data used for training LMs of IWSLT 2011 target languages. See caption of Table 4 for the meaning of symbols.

4.1.2 Performance

The SMT baseline systems are built upon the open-source MT toolkit Moses (Koehn et al., 2007). The translation and the lexicalized reordering models were trained on parallel training data; taking into account the limited amount of training data, 4-gram LMs smoothed through the improved Kneser-Ney technique (Chen and Goodman, 1999) were estimated on monolingual texts via the IRSTLM toolkit (Federico et al., 2008). The weights of the log-linear interpolation models were optimized on the development sets `dev2010` by means of the standard MERT procedure provided within the Moses toolkit. Performance scores were computed with `MultEval`, using the implementation by (Clark et al., 2011).

Table 6 collects the %BLEU, METEOR and TER scores and their standard deviations (“case sensitive+punctuation” mode) of the baseline systems for the considered language pairs. In addition to the scores obtained for `dev2010` after the last iteration of the tuning algorithm, scores measured for the second development set (`tst2010`) and for the official test set (`tst2011`) of the evaluation campaign are reported.

	%bleu	σ	mtr	σ	ter	σ
ar-en						
dev2010	23.35	0.54	47.19	0.39	57.15	0.58
tst2010	22.10	0.44	46.09	0.35	59.38	0.50
tst2011	21.35	0.49	44.74	0.38	61.88	0.60
zh-en						
dev2010	9.53	0.38	33.96	0.37	81.71	0.95
tst2010	11.12	0.30	36.27	0.27	76.39	0.74
tst2011	13.34	0.37	38.77	0.32	65.91	0.41
en-fr						
dev2010	25.28	0.57	46.86	0.46	57.48	0.68
tst2010	28.46	0.49	49.14	0.38	51.69	0.47
tst2011	33.74	0.71	53.68	0.52	44.83	0.61

Table 6: Performance of baselines in terms of %BLEU, METEOR (mtr) and TER scores; σ stands for standard deviation. Values were computed in case sensitive mode and taking into account punctuation marks.

Although models were strictly trained on in-domain data and a quite standard configuration of Moses was used for both training and running translations, results on BLEU and TER compare well with those obtained on `tst2011` by participants at the MT track (Federico et al., 2011), whose ranges are summarized in Table 7. METEOR values seem not to be comparable, likely due to a different setup of the language dependent modules of the scorers.

tst2011	%bleu	mtr	ter
ar-en	19.56–26.32	54.66–61.10	64.65–55.81
zh-en	11.90–16.89	45.91–52.84	70.66–62.80
en-fr	34.39–37.65	24.46–27.14	45.69–41.70

Table 7: Ranges of official scores (“case sensitive+punctuation” mode) obtained by IWSLT 2011 evaluation campaign participants on the evaluation set `tst2011`.

4.2 New Language Pairs

Four new language pairs taken from Table 1 have been here considered, namely Dutch-to-English, German-to-English, German-to-Italian and English-to-Italian. These pairs as a whole cover many interesting issues: translation involving inflected languages at different extent (German, Italian, Dutch), compound words (German, Dutch), translation between non-English languages (German-to-Italian), among others. Moreover, in three cases out of four, the amount of available parallel training data is of the order of 1 million words.

4.2.1 Data

Texts used for these experiments are available at the WIT³ website. The same talks defining the IWSLT `dev2010` and `tst2010` sets were used for tuning and evaluation purposes, respectively. The rest of parallel data was used for training translation and reordering models. LMs were estimated on all talks available for each target language excluding the talks of development and test sets. Tables 8 and 9 show some statistics of collected texts after tokenization.

4.2.2 Performance

Baselines were developed exactly as for the IWSLT 2011 language pairs. Table 10 provides performance on both the tuning set `dev2010` and the evaluation set `tst2010`. In order to assess the quality of our baseline systems only on

text	#sent.	Dutch		English	
		W	V	W	V
parallel	54.6k	978k	46.0k	1.04M	32.7k
dev2010	932	18.1k	3.8k	20.2k	3.4k
tst2010	1367	24.7k	3.9k	26.2k	3.4k

text	#sent.	German		English	
		W	V	W	V
parallel	63.9k	1.16M	63.1k	1.22M	35.5k
dev2010	930	19.1k	4.2k	20.2k	3.4k
tst2010	1660	30.3k	5.2k	32.0k	3.9k

text	#sent.	German		Italian	
		W	V	W	V
parallel	56.1k	1.06M	59.8k	1.03k	48.9k
dev2010	886	18.4k	4.1k	17.1k	4.0k
tst2010	1597	30.3k	5.2k	29.3k	5.2k

text	#sent.	English		Italian	
		W	V	W	V
parallel	98.1k	1.95M	45.5k	1.80M	65.9k
dev2010	887	19.5k	3.3k	17.1k	4.0k
tst2010	1598	32.0k	3.9k	29.3k	5.2k

Table 8: Statistics on parallel data used for setting up the baselines on additional language pairs. See Table 4 for the meaning of symbols.

monolingual	#sent.	W	V
English	128.3k	2.49M	51.5k
Italian	100.8k	1.85M	67.0k

Table 9: Statistics on monolingual data used for training LMs of additional baselines. See caption of Table 4 for the meaning of symbols.

the basis of their automatic scores, we leverage the large-scale investigation reported in (Coughlin, 2011) where translations judged as acceptable or at least almost acceptable in human evaluations corresponded to %BLEU scores ranging in 20–30. Hence, our baselines provide good translations towards English, despite the quite limited amount of available parallel training data, and adequate for the English-to-Italian pair. On the contrary, the German-to-Italian direction turns out to be more difficult: this could be due either to the scarcity of training data or to the inadequacy of German pre-processing (no word decomposing), or both.

5 WIT³ Website

The WIT³ website address is:

<http://wit3.fbk.eu>

	%bleu	σ	mtr	σ	ter	σ
nl-en						
dev2010	23.31	0.63	46.63	0.48	57.96	0.63
tst2010	30.99	0.53	54.47	0.36	48.75	0.51
de-en						
dev2010	26.71	0.58	51.89	0.39	50.86	0.56
tst2010	25.88	0.46	50.57	0.34	52.13	0.47
de-it						
dev2010	13.17	0.46	28.65	0.48	69.89	0.57
tst2010	13.06	0.34	28.59	0.37	68.87	0.42
en-it						
dev2010	22.43	0.58	39.16	0.55	57.61	0.60
tst2010	22.14	0.42	39.44	0.41	56.08	0.44

Table 10: Performance of baselines on additional language pairs in terms of %BLEU, METEOR (mtr) and TER scores; σ stands for standard deviation. Values were computed in case sensitive mode and taking into account punctuation marks.

The website currently hosts the TED Talks Corpus. We expect to include other collections of talks in the future, too. Concerning the TED Talks, the corpus version will be updated on a regular basis as soon as new translations will be from the original site. For each, version the following information will be available:

XML: the set of XML files with all talks subtitled in each language

Parallel: an active web page resembling Table 1; each entry links to an archive including parallel text for training and, if any, for development and evaluation purposes

DTD: the schema defining the XML format used for storing TED talks.

The website provides the following software tools, too:

`find-common-talks.pl`: given the XML files of TED talks in two languages, it outputs the set of talkid’s (see Sections 2.1 and 2.2) for which subtitles are available in both those languages

`filter-talks.pl`: it selects from a given XML file the talks whose id’s are passed as parameter

`ted-extract-par.pl`: given a pair of XML files, it extracts the text from the transcript field (Sections 2.1, 2.2) of common talks, aligned at the caption level

`ted-extract-mono.pl`: given an XML file, it extracts the text of talks from the `transcript` field (Sections 2.1, 2.2)

`rebuild-sent.pl`: it re-builds sentences from captions (Section 2.2).

By exploiting the XML files and the supplied tools, one can extract the set of common talks for each possible language pair, as well as the monolingual text.

For many language pairs, the site will already provide training, development, and evaluation data sets, while for others, only the parallel text.

It is worth noticing that the `url` tag (see Section 2.1) allows the retrieval of the original HTML document of each talk, this way giving the possibility to users to build from scratch their own linguistic resource based on TED talks.

6 Summary

In this paper, we have described WIT³, a web inventory distributing the multilingual subtitles available under the TED Talks website. We believe, this collection represents a precious resource for the MT community given its size and its variety in terms of both languages and topics covered. In fact, more than 900 talks had been collected at the end of 2011, subtitled in up to 82 languages and spanning the whole of human knowledge.

We hope WIT³ will offer an adequate service to the research community by distributing: (i) parallel texts, benchmarks and reference MT results for some language pairs; and (ii) original formatted files and tools for processing them to let anyone build his/her own data sets for any language pair.

Acknowledgments

This work was supported by the EU-Bridge project (FP7-ICT-2011-7), which is funded by the EC under the 7th Framework Programme.

The authors also thank the three anonymous reviewers for their helpful comments.

References

Callison-Burch, C., P. Koehn, C. Monz, and O. Zaidan. 2011. Findings of the 2011 Workshop on Statistical Machine Translation. In *Proc. of WMT*, pp. 22-64, Edinburgh, Scotland, UK.

Chen, S. F. and J. Goodman. 1999. An Empirical Study of Smoothing Techniques for Language Modeling. *Computer Speech and Language*, 4(13):359-393.

Clark, J., C. Dyer, A. Lavie, and N. Smith. 2011. Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability. In *Proc. of ACL*, Portland, US-OR.

Coughlin, D. A. 2003. Correlating Automated and Human Assessments of Machine Translation Quality. In *Proc. of MT Summit*, pp. 23-27, New Orleans, US-LA.

Diab, M., K. Hacioglu, and D. Jurafsky. 2004. Automatic Tagging of Arabic Text: from Raw Text to Base Phrase Chunks. In *Proc. of HLT-NAACL: Short Papers*, pp. 149-152, Boston, US-MA.

Eisele, A. and Y. Chen. 2010. MULTIUN: A Multilingual Corpus from United Nation Documents. In *Proc. of LREC*, Valletta, Malta.

Federico, M., N. Bertoldi, and M. Cettolo. 2008. IRSTLM: an Open Source Toolkit for Handling Large Scale Language Models. In *Proc. of Interspeech*, pp. 1618-1621, Melbourne, Australia.

Federico, M., L. Bentivogli, M. Paul, and S. Stücker. 2011. Overview of the IWSLT 2011 Evaluation Campaign. In *Proc. of IWSLT*, San Francisco, US-CA.

Girardi. 2011. The HLT Web Manager. *FBK Technical Report n. 23969*. Trento, Italy. <https://wit3.fbk.eu/tools/WebManagerManual.pdf>

Koehn, P. et al. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proc. of ACL: Demo and Poster Sessions*, pp. 177-180, Prague, Czech Republic.

Koehn, P. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proc. of MT Summit*, pp. 79-86, Phuket, Thailand.

Paul, M., M. Federico, and S. Stücker. 2010. Overview of the IWSLT 2010 Evaluation Campaign. In *Proc. of IWSLT*, pp. 3-27, Paris, France.

Takezawa, T., G. Kikui, M. Mizushima, and E. Sumita. 2007. Multilingual Spoken Language Corpus Development for Communication Research. *Computational Linguistics and Chinese Language Processing*, 12(3):303-324.

Tiedemann, J. 2009. News from OPUS - a Collection of Multilingual Parallel Corpora with Tools and Interfaces. In *Recent Advances in Natural Language Processing (vol V)*, pp. 237-248. John Benjamins, Amsterdam/Philadelphia.

Tseng, H., P. Chang, G. Andrew, D. Jurafsky, and C. Manning. 2005. A Conditional Random Field Word Segmenter. In *Proc. of SIGHAN Workshop on Chinese Language Processing*, Jeju Island, Korea.

A Hybrid System for Patent Translation

Ramona Enache **Cristina España-Bonet** **Aarne Ranta** **Lluís Màrquez**
Dept. of Computer Science and Engineering TALP Research Center
Chalmers University of Technology LSI Department
University of Gothenburg Universitat Politècnica de Catalunya
{enache, aarne}@chalmers.se {cristinae, lluism}@lsi.upc.edu

Abstract

This work presents a HMT system for patent translation. The system exploits the high coverage of SMT and the high precision of an RBMT system based on GF to deal with specific issues of the language. The translator is specifically developed to translate patents and it is evaluated in the English-French language pair. Although the number of issues tackled by the grammar are not extremely numerous yet, both manual and automatic evaluations consistently show their preference for the hybrid system in front of the two individual translators.

1 Introduction

The predominant core of machine translation (MT) systems has been changing through the years. From the very beginnings in the 50s where only dictionary-based MT systems existed, the technology evolved towards rule-based systems (RBMT). Later in the 90s the everyday more powerful computers allowed to develop empirical translation systems. Recently a type of empirical system, the statistical one (SMT), has become a widely used standard for translation. At this point the two main paradigms, RBMT and SMT, coexist with their strengths and weaknesses. Luckily these strengths and weaknesses are complementary and current efforts are being made to hybridise both of them and develop new technologies. A classification and description of hybrid translation can be found in (Thurmair, 2009).

In general RBMT provides high precision, due to an analysis of the text, but has limited coverage

and a considerable amount of effort and linguistic knowledge is required in order to build such a system. On the other hand, SMT can achieve a huge coverage and is good at lexical selection and fluency but has problems in building structurally and grammatically correct translations.

Hybrid MT (HMT) is an emerging and challenging area of machine translation, which aims at combining the known techniques into systems that retain the best features of their components, and reduce the disadvantages displayed by each of the methods when used individually.

This work presents a hybrid translation system specifically designed to deal with the translation of patents. The language of patents follows a formal style adequate to be analysed with a grammar, but at the same time uses a rich and particular vocabulary adequate to be gathered statistically. We focus on the English-French language pair so that the effects of translating into a morphologically rich language can be studied.

With respect to the engine, a grammar-based translator is developed to assure grammatically correct translations. We extend GF (Grammatical Framework, Ranta (2011)) and write a new grammar for patent translation. The SMT system that complements the RBMT is based on Moses (Koehn et al., 2007). This system works on two different levels. First, it is used to build the parallel lexicon of the GF translator on the fly. Second, it is the top level decoder that takes the final decision about which phrases should be used.

In the following Section 2 describes recent work both in patent translation and hybrid systems. Section 3 explains our hybrid system and Section 4 evaluates its performance. Finally, Section 5 summarises the work and outlines possible lines to follow.

2 Related work

This work tackles two topics which are lately attracting the attention of researchers, patent translation and hybrid translators.

The high number of patents being registered and the necessity for these patents to be translated into several languages are the reason so that important efforts are being made in the last years to automate its translation between various language pairs. Different methods have been used for this task, ranging from SMT (Ceausu et al., 2011; España-Bonet et al., 2011a) to hybrid systems (Ehara, 2007; Ehara, 2010). Besides full systems, various components associated to patent translation are being studied separately (Sheremetyeva, 2003; Sheremetyeva, 2005; Sheremetyeva, 2009).

Part of this work is being done within the framework of two European projects, PLuTO (Patent Language Translations Online¹) and MOLTO (Multilingual Online Translation²). PLuTO aims at making a substantial contribution to patent translation by using a number of techniques that include hybrid systems combining example-based and hierarchical techniques. On the other hand, one of MOLTO's use cases aims at extending a grammar-based translator with an SMT to gain robustness in the translation of patents. This paper is carried out within MOLTO.

HMT is not only useful in this context but is being applied in different domains and language pairs. Besides system combination strategies, hybrid models are designed so that there is one leading translation system assisted or complemented by other kinds of engines. This way the final translator benefits from the features of all the approaches. A family of models are based on SMT systems enriched with lexical information from RBMT (Eisele et al., 2008; Chen and Eisele, 2010). On the other side there are the models that start from the RBMT analysis and use SMT to complement it (Habash et al., 2009; Federmann et al., 2010; España-Bonet et al., 2011b).

Our work can be classified in the two families. On the one hand, SMT helps on the construction of the RBMT translator but, on the other hand, there is the final decoding step to integrate translations and complete those phrases untranslated by RBMT. We use GF as rule-based system.

GF is a type-theoretical grammar formalism,

¹<http://www.pluto-patenttranslation.eu/>

²<http://www.molto-project.eu/>

mainly used for multilingual natural language applications. Grammars in GF are represented as a pair of an *abstract syntax* –an interlingua that captures the semantics of the grammar on a language-independent level, and a number of *concrete syntaxes* –representing target languages. There are also two main operations defined, *parsing* text to an abstract syntax tree and *linearising* trees into raw text. In this way one can translate between two target languages of the same multilingual grammar, by combining parsing and linearization.

The GF resource library (Ranta, 2009) is the most comprehensive grammar for dealing with natural languages, as it features an abstract syntax which implements the basic syntactic operations such as predication and complementation, and 20 concrete syntax grammars corresponding to natural languages. This layered representation makes it possible to regard multilingual GF grammars as a RBMT system, where translation is possible between any pair of languages for which a concrete syntax exists. However, the translation system thus defined is first limited by the fixed lexicon defined in the grammar, and secondly by the syntactic constructions that it covers. For this reason, GF grammars have a difficult task in parsing free text. There is some recent work on parsing the Penn Treebank with the GF resource grammar for English (Angelov, 2011), whereas the current work on patent translation is the first attempt to use GF for parsing un-annotated free text.

3 HMT system

The patent translator is a hybridisation between rule-based and statistical techniques. So, the final system is not only a combination of two different engines but the subsystems also mix different components. We have developed a GF translator for the specific domain that uses an in-domain SMT system to build the lexicon; an SMT system is on top of it to translate those phrases not covered by the grammar. In the following we describe the individual translators and the data used for their development.

3.1 Corpus

A parallel corpus in English and French has been gathered from the corpus of patents given for the CLEF-IP track in the CLEF 2010 Conference³. These data are an extract of the MAREC corpus,

³<http://clef2010.org/>

containing over 2.6 million patent documents pertaining to 1.3 million patents from the European Patent Office⁴ (EPO). Our parallel corpus is a subset with those patents with translated claims and abstracts into the two languages. From this first subset we selected those patents that deal with the biomedical domain.

The final corpus built this way covers 56,000 patents out of the 1.3 million. That corresponds to 279,282 aligned parallel fragments extracted from the claims. A fragment is the minimum aligned segment in the two languages, so, it is shorter than a claim and, consequently, shorter than a sentence. The length of the fragments is variable and depends on the aligned units that can be extracted from the xml mark-up within the patent such as paragraph tags for example. Two small sets for development and test purposes have also been selected with the same restrictions: 993 fragments for development and 1008 for test.

3.2 In-domain SMT system

The first component is a standard state-of-the-art phrase-based SMT system trained on the biomedical domain with the corpus described in Section 3.1. Its development has been done using standard freely available software. A 5-gram language model is estimated using interpolated Kneser-Ney discounting with SRILM (Stolcke, 2002). Word alignment is done with GIZA++ (Och and Ney, 2003) and both phrase extraction and decoding are done with the Moses package (Koehn et al., 2006; Koehn et al., 2007). Our model considers the language model, direct and inverse phrase probabilities, direct and inverse lexical probabilities, phrase and word penalties, and a non-lexicalised reordering. The optimisation of the weights of the model is trained with MERT (Och, 2003) against the BLEU (Papineni et al., 2002) evaluation metric.

A wider explanation of this system, the pre-process applied to the corpus before training the system and a deep evaluation of the translations can be found in España-Bonet et al. (2011a).

3.3 GF system

As explained in Section 2, the extension of GF to a new domain implies the construction of a specialised grammar that expands the general resource grammar. Since in our case of applica-

⁴<http://www.epo.org/>

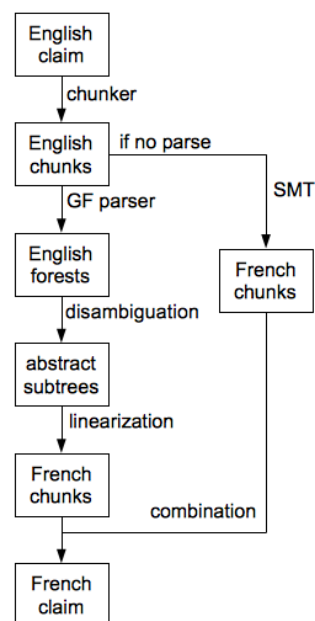


Figure 1: Architecture of the GF translation system.

tion we are far from a close and limited domain, some probabilistic components are also necessary. The general architecture is illustrated by Figure 1. A GF grammar-based system alone cannot parse most patent sentences. Consequently, the current translation system aims at using GF for translating patent chunks, and assemble the results in a later phase.

As a pre-process, claims are tagged with part-of-speech (PoS) with Genia (Tsuruoka et al., 2005), a PoS tagger trained on the biomedical domain. From the PoS-tagged words only the ones labelled as nouns, adjectives, verbs and adverbs are kept, since the GF library already has an extensive list of functional parts of speech such as prepositions and conjunctions. We use the extensive GF English lexicon⁵ as a lemmatiser for the PoS-tagged words, so that one can build their correspondent abstract syntax entry. Moreover, all the inflection forms of a given word are obtained from the same resource.

This process is made online. For every sentence to translate, the lexicon is enlarged with the corresponding vocabulary. The French version of the lexicon is built by translating the individual entries from the English lexicon (all inflection forms) with the SMT individual system trained on the patent

⁵The GF English lexicon is based on the Oxford Advanced Learner's Dictionary, and contains around 50,000 English words.

corpus. The French translations are lemmatised with an extensive GF French lexicon, based on the large morphological lexicon Morphalou (Romary et al., 2004) in order to get their inflection table. The part-of-speech is assumed to be the same as in the English counterpart.

When this procedure is applied on the test set, the part-of-speech tagger is able to find 2,013 lexicon entries. However, due to part-of-speech mismatching or to the fact that a given word was not found in the SMT lexical table, 43.81% of the entries could not be translated to French.

In order to increase the coverage of the final GF translation, the grammar is adapted to deal with chunks instead of with full sentences. So, the source text is chunked into noun phrases (NP), adjective phrases (AP), adverbial and prepositional phrases (PP), relative pronouns (RP) and verb phrases (VP). Other kinds are ignored.

Some technical details have to be taken into account in order to build the patents grammar for chunks. Whereas NPs can be translated directly, a VP, RP or AP needs to have an NP to agree with, otherwise the GF grammar cannot know which linearisation form to choose. For NP and PP which can be translated independently, a mapping into corresponding GF categories is defined, whereas for VP, RP and AP, their GF mapping requires an NP in order to build their correspondent linearisation. If the required NP is not found, the chunk is sent to the SMT. Also, the VP category from the English and French GF resource grammars is implemented as a discontinuous category, so that it can handle discontinuous constituents in English and clitics in French. The patent grammar uses a category built on top of VP, which represents the flattened version of a VP, with all the constituents combined.

Because the syntactical structure of chunks is important in this case, a post-processing step is needed. This is meant to ensure that the PoS-tagging is consistent and that certain aspects captured in the grammar can be properly reflected in the claims. One can see the importance of this step with an example.

Ex1 *The use of claim 1, wherein said use is intramuscular.*

In the previous example, “said”, a frequent used word in patent claims, acts as a definite article, whereas Genia tags it as a verb and therefore is

Word	PoS Genia	Chunk Genia	PoS Final	Chunk Final
the	DT	B-NP	DT	B-NP
use	NN	I-NP	NN	I-NP
of	IN	B-PP	IN	I-NP
claim	NN	B-NP	NN	I-NP
1	CD	I-NP	CD	I-NP
,	,	O	,	O
wherein	IN	B-PP	RP	B-RP
said	V	B-VP	DT	B-NP
use	NN	B-NP	NN	I-NP
is	VBZ	B-VP	VBZ	B-VP
intramuscular	JJ	B-ADJP	JJ	I-VP
.	.	O	.	O

Table 1: Chunk detection for the example sentence Ex1.

it not merged with the following noun into a noun phrase. Moreover, the relative pronoun “wherein” is labelled as an adverb or noun phrase. The post-processing process updates the tags of certain entries and the tag of the following word, when needed.

Table 1 shows how the original tagging from Genia is converted into the correct GF parse chunks: *the use* (NP), *of claim 1* (PP), *wherein* (RP), *said use* (NP), *is intramuscular* (VP). As one can notice, chunks are merged when needed, like for the PP *of claim 1*, where the preposition was merged with the NP into a single chunk. The same goes for the VP chunk, as it is aimed to combine two-placed verbs or copulas with their objects before parsing.

GF parses the corresponding English chunks to obtain a forest of abstract syntax trees. In order to disambiguate among the possible options, all of them are linearised, looked up in the French corpus and the most frequent linearisation is kept as the best translation.

The translation sequence is done from left to right, so that the last-occurring NP is retained, and is used to make the agreement with VP, RP or AP. If no such NP can be found, or if the GF grammar is not capable to parse the one indicated by the chunker, the current chunk is passed to the SMT. In the working example, this is not necessary, and GF grammar alone obtains a translation for the full sentence:

1. *the use* → “l’ utilisation” (NP)
2. *of claim 1* → “selon la revendication 1” (PP)
3. *wherein* → “dans laquelle” (RP agreeing with “l’ utilisation”)

4. *said use* → “ladite utilisation” (NP)

5. *is intramuscular* → “est intramusculaire” (VP agreeing with “*ladite utilisation*”)

Finally, chunks are combined together with the punctuation marks, other non-included elements and untranslated chunks in the same order as in the source language.

3.4 Top SMT layer

The grammar-based translator already makes use of the SMT system trained on patents to translate the GF English lexicon. This way, the vocabulary is disambiguated towards the biomedical domain, but still there are non-parseable chunks with unknown vocabulary in the lexicon that cannot be translated using the grammar.

To gain robustness in the final system, the output of the GF translator is used as *a priori* information for a higher level SMT system. The SMT baseline is fed with phrases which are integrated in two different ways. In both cases SMT leads the translation since it is the system that chooses the final reordering of the translation, GF constraints parts of the translation.

Hard Integration (HI): Phrases with GF translation are forced to be translated this way. The system can reorder the chunks and translates the untranslated chunks, but there is no interaction between GF and pure SMT phrases.

Soft Integration (SI): Phrases with GF translation are included in the translation table with a certain probability so that the phrases coming from the two systems interact. Probabilities in the SMT system are estimated from frequency counts in the usual way; the probabilities in the GF system are a fixed value in the interval [0, 1] for all the phrases. This probability is given to the chunk translation pair as a whole, so when competing with SMT translations that have four translation probabilities (phrase-to-phrase and word-to-word in the two directions) the probability mass is divided among them to combine the systems in the translation table. Notice that a probability of one for a phrase does not imply a sure translation not only because of this, but also because at the end, the language model chooses the translation.

4 Results and discussion

The complete hybrid system and the individual components introduced in Section 3 are evaluated

	GF	SMT
NP	2,366 (14.9%)	2,199 (13.8%)
VP	275 (1.7%)	1,302 (8.2%)
AP	1,960 (12.3%)	1,935 (12.2%)
RP	648 (4.1%)	86 (0.5%)
Other	–	5,099 (32.0%)
<i>Total</i>	<i>5,301 (33.3%)</i>	<i>10,621 (66.7%)</i>

Table 2: Number and percentage of individual chunks translated by the HI system.

on the patents test set both automatic and manually.

After the pre-process, the test set is divided in 15,922 chunks. From these chunks 33.3% can be translated using the GF patents grammar, and the remaining 66.7% must be passed to the SMT system. Table 2 shows the concrete percentages for every kind of chunk. Notice that GF only is designed to deal with the four most frequent types of chunks, and punctuation and conjunctions for example are ignored by GF. For these majority categories, GF can handle half of NP and AP, almost all RP but only 17.4% of VP.

There are several reasons why GF cannot translate the chunks. In 18.3% of the cases the chunks could not be parsed by the GF English grammar. When parsed, 15.5% of the chunks could not be translated due to missing words in the bilingual lexicon and to a lesser extent 1.1% could not be translated because of the missing information about agreement. 31.3% of the chunks are labelled as *Other* (punctuation marks, item markers, etc.) and ignored by GF.

Splitting the sentences in chunks proved to be crucial for the final translation. 84.7% of the fragments to be translated contained at least one chunk that could not be parsed by the English grammar, and even more, 93.1% of the fragments contained at least one chunk that could not be translated. So, the coverage of a GF translation at sentence level would be of only 6.9%. At chunk level the coverage increases up to 33.3%.

Still this limited coverage cannot compete with that of a statistical system. Table 3 reports an automatic evaluation using several lexical metrics for both GF and SMT individual systems (top rows). This set of metrics is a subset of the metrics available in the Asiya evaluation package (Giménez and Márquez, 2010). For all the metrics the SMT sys-

	WER	PER	TER	BLEU	NIST	GTM-2	MTR-pa	RG-S*	ULC
GF	60.96	50.08	58.90	26.56	5.57	22.74	38.76	29.00	16.17
SMT	27.03	17.50	25.32	63.18	9.99	44.58	71.64	72.65	67.14
HI	33.56	21.95	31.24	55.88	9.24	38.81	67.30	67.80	58.84
SI1.0	26.76	17.39	25.10	63.56	10.02	44.86	71.96	72.89	67.56
SI0.5	26.63	17.32	25.02	63.60	10.03	44.84	71.94	72.93	67.60
SI0.0	27.08	17.48	25.36	63.15	9.99	44.54	71.60	72.66	67.11

Table 3: Automatic evaluation of the baselines and hybrid systems.

	SMT	Tied	SI0.5
Tester1	4	9	10
Tester2	3	13	7
Tester3	2	17	4
Tester4	6	5	12
Total	15	44	33

Table 4: Manual evaluation of the 23 different sentences from a random subset of 100 sentences.

tem beats the GF one in a significant way. This is mainly due to the coverage, SMT is able to translate the whole sentence which is not the case of GF. However, GF is able to deal with some grammatical issues that cannot be recovered statistically. The most evident example is agreement in gender and number. Contrary to English, French adjectives and nouns agree in gender and number and relative pronouns agree with their relative. This is taken into account by construction in GF so that mistaken SMT translations such as “le médicament séparée” is correctly translated as “le médicament séparé” (*the separate medication*) or “composition pharmaceutique selon la revendication 1, dans lequel” is correctly translated as “composition pharmaceutique selon la revendication 1, dans laquelle” (*the pharmaceutical composition of claim 1, wherein*).

These are minor details from the point of view of the lexical evaluation metrics however, they make a difference to the reader. Although in few occasions the understanding of the sentence is compromised because of the lack of agreement, the fluency of the output is not harmed.

Therefore we incorporate these well-formed translations into the SMT system. A hard integration of the translations does not allow them to interact. GF translations are always used and the statistical decoder reorders them and completes the

translation with its own phrase table. This system is named HI in Table 3. Results are below those of the SMT system because the system is being forced to use the high quality translations together with translations of elements not considered. Just to give an example, GF will highly benefit from incorporating a grammar to deal with compounds and numbers. Currently these elements typical of the domain are not specifically approached.

A softer integration of the translations is done by the family of systems denoted by SI in Table 3. In this case, GF translations are given a probability which ranges from null to one with the same value given to all the phrases. Several experiments have been carried out for different values in the interval. We show in the bottom rows of Table 3 just three of them: 0, 0.5 and 1. Relative probabilities between the systems result not to be as important as the fact of allowing the interaction.

The combination of all the phrases improves the translations according to all the lexical metrics considered. There is an increment of 0.42 points of BLEU, 0.30 of TER and 0.46 of ULC, an uniform linear combination of 13 variants of the metrics considered. Improvements are moderate because of two reasons. First, SMT translations are already good for a start. Second, the amount of issues that GF handles are limited to be reflected on automatic metrics.

We have conducted a manual evaluation of the translations. To do this, 100 sentences have been randomly selected and four evaluators have been asked to indicate the grammatically most syntactically correct translation between two options: the SMT translation and the SI0.5 hybrid translation. The main aspects that we evaluated were correct agreement and properly inflected words.

For the whole testing corpus, 78.47% of the sentences were identically translated by the SMT and HMT. For our manually tested corpus, we only in-

spected the 23 sentences where the systems had a different output. The results can be seen in Table 4. The hybrid system is better than the SMT one according to the four evaluators, and the improvements come from discrepancies in gender, number and agreement. The SMT translations were preferred in the cases where the hybrid translation failed to translate certain words, so that the final claim has a visible hole –which makes it syntactically incorrect.

Figure 2 shows an example sentence where these features are observed. GF is doing the gender agreement between noun and adjective correctly (“séparée” vs. “séparé”) but is not able to translate the full sentence (“at the same time as”). The two hybrid systems in this case are able to construct the correct translation which coincides with the reference.

5 Conclusions and future work

This work presents a HMT system for patent translation. The system exploits the high coverage of statistical translators and the high precision of GF to deal with specific issues of the language.

At this moment the grammar tackles agreement in gender, number and between chunks, and re-ordering within the chunks. Although the cases where these problems apply are not extremely numerous both manual and automatic evaluations consistently show their preference for the hybrid system in front of the two individual translators.

The coverage of the grammar can be extended in order to deal with more typical structures present in patent documents. The coverage of VP is particularly low because of the missing verbs from the French lexicon and the syntactically complex verb phrases –such as cascades of nested verbs, which are not handled by the patents grammar yet. Also, a grammar to translate compounds will be included as they are a significant part of the biomedical documents. Moreover, the grammar component can be extended to handle the ordering at sentence level besides of the reordering within the chunks. This is specially interesting to deal with languages like German where the structure of the sentence is different from the structure in English for example.

The previous improvements will increase the number of chunks that can be parsed by the grammar; in order to increase the percentage of translations it is also necessary to improve the lexicon building procedure. An obvious improvement

would be a bilingual dictionary of idioms, so that the translation would not just map word-to-word, but also phrase-to-phrase.

Finally, we plan to implement another version of the hybrid system where GF grammars are applied at an later stage –after the English chunks are translated into French by the SMT system. The GF grammars will be used to restore the agreement for chunks like VP, RP and AP, like before. The main difference is that due to an earlier use of SMT, one can capture idiomatic constructions better, and use GF just in the end for improving syntactic correctness.

Acknowledgements

This work has been partially funded by the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement number 247914 (MOLTO project, FP7-ICT-2009-4-247914).

References

- Angelov, Krasimir. 2011. *The Mechanics of the Grammatical Framework*. Ph.D. thesis, Chalmers University of Technology, Gothenburg, Sweden.
- Ceausu, A., J. Tinsley, A. Way, J. Zhang, and P. Sheridan. 2011. Experiments on Domain Adaptation for Patent Machine Translation in the P LuTO project. *Proceedings of the 15th Annual Conference of the European Association for Machine Translation (EAMT 2011)*.
- Chen, Y. and A. Eisele. 2010. Hierarchical hybrid translation between english and german. In Hansen, Viggo and Francois Yvon, editors, *Proceedings of the 14th Annual Conference of the European Association for Machine Translation*, pages 90–97. EAMT, EAMT, 5.
- Ehara, Terumasa. 2007. Rule based machine translation combined with statistical post editor for japanese to english patent translation. *MT Summit XI Workshop on patent translation, 11 September 2007, Copenhagen, Denmark*, pages 13–18.
- Ehara, Terumasa. 2010. Statistical Post-Editing of a Rule-Based Machine Translation System. *Proceedings of NTCIR-8 Workshop Meeting, June 1518, 2010*, pages 217–220.
- Eisele, A., C. Federmann, H. Saint-Amand, M. Jellinghaus, T. Herrmann, and Y. Chen. 2008. Using mooses to integrate multiple rule-based machine translation engines into a hybrid system. In *Proceedings of the Third Workshop on Statistical Machine Translation, StatMT ’08*, pages 179–182.

GF	Une utilisation selon la revendication 3, dans laquelle le médicament séparé est administré at the same time as...
SMT	Utilisation selon la revendication 3, dans laquelle le médicament séparée est administré en même temps que...
HI	Une utilisation selon la revendication 3, dans laquelle le médicament séparé est administré en même temps que...
SI0.5	Utilisation selon la revendication 3, dans laquelle le médicament séparé est administré en même temps que...
Ref.	Utilisation selon la revendication 3, dans laquelle le médicament séparé est administré en même temps que...

Figure 2: Example where GF translates with the correct gender of the adjective and the SMT completes the untranslated words.

- España-Bonet, C., R. Enache, A. Slaski, A. Ranta, L. Màrquez, and M. González. 2011a. Patent translation within the molto project. In *Proceedings of the 4th Workshop on Patent Translation, MT Summit XIII*, pages 70–78, Xiamen, China, sep.
- España-Bonet, C., G. Labaka, A. Díaz de Ilaraza, L. Màrquez, and K. Sarasola. 2011b. Hybrid machine translation guided by a rule-based system. In *Proceedings of the 13th Machine Translation Summit*, pages 554–561, Xiamen, China, sep.
- Federmann, C., A. Eisele, Y. Chen, S. Hunsicker, J. Xu, and H. Uszkoreit. 2010. Further experiments with shallow hybrid mt systems. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 77–81, July.
- Giménez, Jesús and Lluís Màrquez. 2010. Asiya: An Open Toolkit for Automatic Machine Translation (Meta-)Evaluation. *The Prague Bulletin of Mathematical Linguistics*, (94):77–86.
- Habash, N., B. Dorr, and C. Monz. 2009. Symbolic-to-statistical hybridization: extending generation-heavy machine translation. *Machine Translation*, 23:23–63.
- Koehn, Philipp, Wade Shen, Marcello Federico, Nicola Bertoldi, Chris Callison-Burch, Brooke Cowan, Chris Dyer, Hieu Hoang, Ondrej Bojar, Richard Zens, Alexandra Constantin, Evan Herbst, and Christine Moran. 2006. Open Source Toolkit for Statistical Machine Translation. Technical report, Johns Hopkins University Summer Workshop. <http://www.statmt.org/jhuws/>.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch Mayne, Christopher Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Annual Meeting of the Association for Computational Linguistics (ACL), Demonstration Session*, pages 177–180, Jun.
- Och, Franz Josef and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.
- Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proc. of the Association for Computational Linguistics*, Sapporo, Japan, July 6-7.
- Papineni, Kishore, Salim Roukos, Todd Ward, and Weijing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the Association of Computational Linguistics*, pages 311–318.
- Ranta, Aarne. 2009. The GF resource grammar library. *Linguistic Issues in Language Technology*, 2(1).
- Ranta, Aarne. 2011. *Grammatical Framework: Programming with Multilingual Grammars*. CSLI Publications, Stanford. ISBN-10: 1-57586-626-9 (Paper), 1-57586-627-7 (Cloth).
- Romary, Laurent, Susanne Salmon-Alt, and Gil Francopoulo. 2004. Standards going concrete: from lmf to morphalou. In *Proceedings of the Workshop on Enhancing and Using Electronic Dictionaries, ElectricDict '04*, pages 22–28, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Sheremetyeva, Svetlana. 2003. Natural language analysis of patent claims. In *Proceedings of the ACL-2003 workshop on Patent corpus processing - Volume 20, PATENT '03*, pages 66–73, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Sheremetyeva, Svetlana. 2005. Less, Easier and Quicker in Language Acquisition for Patent MT. *MT Summit X, Phuket, Thailand, September 16, 2005, Proceedings of Workshop on Patent Translation*, pages 35–42.
- Sheremetyeva, Svetlana. 2009. On Extracting Multiword NP Terminology for MT. *EAMT-2009: Proceedings of the 13th Annual Conference of the European Association for Machine Translation*, ed. Lluís Màrquez and Harold Somers, 14-15 May 2009, Universitat Politècnica de Catalunya, Barcelona, Spain, pages 205–212.
- Stolcke, A. 2002. SRILM – An extensible language modeling toolkit. In *Proc. Intl. Conf. on Spoken Language Processing*.
- Thurmair, G. 2009. Comparing different architectures of hybrid machine translation systems. In *Proc MT Summit XII*.
- Tsuruoka, Y., Y. Tateishi, J.D. Kim, T. Ohta, J. McNaught, S. Ananiadou, and J. Tsujii. 2005. Developing a robust part-of-speech tagger for biomedical text. In Bozanis, P. and eds. Houstis, E.N., editors, *Advances in Informatics.*, volume 3746, page 382392. Springer Berlin Heidelberg.

Oral Session 4 – Research Papers

Hierarchical Sub-sentential Alignment with Anymalign

Adrien Lardilleux

LIMSI-CNRS

Orsay, France

adrien.lardilleux@limsi.fr

François Yvon

LIMSI-CNRS/University Paris Sud

Orsay, France

francois.yvon@limsi.fr

Yves Lepage

Waseda University, IPS

Waseda, Japan

yves.lepage@waseda.jp

Abstract

We present a sub-sentential alignment algorithm that relies on association scores between words or phrases. This algorithm is inspired by previous work on alignment by recursive binary segmentation and on document clustering. We evaluate the resulting alignments on machine translation tasks and show that we can obtain state-of-the-art results, with gains up to more than 4 BLEU points compared to previous work, with a method that is simple, independent of the size of the corpus to be aligned, and directly computes symmetric alignments. This work also provides new insights regarding the use of “heuristic” alignment scores in statistical machine translation.

1 Introduction

Sub-sentential alignment consists in identifying translation units in sentence-aligned parallel corpora, i.e. in texts in which each sentence has been matched with its translation. This task constitutes the first step in the process of training most data-driven machine translation (MT) systems (statistical or example-based). The most prominent approach nowadays is phrase-based statistical machine translation (SMT), where the core model is a translation table derived from sub-sentential mappings. This table consists in a pre-computed list of phrase¹ pairs, where each (*source*, *target*) pair is associated with a certain number of scores loosely reflecting the likelihood that *source* translates to *target*.

The problem of identifying sub-sentential mappings from parallel texts, e.g. between isolated words or n-grams of words, is well-known, and numerous proposals have been put forward to perform this task. Those methods roughly fall into two main

categories. On the one hand, the *probabilistic* approach, introduced by Brown et al. (1988), considers the problem of identifying *links* between words or groups of words in parallel sentences. This approach consists in defining a probabilistic model of the parallel corpus, the parameters of which are estimated by a global maximization process which simultaneously considers all possible associations in the corpus. The goal is to determine the best set of alignment links between all source and target words of every parallel sentence pair. The most famous representatives in this category are the IBM models (Brown et al., 1993) for aligning isolated words, which have given rise to an impressive series of variants and amendments (see e.g. (Vogel et al., 1996; Wu, 1997; Deng and Byrne, 2005; Liang et al., 2006; Fraser and Marcu, 2007; Ganchev et al., 2008), to cite a few). Generalizing word alignment models to phrase alignment proves to be a much more difficult problem, and in the view of work of Marcu and Wong (2002) and Vogel (2005), such alignments are generally produced by heuristically combining asymmetric 1-*n* word alignments (“oriented”) in both directions (Koehn et al., 2003; DeNero and Klein, 2007). Once the set of alignment links is constituted, it is possible to assign scores to each pair of segments extracted.

On the other hand, *associative* approaches (also called *heuristic* by Och and Ney (2003)), were introduced by Gale and Church (1991). They do not rely on an alignment model: in order to detect translations, they rely on independence statistical measures such as, for instance, Dice coefficient, mutual information (Gale and Church, 1991; Fung and Church, 1994), or likelihood ratio (Dunning, 1993)—see also more recent work by Melamed (2000) and by Moore (2005). Computations are generally limited to a list of association candidates precomputed using patterns and filters, for instance, by focusing exclusively on the most frequent word n-grams. In this approach, a local maximisation process is used, where each sentence is processed

© 2012 European Association for Machine Translation.

¹In this context, a phrase is a sequence of words and does not necessarily correspond to a syntactic phrase.

independently. Alignment links can then be computed, using for instance the greedy algorithm proposed by Melamed (2000) (*competitive linking*).

The probabilistic approach is the most widely used, mainly due to its tight integration with SMT, of which it constitutes a cornerstone since the introduction of IBM models (Brown et al., 1993). The two approaches have shown complementary strengths and weaknesses, as acknowledged by e.g. Johnson et al. (2007), where phrase associations extracted from word alignments are filtered out according to statistical association measures.

Anymalign, introduced in (Lardilleux and Lepage, 2009; Lardilleux et al., 2011a), aims at extracting sub-sentential associations, addressing a number of issues that are often overlooked. It can process any number of languages simultaneously, it does not make any distinction between source and target, is amenable to massive parallelism, scales easily, and is very simple to implement. *Anymalign*'s association scores have proven to produce better results than state-of-the-art methods on bilingual lexicon constitution tasks (evaluation performed by comparing word associations with reference dictionaries). However, *Anymalign*'s phrase tables are not as good as those obtained with standard methods (evaluation performed with standard MT metrics) (Lardilleux et al., 2011b).

One possible explanation for these contrasted results is that, *Anymalign* does not compute any alignment at the word or at the phrase level; instead, it directly computes translation tables along with their associated scores. Those tables have very different profiles than those obtained with probabilistic methods, mainly in terms of their n-gram distribution (Luo et al., 2011). In particular, despite recent improvements (Lardilleux et al., 2011b), the quantity of long n-grams produced remains relatively small compared with Moses's translation tables.

In this paper, we complement *Anymalign* with a simple alignment algorithm, so as to better understand its current limitations. The resulting alignments improve *Anymalign*'s phrase tables to a point where they can be used to obtain state-of-the-art results. In passing, we also propose a computationally cheap way to compute ITG alignments based on arbitrary word level association scores.

The rest of this paper is organized as follows: Section 2 describes the alignment method in detail, Section 3 presents an evaluation on machine translation tasks and an analysis of the results, and Section 4 concludes and discusses further prospects.

2 Description of the Method

In a nutshell, our method segments pairs of parallel sentences in two parts, linking the two resulting target segments with their proper translation amongst the two source segments (monotonous or inverted translation), and repeats this process recursively on the segment pairs thus obtained.

This work is strongly inspired by that of Wu (1997) and Deng et al. (2006). The former introduces *inversion transduction grammars*, which generate synchronized binary parse trees in source and target languages. This formalism models both variable-length associations at leaf (terminal) nodes, and reorderings (inversions) at any level of the parse tree. As we are only interested in computing alignment based on arbitrary lexical association scores, we will dispense here from using the full apparatus of stochastic grammars, yielding algorithms that are computationally much cheaper. The latter uses a similar concept, where more or less coarse bi-segments are extracted from non-sentence-aligned parallel texts by iteratively recursively applying a *top-down* binary segmentation algorithm. We reproduce the same approach here at the sentence level, using different local association scores.

2.1 Alignment Matrix

Our starting point are (1) a sentence-aligned bitext; and (2) a function w measuring the strength of the translation link between any *source* and *target* pair of words. Several definitions of w are possible; it is nevertheless natural to define it endogenously from word occurrences in the bitext. The scores we will first use will be obtained using *Anymalign*'s output. We will see later that they lead to better results than scores obtained using other standard measures.

In the following, the score $w(s, t)$ between a source word s and a target word t is defined as the product of the two translation probabilities $p(s|t) \times p(t|s)$, produced by *Anymalign*:

$$\begin{aligned} w(s, t) &= p(s|t) \times p(t|s) \\ &= \frac{\sum_{n=1}^N [(s, t) \in (S_n, T_n)] k_n}{\sum_{n'=1}^N [s \in S_{n'}] k_{n'}} \times \frac{\sum_{n=1}^N [(s, t) \in (S_n, T_n)] k_n}{\sum_{n'=1}^N [t \in T_{n'}] k_{n'}} \\ &= \frac{(\sum_{n=1}^N [(s, t) \in (S_n, T_n)] k_n)^2}{(\sum_{n'=1}^N [s \in S_{n'}] k_{n'}) \times (\sum_{n'=1}^N [t \in T_{n'}] k_{n'})} \end{aligned}$$

where:

- $\llbracket x \rrbracket = 1$ if x is true, 0 otherwise;
- N is the number of entries (source–target phrase pairs) in *Anymalign*'s translation table;
- S_n (resp. T_n) is the source (resp. target) part of an entry in the translation table;
- k_n is the count associated to the pair (S_n, T_n) in the translation table. This figure is not by itself

S_n	C_n	k_n
<i>pays</i>	countries	151,190
<i>pays</i>	<i>country</i>	17,717
<i>pays tiers</i>	third countries	10,865
<i>les pays</i>	countries	6,284
<i>mon pays</i>	<i>my country</i>	4,057
<i>ces pays</i>	these countries	3,742
<i>pays .</i>	<i>country .</i>	2,007
<i>état</i>	<i>country</i>	122

$$\begin{aligned}
w(\textit{pays}, \textit{country}) &= \frac{p(\textit{pays}|\textit{country}) \times p(\textit{country}|\textit{pays})}{\frac{17,717 + 4,057 + 2,007}{151,190 + 17,717 + 10,865 + 6,284 + 4,057 + 3,742 + 2,007}} \\
&= \frac{17,717 + 4,057 + 2,007}{17,717 + 4,057 + 2,007 + 122} \\
&\simeq 0.121
\end{aligned}$$

Figure 1: Computing a score between source word *pays* and target word *country* from a subset of a translation table produced by Anymalign with the French and English parts of the Europarl corpus (Koehn, 2005).

an indicator of the quality of the entry; it is just the number of times the translation pair has been produced by Anymalign (see (Lardilleux et al., 2011a) for details).

This computation is illustrated on Figure 1.

What we do here is tantamount to a very simplified version of the algorithm that is used to train standard translation models: starting with lexical associations, we derive by heuristic means an optimal (Viterbi) alignment, from which the translation tables are finally computed. Our procedure is much simpler, though, as we do not iterate the procedure (like in EM training) and directly manipulate symmetric representations at the phrase level.

2.2 Segmentation Criterion

The segmentation criterion described hereafter is inspired by the work of Zha et al. (2001) on document clustering. Their problem consists in computing the optimal joint clustering of a bipartite graph representing occurrences of terms inside a set of documents. We adapt it to the search of the best alignment between words of a source sentence and those of a target sentence.

To this end, we consider a pair of sentences (S, T) from the parallel corpus, where the source sentence S is made up of I source words and the target sentence T is made up of J target words: $S = [s_1 \dots s_I]$ and $T = [t_1 \dots t_J]$. Moreover, we consider “split” indices x and y which define a binary segmentation of the source and target sentences (the “.” symbol refers to the concatenation of word strings):

$$\begin{aligned}
S &= A.\bar{A} \quad \text{with} \quad A = [s_1 \dots s_{x-1}] \quad \text{and} \quad \bar{A} = [s_x \dots s_I] \\
T &= B.\bar{B} \quad \text{with} \quad B = [t_1 \dots t_{y-1}] \quad \text{and} \quad \bar{B} = [t_y \dots t_J]
\end{aligned}$$

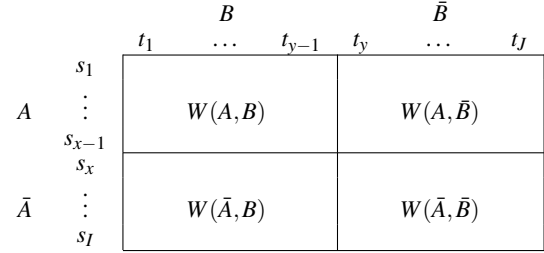


Figure 2: Schematic representation of the segmentation of a pair of sentences $S = A.\bar{A}$ and $T = B.\bar{B}$.

The choice of x and y will be guided by the sum W of the association scores between each source and target words of a block $(X, Y) \in \{A, \bar{A}\} \times \{B, \bar{B}\}$:

$$W(X, Y) = \sum_{s \in X, t \in Y} w(s, t)$$

These notations are summarized in Fig. 2.

Then, we define the total score of a segmentation:

$$\text{cut}(X, Y) = W(X, \bar{Y}) + W(\bar{X}, Y)$$

Note that $\text{cut}(X, Y) = \text{cut}(\bar{X}, \bar{Y})$. In our case, a low value indicates that the association scores between the words of X and that of \bar{Y} on the one hand, and between the words of \bar{X} and that of Y on the other hand, are low; in other words, those two blocks are unlikely to correspond to good translations, contrarily to (X, Y) and (\bar{X}, \bar{Y}) . We would thus like to identify the pair (x, y) that leads to the lowest possible value of $\text{cut}(X, Y)$.

As pointed out by Zha et al. (2001), this quantity tends to produce unbalanced segments (document clusters in their case) because of the absence of normalisation, which warrants its replacement by:

$$\text{Ncut}(X, Y) = \frac{\text{cut}(X, Y)}{\text{cut}(X, Y) + 2 \times W(X, Y)} + \frac{\text{cut}(\bar{X}, \bar{Y})}{\text{cut}(\bar{X}, \bar{Y}) + 2 \times W(\bar{X}, \bar{Y})}$$

This variant adds a density constraint on (X, Y) and (\bar{X}, \bar{Y}) , which is partially satisfied by the introduction of the denominators in the above expression. Its values are in the range $[0, 2]$.

Our problem eventually consists in determining the pair (x, y) that minimizes Ncut . Although efficient search methods exist and are commonly used in graph theory, our “graphs” (pairs of sentences) are small in practice: about 30 words per sentence in average in the Europarl corpus used in the following experiments. We thus content ourselves with determining the best segmentation through an exhaustive enumeration.

2.3 Alignment Algorithm

We can now recursively segment and align a pair of sentences. At each step, we test every possible pair (x, y) of indices in order to determine

```

procedure align( $S, T$ ) :
  if length( $S$ ) = 1 or length( $T$ ) = 1 :
    link each word of  $S$  to each word of  $T$ 
  stop procedure
   $minNcut = 2$ 
   $(X, Y) = (S, T)$ 
  for each  $(i, j) \in \{2 \dots I\} \times \{2 \dots J\}$  :
    if  $Ncut(A, B) < minNcut$  :
       $minNcut = Ncut(A, B)$ 
       $(X, Y) = (A, B)$ 
    if  $Ncut(A, \bar{B}) < minNcut$  :
       $minNcut = Ncut(A, \bar{B})$ 
       $(X, Y) = (A, \bar{B})$ 
  align( $X, Y$ )
  align( $\bar{X}, \bar{Y}$ )

```

Figure 3: Recursive alignment algorithm.

the lowest Ncut. The worst case happens when the matrix is cut in the most unbalanced possible way; the complexity of the algorithm is thus cubic ($O(I \times J \times \min(I, J))$) in the length of the input sentences. Using a greedy strategy only delivers sub-optimal solutions, yet it does so much faster than exact ITG parsing, which is cubic in the product $I \times J$ (Wu, 1997). For a given pair (x, y) , two values are computed: one corresponds to a monotonous alignment ($Ncut(A, B)$) and the other one to an inversion of the two segments ($Ncut(A, \bar{B})$). We then apply the process recursively on each of the two segment pairs that correspond to the minimal Ncut. It ends when one of the segments contains only one word and produces $1-n$ or $n-1$ alignments. In this approach, all words are aligned. By considering different stopping criteria, eg. based on thresholds on Ncut, variants of the algorithm are readily obtained, which enable to balance the granularity of the alignment with its precision, by choosing to build larger and safer blocks ($m-n$ alignments) instead of smaller and less sure ones. We leave this for future work. Figure 3 presents the complete algorithm, and Fig. 4 illustrates the process on two actual examples. In the following, we refer to this algorithm under the name of “Cutnalign.”

The algorithm itself is independent of the size of the parallel corpus to align, because each sentence pair is processed independently. Aligning a corpus can thus easily be made parallel: the total running time is divided by the number of available processors. Another advantage is that the alignments produced are symmetric during the whole process, contrary to more widely spread models such as IBM models that produce better result when run in both translation directions and their outputs combined using heuristics.

3 Evaluation

3.1 Description of Experiments

Our alignment method is evaluated within a phrase-based SMT system. We use the Moses toolkit (Koehn et al., 2007), and data extracted from the Europarl corpus (Koehn, 2005), in three languages: Finnish–English (agglutinating language–isolating language), French–Spanish, and Portuguese–Spanish (very close languages). For each pair, we use a training set made up of 350,000 sentence pairs (avg.: 30 words/sentence in English), and development and test sets made up of 2,000 sentence pairs each. The systems are optimized with MERT (Och, 2003). Unless otherwise specified, a lexicalized reordering model is used. Translations are evaluated using BLEU (Papineni et al., 2002) and TER² (Snover et al., 2006).

Five approaches are compared:

MGIZA++ (Gao and Vogel, 2008), implements the IBM models (Brown et al., 1993) and the HMM of Vogel et al. (1996). Integrated to Moses, it remains the reference in the domain. It is run with default settings: 5 iterations of IBM1, HMM, IBM3, and IBM4, in both directions (source to target and target to source). The alignments are then made symmetric and a translation table is produced from the alignments using Moses tools (*grow-diag-final-and* heuristic for phrase pair extraction).

Anymalign (Lardilleux et al., 2011a), used to directly build the translation tables. As this tool can be stopped at any time, its running time is set so that it runs for the same duration as MGIZA++. The same experiment is repeated by varying the length of output phrases from 1 to 4 (see (Lardilleux et al., 2011b) for details). In the following, we refer to it under the names “Anymalign-1” to “Anymalign-4.” The reordering model used in this configuration is a simple distance-based model, because Anymalign alone cannot provide the information required for a lexicalized reordering model.

Anymalign + Cutnalign: we apply the algorithm described in previous section to each of the four translation tables produced by Anymalign-1 to Anymalign-4. Although every intermediary segmentation step (all possible rectangles in Fig. 4) actually corresponds to a phrase pair that could be extracted and fit in a phrase-table, in our experiments, we only rely on *terminal alignment points*, that are then passed to the Moses toolkit to build new translation tables (using again the *grow-diag-final-and*

²Contrary to BLEU, lower scores are better.

				the	level	of	budgetary	implementation	;
le	0.037	ϵ	0.001	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ
niveau	ϵ	0.591	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ
d'	ϵ	ϵ	0.003	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ
exécution	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0.060	ϵ	ϵ
budgétaire	ϵ	ϵ	ϵ	ϵ	ϵ	0.659	ϵ	ϵ	ϵ
;	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0.287

		finally	,	what	our	fellow	citizens	are	demanding	is	the	right	to	information	.
enfin	0.607	0.001	ϵ	ϵ	ϵ	0	ϵ	ϵ	0	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ
,	0.001	0.445	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0.001	ϵ	0.001	ϵ	0.001
c'	ϵ	ϵ	0.001	ϵ	ϵ	ϵ	ϵ	0	0.036	0.001	ϵ	ϵ	ϵ	ϵ	ϵ
est	ϵ	ϵ	0.001	ϵ	ϵ	ϵ	ϵ	0	0.223	0.016	ϵ	0.001	ϵ	0.001	0.001
un	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0.005	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ
droit	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0	ϵ	ϵ	0.084	ϵ	ϵ	ϵ	ϵ
à	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0.001	ϵ	0.001	0.003	0.001	0.018	ϵ	ϵ	ϵ
l'	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0.002	0.009	ϵ	0.002	ϵ	ϵ	ϵ
information	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0.499	ϵ
que	ϵ	ϵ	0.002	ϵ	ϵ	ϵ	0.001	ϵ	0.002	0.001	ϵ	ϵ	0.001	ϵ	ϵ
réclament	0	0	ϵ	ϵ	ϵ	ϵ	ϵ	0.152	ϵ	ϵ	0	0	0	0	ϵ
nos	ϵ	ϵ	ϵ	0.171	0.004	0.001	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ
concitoyens	0	ϵ	ϵ	0.001	0.323	0.009	ϵ	ϵ	ϵ	ϵ	0	ϵ	0	0	ϵ
.	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0.001	0.001	ϵ	ϵ	ϵ	ϵ	0.954

Figure 4: Two examples of segmentation-alignment. The number in each cell corresponds to the value of the function w , with $0 < \epsilon \leq 0.001$. A null value indicates that the two words never appear together in the translation table. Alignment points retained by the algorithm, i.e. at maximum level of recursion, are in boldface. In the first example, the translation is monotonous except for the name/adjective inversion (*exécution budgétaire/budgetary implementation*), therefore most alignment links are along the diagonal. The second example, more complex, attests for the inversion of propositions inside the sentence.

heuristic). This approach yields more phrase pairs as it allows to extract together segments on both sides of a split point, e.g. *le niveau/the level*.

Simple probabilities + Cutnalign: the purpose of this configuration is to evaluate the choice of w , rather than the algorithm itself. To this end, we use a very simple association score: the probability that a source word and a target word are translations of one another (product of the two translation probabilities), where this probability is computed from their co-occurrence counts over the training corpus. The definition of w is thus the same as in Sec. 2.1, with two minor differences: (1) counts are directly computed over the training bitext; and (2) $k_n = 1, \forall n$.

Anymalign + Cutnalign / MGIZA++: This is a combination of the MGIZA++ and Anymalign+Cutnalign approaches. We do this by taking the union of the two alignment sets. In practice, we simply concatenate the two alignment files produced by the aligners, and duplicate the training bicorpus so that we end up with a new, twice as large, training bicorpus and alignment file, from which the phrase table is extracted.

In terms of runtime, although Cutnalign is currently implemented in a high-level programming language (Python) and its complexity is cubic in the

length of the sentence pairs to process, the fact that each sentence pair can be aligned independently makes it amenable to massive parallelism if numerous CPUs are available.

3.2 Results

Results are in Table 1. For each task, using the basic version of Anymalign yields worse scores than MGIZA++-based system, even though extending the phrase length reduces this gap by roughly a half, except for the Finnish–English pair. Those results are in line with (Lardilleux et al., 2011b).

Cutnalign leads to significant gains in all configurations: from 1.6 to 4.6 BLEU points (fr-en, Anymalign-1 + Cutnalign), with an average gain of 2.6 BLEU and 2.7 TER points. Anymalign + Cutnalign is still 1.1 to 1.6 BLEU points below in Finnish–English relatively to MGIZA++ but produces results of comparable quality in French–English and Portuguese–Spanish.

The “simple probabilities + Cutnalign” configuration produces intermediary quality results, generally between “basic” Anymalign and Anymalign + Cutnalign. This shows that the function w has a significant impact on the behavior of the alignment method. Assuming the function used in these experiments is one of the simplest possible, there is ample room here for improvements. Merging both phrase tables is almost always the best strategy, at

Task	System	BLEU (%)	TER (%)	Entries (millions)	Length of entries	Links	Length of extracted blocks
fi-en	MGIZA++	22.27	62.92	22.2	3.24	26	1.16
	Anymalign-1	18.68	67.30	11.8	1.87		
	Anymalign-2	17.86	68.60	4.4	2.09		
	Anymalign-3	18.06	68.13	3.0	2.32		
	Anymalign-4	18.06	68.53	2.1	2.42		
	Anymalign-1 + Cutnalign	21.14	63.74	7.7	3.26	62	1.45
	Anymalign-2 + Cutnalign	21.14	64.69	7.5	3.27	69	1.48
	Anymalign-3 + Cutnalign	20.83	64.18	7.3	3.29	73	1.50
	Anymalign-4 + Cutnalign	20.64	64.52	7.1	3.29	78	1.53
	Simple prob. + Cutnalign	19.09	67.09	5.5	3.23	74	1.78
	Anymalign-1 + Cutnalign / MGIZA++	22.66	62.45	27.0	3.24	44	1.30
	Anymalign-2 + Cutnalign / MGIZA++	22.68	62.91	26.9	3.24	47	1.31
	Anymalign-3 + Cutnalign / MGIZA++	22.73	62.82	26.8	3.24	49	1.32
	Anymalign-4 + Cutnalign / MGIZA++	22.78	62.11	26.7	3.24	52	1.33
fr-en	MGIZA++	29.65	55.25	25.6	4.29	31	1.17
	Anymalign-1	25.10	59.36	6.1	1.27		
	Anymalign-2	26.60	58.16	6.3	1.99		
	Anymalign-3	27.02	57.96	3.9	2.29		
	Anymalign-4	26.85	58.00	2.6	2.42		
	Anymalign-1 + Cutnalign	29.65	55.22	12.9	4.21	50	1.49
	Anymalign-2 + Cutnalign	29.69	55.44	13.1	4.22	48	1.48
	Anymalign-3 + Cutnalign	29.26	55.49	13.0	4.23	50	1.49
	Anymalign-4 + Cutnalign	29.16	55.46	12.8	4.23	52	1.51
	Simple prob. + Cutnalign	27.97	56.85	10.2	3.95	54	1.62
	Anymalign-1 + Cutnalign / MGIZA++	30.02	54.81	31.9	4.24	41	1.32
	Anymalign-2 + Cutnalign / MGIZA++	29.91	54.88	31.9	4.24	40	1.32
	Anymalign-3 + Cutnalign / MGIZA++	30.22	54.94	31.9	4.24	41	1.32
	Anymalign-4 + Cutnalign / MGIZA++	29.91	54.87	31.8	4.24	42	1.33
pt-es	MGIZA++	38.53	48.46	32.2	4.30	30	1.09
	Anymalign-1	35.20	50.89	5.7	1.26		
	Anymalign-2	36.80	49.60	5.9	1.99		
	Anymalign-3	36.82	49.67	3.7	2.26		
	Anymalign-4	36.96	49.80	2.4	2.37		
	Anymalign-1 + Cutnalign	37.35	49.55	17.9	4.30	50	1.32
	Anymalign-2 + Cutnalign	38.96	48.04	18.0	4.30	48	1.32
	Anymalign-3 + Cutnalign	38.55	48.40	17.7	4.31	50	1.33
	Anymalign-4 + Cutnalign	38.56	48.37	17.3	4.31	54	1.35
	Simple prob. + Cutnalign	37.71	49.04	13.9	4.09	50	1.41
	Anymalign-1 + Cutnalign / MGIZA++	38.77	48.12	37.7	4.25	40	1.20
	Anymalign-2 + Cutnalign / MGIZA++	38.69	48.39	37.9	4.25	39	1.20
	Anymalign-3 + Cutnalign / MGIZA++	38.94	48.12	37.8	4.25	40	1.20
	Anymalign-4 + Cutnalign / MGIZA++	38.82	48.18	37.8	4.25	42	1.21

Table 1: Summary of results obtained in our experiments. The first two columns (BLEU and TER) report performance in machine translation. The two middle columns display various characteristics of the translation tables: the number of entries and their length in words. The last two columns present characteristics of the alignments prior to the production of the translation table: average number of alignment links per training sentence pair and average length of the source part of minimal blocks extracted (translations of the phrases that are consistent with word alignments).

the most of much larger models.

3.3 Analysis of Alignments

One motivation for proposing this new alignment method is that Anymalign still lacks the ability to extract long n-gram translations in sufficient quantity. In this section, we study some characteristics of the alignments thus produced (see Table 1).

Regarding translation tables first, we observe that those obtained from Cutnalign contain many more entries than those produced by Anymalign alone³ (three times more in average), except for Anymalign-1 in Finnish–English. Nevertheless, they are still much smaller than tables obtained from MGIZA++, as they contain twice less entries in average. In addition, the average length of those entries is almost equal to that of those in MGIZA++’s translation tables, while those produced by Anymalign are much shorter: producing a translation table from alignment links allows to make up for the lack of long n-grams as desired.

Secondly, we study the alignment links themselves. The column “Links” of Table 1 shows that our method produces more alignment links than MGIZA++: between 1.5 and 3 times more, depending on the task. The last column gives the main reason: alignment blocks extracted by our method, i.e. rectangles obtained at maximal recursion depth, are always longer than minimum blocks obtained from MGIZA++’s alignments (+ 26% in average). Since we systematically align all source words with all target words in such a rectangle, and since all words of a sentence pair are therefore necessarily aligned, the total number of alignments produced is naturally high. This also explains the fact that the number of entries in our translation tables is always much lower than those obtained from MGIZA++, as the latter produces 0–1 alignments that are at the origin of numerous phrases extracted during the constitution of the table by Moses (*grow-diag-final* and heuristic by default) (Ayan and Dorr, 2006). Despite this, alignments produced by our method lead to state-of-the-art scores in two machine translation tasks over three in our experiments.

4 Conclusion

We have presented a sub-sentential alignment method based on a recursive binary segmentation process of the alignment matrix between a source sentence and its translation. Inspired by work on

³These tables were produced by running Anymalign for an identical amount of time in all configurations, which explains why larger values of the length parameter lead to smaller tables—see details in (Lardilleux et al., 2011b).

alignment by Wu (1997) and Deng et al. (2006) and work on document clustering by Zha et al. (2001), we have shown that despite its simplicity, this method leads to state-of-the-art results in two tasks over three in our experiments. When fed with Anymalign’s scores, it yields significant gains (up to 4.6 BLEU points in French–English) in comparison with Anymalign alone. These experiments confirm that Anymalign’s main handicap concerns the translation of long n-grams. A complementary alignment step, strictly speaking, is thus desired in order to improve its results in machine translation. The alignment method proposed here is simple, symmetric with respect to the translation direction, and the use of local computations makes it scale up easily. Many improvements are possible, amongst which the use of early stopping criteria during segmentation of the alignment matrix so as to trade alignment granularity for confidence; the use more sophisticated metrics for scoring blocks, or the exploration of richer (e.g. ternary) segmentation schemes, enabling to account for more complex linguistic constructs.

References

- Ayan, Necip Fazil and Bonnie J. Dorr. 2006. Going beyond AER: An extensive analysis of word alignments and their impact on MT. In *Proc. of Coling/ACL’06*, pages 9–16, Sydney, Australia.
- Brown, Peter, John Cocke, Stephen Della Pietra, Vincent Della Pietra, Fredrick Jelinek, Robert Mercer, and Paul Roossin. 1988. A statistical approach to language translation. In *Proc. of Coling’88*, pages 71–76, Budapest.
- Brown, Peter, Stephen Della Pietra, Vincent Della Pietra, and Robert Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.
- Dagan, Ido and Ken Church. 1994. Termight: identifying and translating technical terminology. In *Proc. of the 4th conference on Applied natural language processing*, pages 34–40, Stuttgart.
- DeNero, John and Dan Klein. 2007. Tailoring word alignments to syntactic machine translation. In *Proc. of ACL’07*, pages 17–24, Prague.
- Deng, Yonggang and William Byrne. 2005. HMM word and phrase alignment for statistical machine translation. In *Proc. of HLT/EMNLP’05*, pages 169–176, Vancouver, British Columbia, Canada, October.
- Deng, Yonggang, Shankar Kumar, and William Byrne. 2006. Segmentation and alignment of parallel text for statistical machine translation. *Natural Language Engineering*, 13(3):235–260.
- Dunning, Ted. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

- Fraser, Alexander and Daniel Marcu. 2007. Getting the structure right for word alignment: LEAF. In *Proc. of EMNLP/CoNLL'07*, pages 51–60, Prague.
- Fung, Pascale and Kenneth Church. 1994. K-vec: A new approach for aligning parallel texts. In *Proc. of Coling'94*, volume 2, pages 1096–1102, Kyōto.
- Fung, Pascale and Lo Yuen Yee. 1998. An IR approach for translating new words from nonparallel, comparable texts. In *Proc. of Coling/ACL'98*, volume 1, pages 414–420, Montreal.
- Gale, William and Kenneth Church. 1991. Identifying word correspondences in parallel texts. In *Proc. of the 4th DARPA workshop on Speech and Natural Language*, pages 152–157, Pacific Grove.
- Ganchev, Kuzman, João Graça, and Ben Taskar. 2008. Better alignments = better translations? In *Proc. of ACL'08*, pages 986–993, Columbus, Ohio.
- Gao, Qin and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57, Columbus (Ohio, USA).
- Gaussier, Éric and Jean-Marc Langé. 1995. Modèles statistiques pour l'extraction de lexiques bilingues. *Traitement Automatique des Langues*, 36(1-2):133–155.
- Johnson, Howard, Joel Martin, George Foster, and Roland Kuhn. 2007. Improving translation quality by discarding most of the phrasetable. In *Proc. of EMNLP/CoNLL'07*, pages 967–975, Prague.
- Koehn, Philipp, Franz Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proc. of HLT/NAACL'03*, pages 48–54, Edmonton.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. of ACL'07*, pages 177–180, Prague.
- Koehn, Philipp. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proc. of MT Summit X*, pages 79–86, Phuket.
- Lardilleux, Adrien and Yves Lepage. 2009. Sampling-based multilingual alignment. In *Proc. of RANLP*, pages 214–218, Borovets.
- Lardilleux, Adrien, Yves Lepage, and François Yvon. 2011a. The contribution of low frequencies to multilingual sub-sentential alignment: a differential associative approach. *International Journal of Advanced Intelligence*, 3(2):189–217.
- Lardilleux, Adrien, François Yvon, and Yves Lepage. 2011b. Généralisation de l'alignement sous-phrastique par échantillonnage. In *Proc. of TALN 2011*, volume 1, pages 507–518, Montpellier, France.
- Liang, Percy, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proc. of the HLT/NAACL'06*, pages 104–111, New York City.
- Luo, Juan, Adrien Lardilleux, and Yves Lepage. 2011. Improving sampling-based alignment by investigating the distribution of n-grams in phrase translation tables. In *Proc. of PACLIC 25*, pages 150–159, Singapore.
- Marcu, Daniel and Daniel Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proc. of EMNLP'02*, pages 133–139, Philadelphia.
- Melamed, Dan. 2000. Models of translational equivalence among words. *Computational Linguistics*, 26(2):221–249.
- Moore, Robert. 2004. On log-likelihood-ratios and the significance of rare events. In *Proc. of EMNLP'04*, pages 333–340, Barcelona.
- Moore, Robert. 2005. Association-based bilingual word alignment. In *Proc. of the ACL Workshop on Building and Using Parallel Texts*, pages 1–8, Ann Arbor.
- Och, Franz and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29:19–51.
- Och, Franz. 2003. Minimum error rate training in statistical machine translation. In *Proc. of ACL'03*, pages 160–167, Sapporo.
- Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proc. of ACL'02*, pages 311–318, Philadelphia.
- Smadja, Frank, Vasileios Hatzivassiloglou, and Kathleen McKeown. 1996. Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22(1):1–38.
- Snoover, Matthew, Bonnie Dorr, Richard Schwartz, Linea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proc. of AMTA'06*, pages 223–231, Cambridge, August.
- Vogel, Stephan, Hermann Ney, and Christoph Tillman. 1996. Hmm-based word alignment in statistical translation. In *Proc. of Coling'96*, pages 836–841, Copenhagen.
- Vogel, Stephan. 2005. PESA: Phrase pair extraction as sentence splitting. In *Proc. of MT Summit X*, pages 251–258, Phuket.
- Wu, Dekai. 1997. Stochastic inversion transduction grammar and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–404.
- Zha, Hongyuan, Xiaofeng He, Chris Ding, Horst Simon, and Ming Gu. 2001. Bipartite graph partitioning and data clustering. In *Proc. of the 10th international conference on Information and knowledge management*, pages 25–32, Atlanta.

Adjunct Alignment in Translation Data with an Application to Phrase-Based Statistical Machine Translation

Sophie Arnoult

Institute of Logic, Language
and Computation (ILLC)
University of Amsterdam
Amsterdam, the Netherlands
s.arnoult@gmail.com

Khalil Sima'an

Institute of Logic, Language
and Computation (ILLC)
University of Amsterdam
Amsterdam, the Netherlands
k.simaan@uva.nl

Abstract

Enriching statistical models with linguistic knowledge has been a major concern in Machine Translation (MT). In monolingual data, adjuncts are optional constituents contributing secondarily to the meaning of a sentence. One can therefore hypothesize that this secondary status is preserved in translation, and thus that adjuncts may align consistently with their adjunct translations, suggesting they form optional phrase pairs in parallel corpora. In this paper we verify this hypothesis on French-English translation data, and explore the utility of compiling adjunct-poor data for augmenting the training data of a phrase-based machine translation model.

1 Introduction

Phrase-Based Statistical Machine Translation (PB-SMT) (Koehn et al., 2003) exploits symmetrized word alignments (Brown et al., 1993) to form phrase pairs that capture the translation probabilities of idiomatic expressions. However, data sparsity is a major issue for phrase-based systems. It affects longer phrase pairs in particular, which are overestimated by the unsmoothed heuristic counts. Smoothing has been proposed to improve probability estimations in the phrase table (Kuhn et al., 2006; Foster et al., 2006), and minimal phrase pairs to alleviate data sparsity: see the tuples of Schwenk (2007) and the minimal translation units of Quirk and Menezes (2006). In both cases, these new units of translation are utilized in an n-gram translation model that allows to capture contextual dependencies, and their estimates are smoothed.

Morphology has also been proposed to reduce data sparsity, either by integrating morphological information into the translation model or as a preprocessing step. For instance, Nießen and Ney (2004) propose hierarchical lexicon models in a German-English system, with feature functions to integrate different levels of morphosyntactic abstraction, from full word forms to lemmas, whereas El Kholy and Habash (2010) present different morphological tokenization models for Arabic.

In this work we propose to use adjuncts to *augment and smooth training data* for machine translation. The term ‘adjunct’ is used here to refer to both clausal adjuncts and phrase modifiers, regardless of the nature of the modified category, e.g., verbal or nominal. As optional constituents that further qualify a complete clause or phrase, adjuncts can be removed from or added to a monolingual sentence without affecting its grammaticality. This idea is embodied most visibly in Tree-Adjoining Grammar (Joshi, 1983) where recursion in syntax is factored out into auxiliary lexical elements that modify initial sentences by adjunction operation. Here, we hypothesize that adjuncts, by their secondary semantic status, are likely to be preserved in translation. To our knowledge, this constitutes the first study of adjunct alignment in parallel data, though this idea is related to the Direct Correspondence Assumption (DCA) of Hwa et al. (2002). The DCA postulates that syntactic relations, e.g., between heads and arguments or between heads and modifiers, are preserved in translation. Hwa et al. (2002) project English unlabeled dependency parses on Chinese data and report 30.1% precision and 39.1% recall for the Chinese dependencies. Another related work is that of Dorr et al. (2002), which measured structural di-

vergence, in terms of predicate-argument-modifier structure, by automatically detecting regular expressions in a Spanish corpus. The detected expressions were then verified manually, and are seen as giving a lower bound on structural divergence. The authors found that 11% of sentences contained divergent structures, and 35% with relaxed regular expressions.

We measured adjunct alignment on a French-English parallel treebank, showing that English adjuncts tend to be consistent with word alignments for machine translation and to be aligned to French adjunct-like constituents. Section 2 presents criteria to identify English adjuncts in phrase-structure parses and provides alignment measures for these adjuncts into French.

If adjuncts can be paired by word alignments, they can be deleted from or inserted in translation data, thus unfolding latent translation data. The resulting data can then be used to smooth the original distribution. In this work we start by investigating the effect of adjunct deletion, which is far simpler than adjunct insertion. On the linguistic side, adjunct insertion is complicated because modifiers are subjected by their heads to lexical and syntactical constraints, e.g., verbs take adverbial modifiers and nouns take adjectival modifiers. And on the computational side, adjunct deletion can be done in the phrase-based framework, whereas adjunct insertion requires a synchronous grammar with insertion/adjunction as operation, e.g., Synchronous Tree-Adjoining Grammar, (Abeillé et al., 1990; Shieber, 2007). In section 3 we show how adjunct-pair deletion from parallel data allows us to generate more training data to smooth a PBSMT baseline. Section 4 then provides experimental results for the smoothed model. We conclude in Section 5.

2 Adjunct alignment between French and English

As an illustration for adjunct alignment, consider the sentence pair in Figure 1. The English sentence contains three adjuncts that are translated as adjuncts in the French sentence. The example shows that the paired adjuncts can be of a different syntactical nature, as well as the phrases they appear in. Here, “*governing existing vehicles*” is a verb phrase while “*pour les véhicules existants*” is a prepositional phrase; and “*there must be rules*” is only globally equivalent to “*il faut trouver des règles*”. In other words, adjunct pairing can occur

relatively independently of the syntactical realization of the involved adjuncts and of the degree of translation equivalence of the phrases they modify.

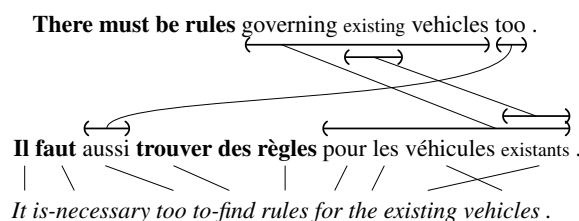


Figure 1: Example sentence pair with adjunct pairs

Conversely, adjuncts are not always preserved in translation. For instance, Example 1 presents a case of head swapping taken from (Dorr et al., 2002).

- (1) *Yo entro el cuarto corriendo*
 I enter the room running
I run into the room

There the manner of motion, i.e., ‘running’, is expressed by the verbal head in the English sentence and by a modifier in the Spanish sentence while the direction, i.e., ‘into’, is expressed by the head in Spanish and by a modifier in English. So, while (Dorr et al., 2002) investigated structural divergence in general and not only on modifiers, we can expect that adjuncts are not always translated as such in the target language.

Another limitation on adjunct alignment is not linguistic but technical. In fact, we depend on word alignments to align adjuncts into the target language. Consequently, to take the example of Figure 1, one can only know that the phrase “*pour les véhicules existants*” is paired with “*governing existing vehicles*” if the word alignments are able to align the semi-equivalent ‘*governing*’ and ‘*pour*’ properly. An unfavorable alignment in this case might align the English phrase with, e.g., “*les véhicules existants*”.

Finally, the method we follow to identify adjunct pairs in the data consists in first identifying English adjuncts, before aligning them to their French counterpart using word alignments. We identify adjuncts using a phrase-structure parser, which allows to quickly parse very large translation corpora, but does not directly annotate modifiers. Instead one can apply categorial and distributional criteria to identify constituents that are likely to be adjuncts. We present our identifica-

tion criteria in section 2.1, and adjunct alignment experiments and results in section 2.2.

2.1 Identifying adjuncts and adjunct pairs

The identification criteria for the English adjuncts are set by manually analysing the fifty first parses of the English Europarl corpus, parsed with the Charniak parser. Constituent categories that function as modifiers in most cases and given some distributional constraints are subsequently regarded as adjuncts. The identification criteria are summed up in Table 1. The tags are those of the Penn

category	parent	additional restriction
ADJP	NP	
JJ	NP	
NNx	NP	NN/NNS right sibling
VP	NP	
S	NP	
PP	≠PP	
SBAR	≠VP	
RB	≠ADVP	
ADVP		
PRN		
NP		adposed: left and right comma

Table 1: English-adjuncts identification criteria

Treebank¹, except for NNx, which stands for NN(P)(S). The English adjuncts thus identified are paired by the GIZA++ word alignments to their French counterpart. The phrase pairs that are consistent with the word alignments are then assumed to be pairs of adjuncts.

2.2 Adjunct alignment between English and French

To assess how well English adjuncts are aligned to French adjuncts, we analyzed adjunct alignment in a parallel treebank. The French treebank was obtained from the automatically annotated Europarl-section of the ‘Arboratoire’ treebank², and contains 30421 sentences and parses that roughly correspond to the beginning of the Europarl corpus. The English treebank was obtained from the English Europarl Corpus with the Charniak parser. After aligning both treebanks with the French and English corpora and with the GIZA++ word alignments trained on the whole corpus and merged with ‘grow-diag-final’, one obtains 13620 aligned parses, sentences and word alignments.

¹ftp.cis.upenn.edu/pub/treebank/doc/manual/root.ps.gz

²<http://corp.hum.sdu.dk/arboratoire.html>

For each English adjunct category, we aligned English adjuncts to their French counterparts, and measured the relative frequency of the following cases: (1) adjuncts pairs that are *not* consistent with the word alignments (nc/A)³; (2) the French counterpart could not be located in the parse ($f_?$); (3) the French counterpart is not consistent with the parse (nc/P); (4) adjuncts aligned to the empty string (f_\emptyset); (5) the French counterpart is consistent with the French parse, i.e., it corresponds to one or more *complete* constituents (c/P). Measurements are reported in Table 2, along with the average number of English adjuncts per sentence (r), and upper bounds (UB) for adjunct alignment into French.

	r	nc/A (%)	$f_?$ (%)	nc/P (%)	f_\emptyset (%)	c/P (%)	UB (%)
JJ	0.98	18.4	0.4	3.0	3.5	74.7	78.2
RB	0.31	35.1	0.8	3.1	5.1	55.9	61.0
ADVP	0.63	24.0	0.9	4.9	6.4	63.9	70.3
SBAR	0.41	26.2	0.9	6.5	0.5	66.0	66.5
S	0.03	27.3	0.9	6.4	0.6	64.8	65.4
VP	0.09	30.4	0.8	6.5	0.9	61.5	62.4
PP	2.05	23.7	0.8	9.0	1.4	65.0	66.4
NNx	0.28	22.9	0.5	9.3	1.9	65.5	67.4
ADJP	0.10	26.9	0.7	9.1	1.0	62.3	63.3
PRN	0.04	19.3	4.3	10.6	5.9	59.9	65.8
NP	0.03	11.5	1.7	17.2	0.7	68.8	69.5

Table 2: English-French adjunct alignment

Depending on the category, 11.5% to 35.1% of the English adjuncts lead to a phrase pair that is not consistent with the word alignments. Low alignment-consistency for the RB category is due in part to discontinuous alignments as, e.g., ‘*not*’/‘*ne ... pas*’. A second informative measure for adjunct alignment is the proportion of aligned French phrases that are not consistent with the French parse, i.e., that fall across constituent boundaries. The worse results are obtained for the parenthetical PRN and adposed NP’s and are caused by the the lack of punctuation handling of the French parse, which results in wrong attachments. The latter issue also concerns the categories PP, ADJP and NNx . Consequently, figures for these categories can be partially imputed to parsing quality. What remains are English adjuncts aligning to zero, one or more French constituents, with

³A phrase pair is consistent with word alignments iff none of the words in one of the two phrases is aligned outside the other phrase, see also (Koehn et al., 2003).

figures varying between 61.0% for RB and 78.2% for JJ. We interpret these figures as an upper bound on adjunct alignment under word alignments, and with the restriction that our identification criteria also include a portion of false adjuncts.

To try and answer how often aligned, parse-consistent French constituents are also adjuncts, we then looked at their categorial tag(s). Depending on the category, English adjuncts have 52 to 1952 different projections on the French side. This has to do with the number of tags used by the Arborescence treebank, 37, all of which but one appear in the projections, in combination with a flat parse structure. To simplify the analysis, we only looked at the three most frequent projections for each category. Results are displayed in Table 3, showing a fairly high dispersion of the projections: in the worst case of VP, the three first projections cover only 38.6% of all 229 cases. In the best case with RB, 84.1% of adjuncts are covered.

category	three most frequent projections (%)				LB
JJ	ADJ - 69.0	N - 10.1	NUM - 3.0		64.8
RB	ADV - 78.5	ADJ - 3.5	N - 2.1		51.1
ADVP	ADV - 60.4	PP - 5.8	ADJ - 3.0		50.6
SBAR	FCL - 35.1	PP - 6.2	NP - 4.3		30.6
S	PP - 45.7	ICL - 6.3	PRP ICL - 2.2		35.7
VP	ICL - 18.6	FCL - 10.2	V-PCP2 PP- 9.8		30.2
PP	PP - 55.1	PP PP - 7.0	NP - 3.5		44.0
NNx	N - 35.2	ADJ - 26.6	PP - 14.5		28.8
ADJP	ADJP - 29.7	PAR - 9.4	N ADJ - 6.6		29.5
PRN	NUM - 29.9	N - 11.6	NP - 10.5		37.0
NP	NP - 28.6	PROP - 24.6	PROP NP - 11.2		45.0

Table 3: Most frequent French projections

The most frequent projections illustrate that English adjunct constituents tend to be aligned to French constituents of comparable nature. The only noticeable anomaly is that of NNx aligning to N in French. French uses much less nominal qualifiers than English, and a closer look reveals that in most cases, the NNx constituent was translated by a PP modifier in French, but that the word alignments aligned it to the PP’s nominal constituent, instead of the entire PP.

With the exclusion of the NNx→N derivation, taking the proportion of parse-consistent French constituents with the three most frequent projections, and adding it to the proportion of null-aligned English adjuncts gives a lower bound (LB) on adjunct alignment. The resulting lower-bound figures displayed in Table 3 could be much refined. On one hand, we assume here, based on a succinct qualitative analysis of the data, that all first three

projections, excepted NNx→N, actually concern French adjuncts; on the other hand, considering more projections for each category is bound to increase figures.

3 Smoothing a PBSMT model by factoring out adjuncts

Figure 2 shows a schematic view of the procedure to smooth a phrase-based model by adjunct-pair deletion. We train a baseline using the Moses toolkit (Koehn et al., 2007). Besides, the training data and the word alignments trained on this data are used to generate new training data by adjunct-pair deletion; Section 3.1 explains how this is done. We then execute part of the Moses training to extract and score phrase pairs from the generated data. Finally, the resulting model is interpolated with the baseline as explained in section 3.2.

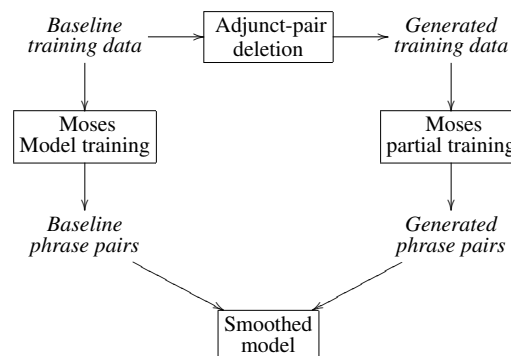


Figure 2: Building a smoothed model

3.1 Training-data generation

We identified 4.9M English adjuncts in the 0.95M parsed sentences of the English Europarl corpus, 3.7M of which lead to consistent phrase pairs. For each sentence pair, we try to generate as many sentence pairs and associated word alignments as there are combinations of adjunct pairs. Data growth is then exponential, and we obtain 95M possible adjunct combinations, though more than half of these contain overlapping adjuncts. To further limit the amount of generated data, combinations are filtered based on the distance between adjuncts. This filtering is combined with measures to control the quality of the generated data. These measures lead to the generation of 9.4M sentence pairs, which can be further brought down by a language-model filter. Next we flush out the details of the filtering methods and explain how we interpolate the model trained on the original data

with the model trained on the thus generated data.

Distance-based filtering

Deleting adjunct-pair combinations allows to obtain more phrase pairs than would be possible by deleting adjunct pairs separately. However, as phrase length is typically limited in phrase-based models, there is no benefit in deleting combinations of distant adjunct pairs. We therefore only considered combinations in which all English adjuncts are separated by less than $l_M - 1$ tokens, where l_M is the maximum phrase length. Note that using only the distance between English adjuncts relies on the assumption that French adjuncts will be distant if their English counterparts are.

Adjunct-gap junction correction

As adjuncts can be marked typographically, typically by surrounding commas, we try to prevent adjunct deletion from resulting in incorrect sequences of punctuation marks. A sequence of at least two of the following tokens is considered incorrect, as is the occurrence of any of these tokens at the start of a sentence:

, . : ; ? ! -

We try to remove misplaced punctuation marks as follows: if punctuation is aligned to the empty string or if it is aligned to something together with some other token, then it is deleted; if punctuation is aligned to a punctuation mark, and no other token is aligned to that punctuation mark, then punctuation is deleted on both sides. Sentence pairs that contain sequences of incorrigible punctuation marks are discarded.

Conversely, one also uses punctuation to try and increase the number of potentially interesting adjunct pairs: If a given adjunct pair is found not to be consistent with word alignments, one tries to extend it to adjoining punctuation.

A second measure aiming at improving the quality of the generated data consists in ensuring that if an English adjunct is deleted just after the indefinite article ‘*a*’/‘*an*’, the form of the article is modified to account for the first letter of the new following word: ‘*a*’ is changed to ‘*an*’ if it is now followed by a vowel, and likewise for the form ‘*an*’.

Language-model filter

A final filtering measure consists in comparing the language-model probability P_{LM} of each generated French sentence f and English sentence e with that of the French sentence f_0 and the English

sentence e_0 they are generated from. Sentences are corrected for length, and an additional threshold k is used to control the amount of generated data. Accordingly, only those sentence pairs that satisfy the following equations are actually generated:

$$P_{LM}(e)^{1/|e|} \geq k \cdot P_{LM}(e_0)^{1/|e_0|} \quad (1)$$

$$P_{LM}(f)^{1/|f|} \geq k \cdot P_{LM}(f_0)^{1/|f_0|} \quad (2)$$

3.2 Model smoothing

The baseline’s translation model is smoothed by linear interpolation with the model trained on the generated data, following Equation 3.

$$\phi_I(\bar{s}|\bar{t}) = \lambda\phi_B(\bar{s}|\bar{t}) + (1 - \lambda)\phi_A(\bar{s}|\bar{t}) \quad (3)$$

where $\phi_B(\bar{s}|\bar{t})$ and $\phi_A(\bar{s}|\bar{t})$ are the translation probability distributions in the baseline and the new model, respectively. The probability distributions are normalized to ensure model consistency ⁴.

We used either a constant interpolation parameter λ or one inspired from the Good-Turing estimate. In this case, the probability mass allocated to the probability distributions $\phi_A(\bullet, \bar{t})$ increases with the relative frequency of single-occurrence phrase pairs with a constituent \bar{t} . The interpolation parameter $\lambda(\bar{t})$ is defined by:

$$1 - \lambda(\bar{t}) = \frac{n1}{n1 + N} \quad (4)$$

where $n1$ is the count of single-occurrence phrase pairs, and N the total count of phrase pairs with a constituent target phrase \bar{t} . As most target constituent phrases in the baseline are associated with singleton phrase pairs, adding $n1$ to the denominator of Equation 4 ensures that $1 - \lambda(\bar{t})$ never reaches 1. To prevent the opposite, $1 - \lambda(\bar{t})$ is set to 10^{-4} by default.

The phrase-pair tables contain both translation probability estimates conditioned on the target phrases, and inverse translation probability estimates conditioned on the source phrases. Interpolation is performed for both distributions.

Probabilities in the reordering model are estimated individually for each phrase pair, consequently one can directly enrich the reordering table with the new model’s table without smoothing. The enriched reordering model consists therefore of the baseline model and of the new model’s reordering probabilities for the phrase pairs in $A-B$.

⁴The normalization factor is λ , 1 or $1 - \lambda$, depending on whether the conditioned target phrase is known to the baseline model only, to both models, or to the new model only.

4 Experiments

The basic set-up for the experiments uses the 2007 Workshop on Machine Translation (WMT07) baseline’s training data. The generated training data is obtained with a language-model filter threshold $k = 0.7$, yielding 4M sentence pairs. Models are built to decode from French to English. The tuning parameters of the baseline are re-used for the smoothed models.

We used four test sets: the in-domain WMT07 test set `devtest`, the out-of-domain WMT07 test set `nc-test`, an adjunct-poor test set `adjpoor` and a second out-of-domain test set `hansards` derived from the Hansards corpus.

The `adjpoor` test set is derived from `devtest` by adjunct-pair deletion, following the same procedure as for the training data: The new test set contains the sentence pairs that are generated by removing combinations of adjunct pairs in `devtest`, without replication of the original sentence pairs. The language-model threshold is set to 1.0 in order to enhance the quality of the generated sentence pairs while limiting their number. The resulting test set consists of 8586 sentence pairs. While not all sentence pairs are equally grammatical, the test set allows to compare the performance of the generated models and of the baseline on adjunct-poor data.

The `hansards` test set consists of the 2000 first non-comment sentence pairs of the Hansards’ House Debates Test Set, where non-comment sentence pairs are defined as ones for which the English sentence ends with a period. The selected sentence pairs are tokenized and lowercased as for the WMT07 test sets.

4.1 Results

Table 4 reports the BLEU scores obtained by the baseline and the smoothed models when varying the amount of generated data with the language-model filter, and using two interpolation parameters, $\lambda = 0.999$ or the Good-Turing inspired λ_{GT} .

The smoothed models perform only slightly better than the baseline on the in-domain test set `devtest`, but significantly ⁵ on the `adjpoor` test set. We found that giving more weight to the generated data, using $\lambda = 0.99$ and $\lambda = 0.9$, de-

⁵Significance was measured at $p = 0.05$ through approximate randomization, using FastMtEval: http://www.computing.dcu.ie/~nstroppa/softs/fast_mt_eval.tgz.

	devtest	adjpoor	nc-test	hansards
baseline	32.47	33.18	24.41	22.24
TD _G = 1M:				
$\lambda = 0.999$	32.47	33.31	24.42	22.18
λ_{GT}	32.50	33.30	24.44	22.09
TD _G = 4M:				
$\lambda = 0.999$	32.52	33.35	24.38	22.16
λ_{GT}	32.51	33.49	24.42	22.12

Table 4: BLEU scores in basic set-up

creased model performance. The benefit of generating more training data is seen best on `adjpoor` with λ_{GT} . In the rest of this document, results are reported for a model smoothed with λ_{GT} .

Table 5 reports the BLEU scores obtained by the baseline and a smoothed model with λ_{GT} when trained on the first 10000 sentence pairs of the normal training set. With a small training set, the

	devtest	adjpoor	nc-test	hansards
baseline	25.94	26.24	15.77	16.56
smoothed	25.97	26.16	15.67	16.66

Table 5: BLEU scores with a small training set

smoothed models still perform but slightly better than the baseline. However, they now fail to outperform the baseline on the `adjpoor` test set.

To assess whether the lack of improvement could be related to the asymmetry of the adjunct-deletion process, we used the generated data to smooth an English-to-French model, but this provided inconclusive again.

We found that the smoothed model could perform significantly better than the baseline, both on `devtest` and `adjpoor`, if one uses the tuning parameters of the smoothed model instead of those of the baseline. Results are given in Table 6.

	devtest	adjpoor	nc-test	hansards
baseline	32.31	33.56	24.55	22.27
smoothed	32.50	33.79	24.46	22.27

Table 6: Effect of retuning

The language model used by the decoder is trained on the training data of the baseline, and as such it may penalize new phrase pairs. As a last experiment, we interpolated a language model

trained on the baseline data with one trained on the generated data, with an interpolation parameter value of 0.999. We did not retune the model, but re-used instead the tuning parameters of the baseline with a language model trained on it. Results are reported in Table 7. These results are nearly

	devtest	adjpoor	nc-test	hansards
baseline	32.47	33.18	24.42	22.23
smoothed	32.52	33.50	24.40	22.10

Table 7: Effect of an interpolated language model

identical to those of Table 4, indicating that there is no benefit in using an interpolated language model.

4.2 Results Analysis

To understand why the smoothed model shows only a minor improvement over the baseline, we looked at the repartition of new phrase pairs at different stages of decoding, and we measured the proportion of test sentences affected by smoothing.

Model contents While the deletion of adjunct pairs allows to generate many new phrase pairs, only few of them are selected by the decoder. Table 8 gives the size of the smoothed model and the repartition of its phrase pairs at three stages: in the training table, in the test-set filtered table, and in the phrase pairs used by the decoder. Phrase pairs are partitioned in the following categories: phrase pairs contained in the baseline’s training data only; phrase pairs contained both in the baseline’s training data and the generated data; generated phrase pairs providing new translation options for source phrases that are known to the baseline; generated phrase pairs containing a source phrase unknown to the baseline.

	table size	base. only (%)	shared (%)	trans. options (%)	new input (%)
training	67.1M	10.4	52.0	7.2	30.4
filtered	4.84M	10.0	72.9	17.0	0.2
decoding	26.7k	1.4	98.4	0.0	0.2

Table 8: Model contents

When tables are filtered for decoding, the proportion of phrase pairs providing new input phrases shrinks, showing that the smoothed model brings proportionally little input phrase pairs that match the test data. Nearly all phrase pairs used for

decoding are shared by the baseline and the generated table, while none of the generated phrase pairs with new translation options are used. It may be interesting to note that regardless of their origin, all the phrase pairs used at decoding have a target constituent that is used both by baseline and generated phrase pairs. Consequently, even when a generated phrase pair with a new input is used, it provides the system with an existing translation option.

Effect on output translation As the contribution of the enriched models in terms of phrase pairs is minimal, it is interesting to see how many output sentences actually differ from the baseline. Table 9 gives the number of sentences with a different translation and the associated BLEU scores for each test-set in the basic set-up. When translation output is identical, one distinguishes sentences with an identical or a different segmentation.

	devtest	adjpoor	nc-test	hansards
≠ translation	645	3722	687	597
BLEU base	29.73	29.71	22.95	20.77
BLEU smoothed	29.81	30.31	22.99	20.44
≠ segmentation	488	2504	440	460
= segmentation	867	2360	880	943
BLEU	34.88	36.38	26.10	23.84

Table 9: Effect of the models on output translation

Table 9 shows that although the enriched model contributes few new phrase pairs, output translation is different for 30% to 43% sentences, indicating that the smoothed probability estimates lead to a different choice of output phrases. This is also reflected by the number of identical translations with a different segmentation (22% to 29%). Note that differences seem very localized, as they tend to concern sequences of two phrases only.

If one only considers different translations, the improvement of the smoothed model over the baseline on devtest is slightly higher than overall, but still not significant. It does however indicate that smoothing helps to improve results.

5 Conclusion

We presented projection figures for English adjuncts into French adjunct-like categories, reporting upper-bound values varying between 61.0% to 78.2% depending on the adjunct category, and lower-bound values between 28.8% and 64.8%.

Besides, we presented a novel way of enriching a PBSMT model by factoring out adjuncts. We

found that a model enriched in this manner only leads to a minor improvement over the baseline. Our system could be improved, notably by extending the class of adjuncts to account for other optional constituents that do not have the status of modifiers, e.g., coordinated elements.

However the main hurdle for our system is that one can only remove adjuncts, and not add any. Consequently, our system performs best on adjunct-poor data, but that is not generally the nature of translation data. Therefore we think that it would be interesting to use adjuncts as a label in a basic SCFG as that of Chiang (2005).

Finally, it would be interesting to investigate the effect of adjunct-pair deletion on other language pairs. While we relied on structural similarity between French and English to align adjuncts, the notion of adjunct is not only syntactical but also has semantic, and therefore cross-linguistic value. Future research might tell whether there is more to gain from adjunct-pair deletion on language pairs that are harder to translate.

References

- Abeillé, Anne and Yves Schabes and Aravind K. Joshi. 1990. Using Lexicalized Tags for Machine Translation. *Proceedings of the 13th International Conference on Computational Linguistics*, Helsinki, Finland, 1–6.
- Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2), 263–311.
- Chiang, David. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, Ann Arbor, MI, 263–270.
- Dorr, Bonnie J., Lisa Pearl, Rebecca Hwa, and Nizar Habash. 2002. DUSTer: A Method for Unraveling Cross-Language Divergences for Statistical Word-Level Alignment. *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas on Machine Translation: From Research to Real Users*, Tiburon, CA, 31–43.
- El Kholy, Ahmed and Nizar Habash. 2010. Orthographic and morphological processing for English-Arabic statistical machine translation. *Actes de Traitement Automatique des Langues Naturelles (TALN)*, Montreal, Canada.
- Foster, George, Roland Kuhn, and Howard Johnson. 2006. Phrasetable Smoothing for Statistical Machine Translation. *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, Sydney, Australia, 53–61.
- Hwa, Rebecca, Philip Resnik, Amy Weinberg, and Okan Kolak. 2002. Evaluating Translational Correspondence using Annotation Projection. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA, 392–399.
- Joshi, Aravind K. 1983. Recursion and Dependencies: an Aspect of Tree Adjoining Grammars (TAG) and a Comparison of Some Formal Properties of TAGs, GPSGs, PLGs, and LPGS. *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, Cambridge, MA, 7–15.
- Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. *Proceedings of the Human Language Technology conference - North American chapter of the Association for Computational Linguistics annual meeting*, Edmonton, Canada, 127–133.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Federico Marcello, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. *Annual meeting of the Association for Computational Linguistics*, demonstration session, Prague, Czech Republic.
- Kuhn, Roland, George Foster, Samuel Larkin, and Nicola Ueffing. 2006. PORTAGE Phrase-Based System for Chinese-to-English Translation. *TCSTAR Workshop on Speech-to-Speech Translation*, Barcelona, Spain, 75–80.
- Nießen, Sonja and Hermann Ney. 2004. Statistical Machine Translation with Scarce Resources Using Morpho-syntactic Information. *Computational Linguistics* 30(2), 181–204.
- Quirk, Chris and Arul Menezes. 2006. Do we need phrases? Challenging the conventional wisdom in Statistical Machine Translation. *Proceedings of the Human Language Technology conference - North American chapter of the Association for Computational Linguistics annual meeting*, New York, NY, 9–16.
- Shieber, Stuart M. 2007. Probabilistic Synchronous Tree-Adjoining Grammars for Machine Translation: The Argument from Bilingual Dictionaries. *Proceedings of the Workshop on Syntax and Structure in Statistical Translation*, Rochester, NY, 88–95.
- Schwenk, Holger, Marta R. Costa-Jussà, and José A.R. Fonollosa. 2007. Smooth Bilingual N-gram Translation. *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Prague, Czech Republic, 430–438.

LTG vs. ITG Coverage of Cross-Lingual Verb Frame Alternations

Kartteek Addanki, Chi-kiu Lo, Markus Saers and Dekai Wu

Human Language Technology Center

Dept. of Computer Science and Engineering

Hong Kong University of Science and Technology

{vskaddanki|jackielo|masaers|dekai}@cs.ust.hk

Abstract

We show in an empirical study that not only did all cross-lingual alternations of verb frames across Chinese–English translations fall within the reordering capacity of Inversion Transduction Grammars, but more surprisingly, about 97% of the alternations were expressible by the far more restrictive Linear Transduction Grammars. Also, about 71% of the cross-lingual verb frame alternations turn out to be monotonic even for diverse language pairs such as Chinese–English. We also observe that a source verb frame alternation pattern translates into a small subset of the possible target verb frame alternation patterns, based on the construction of the source sentence and the frame set definitions. As a part of our evaluation, we also present a novel linear time algorithm to determine whether a particular syntactic alignment falls within the expressiveness of Linear Transduction Grammars. To our knowledge, this is the first study that attempts to analyze the cross-lingual alternation behavior of semantic frames and the extent of their coverage under syntax-based machine translation formalisms.

1 Introduction

In this paper we present a first empirical study on the cross-lingual verb frame alternations by aligning semantic role fillers in parallel sentences. We evaluate how many of these alignments fall within the expressiveness of two well known syntax based machine translation formalisms: Inversion Transduction Grammars (Wu, 1997) and Linear Transduction Grammars (Saers, 2011). As a part of our evaluation, we discuss the reordering of semantic roles within a frame and across frames within a sentence. We also present a novel algorithm to determine whether there exists a canonical parse for an alignment under Linear Transduction Grammars.

While recent years have seen continued improvements in the accuracy of SMT using tree-structured and syntactic models (Wu, 1997; Wu and Chiang, 2009; Wu, 2010; Wu and Fung, 2009b,a), only a few attempts (Wu and Fung, 2009b) have been made towards using semantic roles to guide SMT. Recent studies (Wu and Fung, 2009a) show that most of the glaring errors made by statistical machine translation systems are a result of confused semantic roles which result in serious misunderstanding of the essential meaning. Semantic roles have also been successfully used in evaluating translation utility (Giménez and Márquez, 2007, 2008; Callison-Burch *et al.*, 2007, 2008; Lo and Wu, 2011a,b). However, no effort has been made to identify the reordering of semantic role fillers across languages. Such an analysis is interesting for two reasons: (1) to determine how much reordering we really need in order to preserve meaning while translating, and (2) to determine which existing syntactic SMT models have an inherent bias towards such a reordering. The first reason helps us determine an upper bound on the expressiveness and hence the computational complexity of the syntactic models. The second enables us to choose syntactic SMT models that can be adapted to incorporate semantic knowledge. Such a system should theoretically be able to capture semantically valid syntactic generalizations, thereby improving translation accuracy.

To fulfill the above requirements, we evaluate two well known syntax-based machine translation formalisms: Inversion Transduction Grammars or ITGs (Wu, 1997) and Linear Transduction Grammars or LTGs (Saers, 2011). As discussed in Wu (1997), ITGs allow nearly all possible reorderings (22 out of 24) given up to four semantic role labels within a semantic frame. Further, various forms of empirical confirmation for the effectiveness of ITG expressivity constraints (Zens and Ney, 2003; Zhang and Gildea, 2005, 2004) motivate us to choose it as a likely candidate. Though ITGs are far more constraining than other higher order syntax directed transduction grammars and IBM models, it would be interesting to see how far an even more constrained model is able to handle reorder-

ings of semantic role fillers. For this purpose, we choose LTGs which are bilingual generalizations of linear grammars and highly constrained compared to the ITGs (Saers *et al.*, 2011).

In order to identify reorderings that produce semantically good translations, one should approach the task of aligning semantic role fillers carefully. Such an alignment should accurately match at least the corresponding basic event structure “Who did what to whom, when, where and why” in both source and target languages in order to preserve meaning (Pradhan *et al.*, 2004). Further, a complete analysis of the syntactic alignments generated as a result of aligning semantic role fillers entails examining the reordering of roles both within a frame and across all the frames in one sentence. The possibility that an exact alignment might not exist at all even for semantically valid translations should also be considered.

In this paper, we present an empirical study of the semantic reorderings as a result of aligning cross-lingual semantic role fillers. We also determine to what extent the alignment constraints of ITGs and LTGs permit such reorderings. We use semantically annotated Chinese–English parallel resources and manually align the semantic role fillers. In order to identify the alignments permitted by LTGs we propose a novel linear time algorithm. Our results indicate that ITGs permit all the syntactic reordering occurring from aligning semantic role fillers. Interestingly, we also show that about 97% of the alignments are handled by LTGs. We also observe that all the verb frame alternations of semantic frames fall within the reordering capability of both LTGs and ITGs.

The rest of the paper is organized as follows. In the next section, we state and prove an algorithm to determine whether an alignment corresponding to a given permutation can be parsed by a bracketing linear transduction grammar. In section 3, we describe our experimental setup. Results and the conclusion follow in sections 4 and 5.

2 LTG parsability algorithm

In this section, we present a linear time algorithm to determine whether or not there exists a canonical parse for an alignment under LTGs. Although, LTGs are restricted forms of ITGs, the algorithm for determining whether a permutation corresponding to an alignment can be parsed by a LTG is not a special case of the linear time skeleton algorithm for binarization of synchronous grammars (Huang *et al.*, 2009). The linear time skeleton algorithm builds canonical binarization trees by reducing greedily but such an approach would not work for a LTG. For example, the permutation [3, 2, 0, 1] which can be parsed by an LTG reduces to 2-3, 0-1 on the stack which cannot be further reduced.

We propose an algorithm that makes use of a technique similar to top-down parsing of bisenences using linear transduction grammars. The algorithm is as shown in the procedure parsable. In order to prove the correctness of the algorithm, we use the definition of *permuted sequence* from Huang *et al.* (2009) but we redefine *proper split* in the context of BLTGs. The proof is as follows:

Definition 1. A *permuted sequence* is a permutation of consecutive integers. If a permuted sequence of \mathbf{a} can be split into the concatenation of a permuted sequence \mathbf{b} and a single element of permutation α such that $\mathbf{a} = (\mathbf{b}; \alpha)$ or $\mathbf{a} = (\alpha; \mathbf{b})$, then the corresponding split is called the *proper split* of \mathbf{a} .

The definition of a proper split implicitly imposes the constraints of a linear transduction grammar. Restricting one of the elements in a split to be a single element in the permutation is equivalent to allowing at most one nonterminal in the right hand side of productions. The definition of a permuted sequence enforces the projection constraints of transduction grammars by allowing no gaps in the reorderings within a constituent.

Lemma 1. A split $\mathbf{a} = (\mathbf{b}; \alpha)$ or $\mathbf{a} = (\alpha; \mathbf{b})$ is proper if and only if $\alpha = \max(\mathbf{a})$ or $\alpha = \min(\mathbf{a})$.

Proof. We prove both the forward and reverse implications as follows:

1. If $(\mathbf{b}; \alpha)$ is a proper split of \mathbf{a} , then \mathbf{b} is a permuted sequence. From Definition 1, all the elements in \mathbf{b} should be consecutive. Hence α is either greater than all the elements in \mathbf{b} or less than all the elements in \mathbf{b} which implies $\alpha = \max(\mathbf{a})$ or $\alpha = \min(\mathbf{a})$ respectively. Similar conclusions can be made for the case when $\mathbf{a} = (\alpha; \mathbf{b})$.
2. If $\alpha = \max(\mathbf{a})$ and there exists a split of \mathbf{a} such that $\mathbf{a} = (\mathbf{b}; \alpha)$, then \mathbf{b} is a permuted sequence from $[\min, \dots, \max - 1]$. This makes $(\mathbf{b}; \alpha)$ a proper split. Similarly, when $\alpha = \min(\mathbf{a})$, \mathbf{b} is a permuted sequence from $[\min + 1, \dots, \max]$ and $(\mathbf{b}; \alpha)$ is a proper split. The case when $\mathbf{a} = (\alpha; \mathbf{b})$ is similar. □

Lemma 2. If \mathbf{a} is a permuted sequence covering $[\min, \dots, \max]$, and there exists a proper split of \mathbf{a} such that $\mathbf{a} = (\mathbf{b}; \alpha)$ or $\mathbf{a} = (\alpha; \mathbf{b})$, then \mathbf{b} is a permuted sequence covering $[\min, \dots, \max - 1]$ or $[\min + 1, \dots, \max]$.

Proof. From Lemma 1, $\alpha = \max(\mathbf{a})$ or $\alpha = \min(\mathbf{a})$. Therefore, \mathbf{b} covers the range $[\min, \dots, \max - 1]$ or $[\min + 1, \dots, \max]$ according to whether α is $\max(\mathbf{a})$ or $\min(\mathbf{a})$ respectively. □

Procedure parsable(\mathbf{a}, \min, \max)

```
input : A permuted sequence  $\mathbf{a}$  of range  $[\min, \dots, \max]$ 
output: true or false depending on the whether or not  $\mathbf{a}$  is BLTG parsable
begin
  if  $\max - \min + 1 = 1$  then                                     // base case
  | return true
  else
  | if first( $\mathbf{a}$ ) =  $\min$  then                                       //  $\mathbf{a} = [\alpha : \mathbf{b}]$ 
  | | shift( $\mathbf{a}$ )                                     // remove the first element of  $\mathbf{a}$ 
  | | return parsable( $\mathbf{a}, \min + 1, \max$ )
  | else if first( $\mathbf{a}$ ) =  $\max$  then                                       //  $\mathbf{a} = \langle \alpha : \mathbf{b} \rangle$ 
  | | shift( $\mathbf{a}$ )
  | | return parsable( $\mathbf{a}, \min, \max - 1$ )
  | else if last( $\mathbf{a}$ ) =  $\min$  then                                       //  $\mathbf{a} = \langle \mathbf{b} : \alpha \rangle$ 
  | | pop( $\mathbf{a}$ )                                     // remove the last element of  $\mathbf{a}$ 
  | | return parsable( $\mathbf{a}, \min + 1, \max$ )
  | else if last( $\mathbf{a}$ ) =  $\max$  then                                       //  $\mathbf{a} = [\mathbf{b} : \alpha]$ 
  | | pop( $\mathbf{a}$ )
  | | return parsable( $\mathbf{a}, \min, \max - 1$ )
  | else
  | | // no proper split exists
  | | return false
  |
end
```

Definition 2. A permuted sequence \mathbf{a} is said to be parsable if:

1. \mathbf{a} is a permuted sequence of size 1 i.e., $\mathbf{a} = (\alpha)$
2. there exists a proper split of \mathbf{a} containing a permuted sequence \mathbf{b} , which is also parsable.

This is a recursive definition and associates a hierarchical tree structure with each permutable sequence. The tree structure is equivalent to the biparse tree which parses the alignment represented by the permutable sequence. For the sake of completeness, we define the parse tree below:

Definition 3. A parse tree $t(\mathbf{a})$ of a parsable sequence \mathbf{a} is either:

1. α if $\mathbf{a} = (\alpha)$, or
2. $[\alpha t(\mathbf{b})]$ if $\mathbf{a} = (\alpha; \mathbf{b})$ and $\alpha = \min(\mathbf{a})$, or
3. $\langle t(\mathbf{b}) \alpha \rangle$ if $\mathbf{a} = (\alpha; \mathbf{b})$ and $\alpha = \max(\mathbf{a})$, or
4. $[t(\mathbf{b}) \alpha]$ if $\mathbf{a} = (\mathbf{b}; \alpha)$ and $\alpha = \max(\mathbf{a})$, or
5. $\langle \alpha t(\mathbf{b}) \rangle$ if $\mathbf{a} = (\mathbf{b}; \alpha)$ and $\alpha = \min(\mathbf{a})$ where $t(\mathbf{b})$ is the parse tree of \mathbf{b} .

We use the same notation as Wu (1997) for representing the straight and inverted configurations. We also note that there might exist more than one parse tree for a parsable sequence but we are interested only in whether or not there exists at least one parse tree.

Theorem 1. Procedure parsable runs in time linear to the length of the input and succeeds (i.e., returns **true**) if and only if the input permuted sequence \mathbf{a} is parsable.

Proof. 1. If the procedure returns **true**, then \mathbf{a} is binarizable as we can recover a parse tree from the algorithm.

2. If \mathbf{a} is parsable, then the procedure must return **true**.

We prove this by a complete induction on n , the length of \mathbf{a} .

Base case: $n = 1$, trivial. Assume that the condition holds for all $n' < n$.

From Definition 2, if a permuted sequence is parsable then there exists a proper split. We check for all possible values of α in a proper split (see Lemma 1 and Definition 1). By induction hypothesis, the procedure succeeds as the procedure is called on a permuted sequence of length $n - 1$ after the first split.

As the procedure is recursively called a maximum of n times where n is the length of \mathbf{a} and each procedure call takes $O(1)$ time, the algorithm is linear with respect to the length of the input. The total complexity is $O(n)$. \square

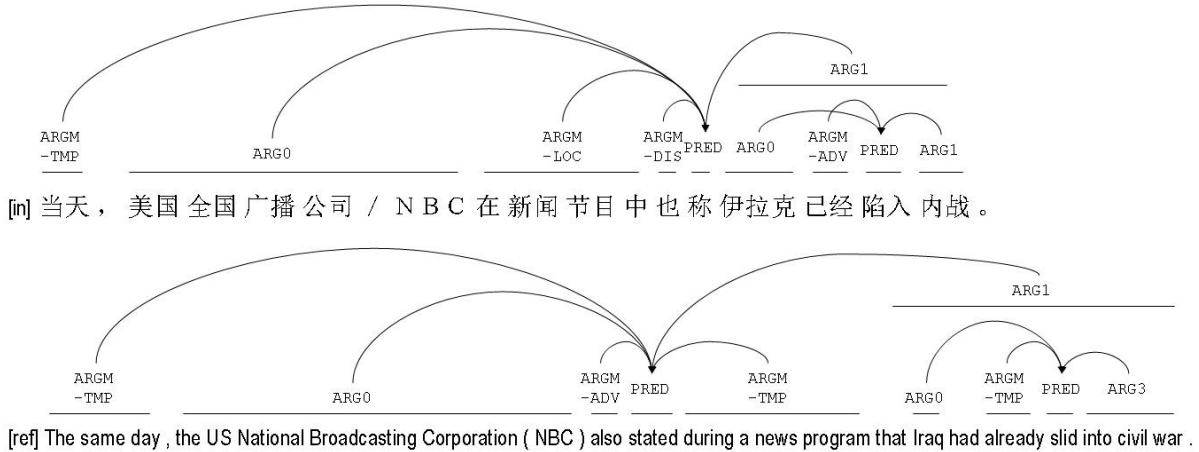


Figure 1: An example of nested semantic role fillers

3 Experimental setup

3.1 Semantic role alignment

As a first step in our experiment, we would like to identify semantic role fillers in the target language sentence that match the basic event structures of “Who did what to whom, when, where and why” in both source and target sentences. We use a randomly sampled subset of 100 sentence pairs from the Chinese–English parallel corpus derived from Phase 2.5 of the DARPA GALE program. The Chinese and English sentences are annotated with gold-standard semantic roles in Propbank Style and belong to the news wire genre. We use a bilingual speaker to manually align the semantic roles. We do not attempt to automatically align semantic role fillers with identical semantic role labels in a frame as the gold standard annotation was done monolingually leading to a possibility of mismatch between the source and target role fillers.

The aligner was instructed to align role fillers that are precise translations of each other. First the predicates corresponding to different frames in source and target sentences are aligned. For each aligned predicate, the role fillers for the frame modifiers are aligned. We assume that if there is not an exact match between predicates in source and target sentences, none of the role fillers for the other modifiers can be aligned. Such a scenario would occur only when the source sentence is paraphrased using a totally different construction in the target language and we ignore such sentence pairs. We only found one such example in our sample and it is shown below:

Source: 报道说，今年以来英国有关征收“绿色税”的争论和猜测不断。

Gloss: Report said, this year throughout Britain related levying “green tax”’s controversy and speculation did not stop.

Target: According to the report, “green taxes” have come under constant controversy and speculation in Britain this year.

One can notice that both source and target sentences convey the same semantic information but have no predicates that can be aligned. While the source sentence has two predicates 征收 (levy) and 不断 (did not stop), none of them match with the predicate in the target sentence which is “come”. We assume that such constructions are rare enough in parallel corpora and treat them as an exception rather than a rule.

We also do not align partially matching semantic role fillers i.e., semantic role fillers that do not contain the same level of information. We have observed that such a mismatch primarily occurs due to the independent annotation of the source and target corpora.

3.2 Extracting semantic reorderings

We extract the semantic reordering information from the manually aligned semantic role fillers both within a single frame and across all the frames in a sentence. While extracting the reorderings we ignore the tokens that: (1) correspond to an unaligned semantic role filler and, (2) are not annotated with any semantic role. Tokens that are not annotated with any semantic role perform the task of providing the syntactic structure to the meaning contained in the semantic role fillers in a sentence. As our goal is primarily to identify the kind of reorderings necessary to preserve meaning we do not deal with these tokens.

Semantic role fillers may contain nested semantic frames. So a bispan corresponding to a semantic role filler alignment might contain bispans corresponding to other semantic roles. This leads to a hierarchical or a compositional syntactic alignment between source and target sentences. However, preserving the compositionality of the syn-

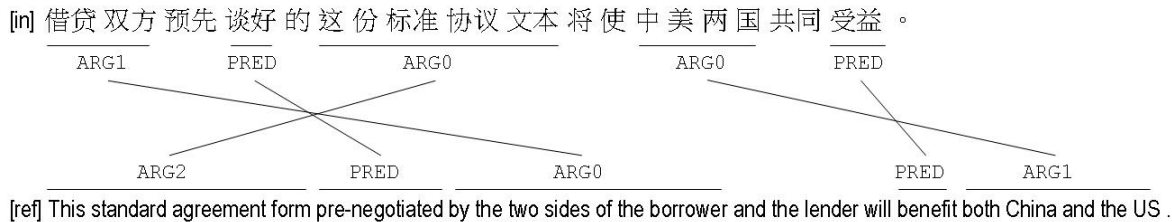


Figure 2: An example of a semantic alignment not parsable by LTG

tactic alignments adds little value in understanding the syntactic reordering necessary to preserve meaning. For example, if a semantic role filler contains a nested semantic frame (or frames) then all the semantic information contained in the encompassing role filler is captured by the role fillers in the nested frame (or frames). An example is shown in figure 1.

So we extract an alignment permutation by identifying the mapping between disjoint semantic role fillers among all the frames in a sentence. We did not encounter any non-disjoint alignments between the role fillers, and we could successfully extract a permutation from all the sentence pairs. In the next stage of the experiment we evaluate whether a given permutation falls within the reordering capacity of ITGs and LTGs.

3.3 Evaluating alignments

We evaluate the alignments of the semantic role fillers both within a frame and across all the frames in a sentence. The reordering within a frame indicates the relation between the cross-lingual verb frame alternation patterns. We also extract the alignments of disjoint semantic role fillers across all the frames in the sentence pairs as discussed in the previous subsection. In both the cases, we determine whether or not there exists a canonical parse for the alignment using an ITG or an LTG. For this purpose we use the shift reduce algorithm proposed in (Huang *et al.*, 2009) for ITGs and the algorithm proposed in Section 2 for the LTGs.

4 Results

We observed that all the cross-lingual alternations of verb frames fall within the reordering capability of both LTGs and ITGs. We did not find any semantic frames which had more than four arguments (including the predicate) in our sample. Both LTGs and ITGs are capable of generating all possible alternations for semantic frames up to three arguments. In the case where there were four arguments we did not encounter any examples where the role fillers formed an *inside-out* ([2, 0, 3, 1] or [1, 3, 0, 2]) which both LTGs and ITGs cannot generate, nor did we find *constituent swapping* ([2, 3, 0, 1]) or *serial inversion* ([1, 0, 3, 2])

which LTGs cannot generate. Note that this result corresponds to the alignment of semantic role fillers within one frame. The alignment of semantic role fillers across all the semantic frames in a sentence is discussed in the next subsection.

4.1 LTGs have high semantic alignment coverage

We observed that the alignment of the semantic role fillers across all the frames in a sentence fall within the reordering capacity of ITGs for all the sentence pairs in our sample. We did not find any translations of semantic frames wherein the role fillers formed an *inside-out* alignment. This observation is consistent with the universal language hypothesis of ITGs (Wu, 1997).

Surprisingly, for about 97% of the sentences, the generated alignments could be expressed by the far more restrictive Linear Transduction Grammars. There were only three sentence pairs that had reordering of verb frame alternations which could not be parsed by an LTG. All the alignments that could not be parsed by an LTG contained the *serial inversion* permutation pattern which can be parsed by an ITG but not by an LTG. Figure 2 shows an example.

In Figure 2, the order of the arguments in two adjacent semantic frames is inverted. Although, the alternation of each semantic frame can be independently parsed by an LTG, the reordering caused by these alternations at the sentence level cannot be parsed. All the alignments in our sample, that could not be parsed an LTG exhibited similar pattern.

Further, we noticed that for about 71% of the sentences, the alignments were monotonic. It is interesting to note that despite reordering at a surface level, the parts that carried the semantic information remained in the same order in both source and target sentences. A possible reason for such a high percentage of monotonic alignments could be the similarity in the word orders of Chinese and English as both languages follow a *subject-verb-object* construction. Further empirical testing is needed to determine whether or not this observation holds true for language pairs with a difference in word order.

Though these sentence pairs are an example of the limitation of LTGs to express reordering that occurs in natural languages, it is interesting to note that the *serial inversion* alignment pattern was infrequent in our sample and there is no occurrence of the *constituent swapping* alignment pattern.

4.2 Topicalization versus fluency

We noticed that reordering of semantic role fillers depends on factors deeper than the syntactic structure of the source sentence. It depends on whether the translation aims to capture the intentional or the extensional semantics of the source sentence. This results in some interesting trade-offs that could be made between topicalization and fluency. Consider the sentence pair in Figure 2. The source sentence contains three semantic frames with predicates 谈好 (negotiated), 使 (cause) and 受益 (benefited) whereas the translation contains only two semantic frames corresponding to the predicates negotiate and benefit.

Source: 借贷双方预先谈好的这份标准协议文本将使中美两国共同受益。

Gloss: Borrower and lender both sides pre-negotiated DE/的 this standard agreement form will cause China US two countries together receive benefit .

Translation: This standard agreement form pre-negotiated by the two sides of the borrower and the lender will benefit both China and the US .

Alternative translation: This standard agreement form pre-negotiated by the two sides of the borrower and the lender will result in both China and the US getting benefited .

In the second frame the VP in the Chinese sentence undergoes a dative alternation upon translation as the double-object construction (NP-NP) for the verb *benefit*, is less fluent and possibly awkward in English. One could argue that the proposed translation is not semantically equivalent to the source sentence because “a difference in the syntactic form always spells a difference in meaning” (Goldberg, 1995). We provide an alternative translation which preserves the topicalization on the possession facet and the possessor rather than on the transfer. While the permutation generated by aligning the source and target sentences cannot be parsed by an LTG ([2, 1, 0, 4, 3]), the alternative translation generates a permutation that can be parsed ([2, 1, 0, 3, 4, 5]).

While both translations manage to convey the meaning in the source sentence correctly, one focuses on fluency and the other on preserving the topicalization. It is not our purpose to compare the translation quality in both cases but to provide an example of the subtle transformation of semantics that occurs while translating and how they affect reordering.

4.3 Verb frame alternation patterns

We studied how alternation patterns change when verb frames are translated from one language to another. If the alternation of the translated verb frame can be estimated based on the source language alternation, it might provide information as to how the target language sentence should be constructed, rather than solely relying on the surface reordering rules. Although Schulte im Walde (2000) showed that verbs can be clustered into semantic categories based on their alternation behavior, little work has been done towards understanding cross-lingual verb frame alternation patterns. As a first step towards understanding the cross-lingual alternation behavior, we collected some statistics and performed a rudimentary qualitative analysis on the alternation patterns of the target language given a source language alternation pattern.

We observed that for about 77% of the semantic frames, the alternation pattern is preserved when translated and for about 4%, the target alternation pattern was a permutation of the source pattern. Surprisingly, for about 19% of the frames, the target frame alternation pattern had a different label which was not present in the source frame. For example, the [*arg0* : *action*] pattern in Chinese gets converted into [*arg1* : *action*] in English. Table 1 shows the counts for the target alternation patterns for some of the frequently occurring source alternations.

A given source alternation pattern is aligned only to a small subset of the possible target frame alternations. For Chinese–English, the alternation pattern remains the same in most cases which could be attributed to the similarity in word order. From Table 1, one can observe that the [*arg0* : *action* : *arg1*], the most frequent source alternation pattern, remains unaltered 88 out of 97 times.

In cases where the target alternation pattern was a permutation of the source pattern, we observed a difference in the voice of the source and target sentences. In most cases, the Chinese sentence in active voice was translated into an English sentence in passive voice. In a few cases, translation demanded a reordering of the source alternation pattern as English had no equivalent construction. For example, in Chinese, when a verb qualifies a noun the verb comes after the noun, while in English it comes before. Hence the phrase 信心增强 (confidence increased) translates into strengthened confidence in English.

For sentence pairs where the source and target alternation patterns differed in labels, we noticed that there were some inconsistencies in the annotation. The sentence pairs were manually annotated with the frame sets defined for Chinese and English

Zh/En alt. patterns	[arg0:action:arg1]	[arg0:action]	[action:arg1]	[arg1:action]	Sum
[arg0:action:arg1]	88	0	0	0	88
[arg0:action]	0	11	0	0	11
[action:arg1]	0	3	39	1	43
[arg1:action]	0	12	6	3	21
[action:arg2]	0	1	5	0	6
[arg0:action:arg2]	3	0	0	0	3
[action:arg4]	0	1	1	0	2
[arg1:action:arg2]	3	0	0	0	3
[arg1:action:arg4]	3	0	0	0	3
Sum	97	28	51	4	

Table 1: Frequency of source and target alternation pattern occurrence

in the Propbank (Palmer *et al.*, 2005). We argue that it is due to the limitation of frame set definitions as they were defined to be consistent within one language but not across languages. For example, in the frame set definition of 死于 (died of), the *arg0* is the *entity who dies*, while in the frame set definition of its translation die, *the deceased is arg1* and there is no *arg0* defined. Similar observations could be made for most of the sentence pairs which differed in source and target alternation labels.

As our initial analysis of cross-lingual verb frame alternation patterns suggests that patterns in one language align with only a restricted subset of patterns in the other language, we believe that it might be possible to learn the target frame alternation patterns given a source frame alternation pattern. However, it is worth noting that it is important to deal with the inconsistencies in the frame set definitions across languages before one attempts such a task. Larger scale experiments are needed in order to reliably identify the relation between source and target alternation patterns.

5 Conclusion

In this paper, we reported a first empirical study of cross-lingual verb frame alternations and made the following observations: (1) the alignments of the semantic role fillers fall within the reordering capacity of ITGs for all the sentences, (2) even highly constrained models such as LTGs are capable of parsing most of these alignments and (3) there appears to be a correlation between the alternation patterns of the source and target verb frames. We also presented a novel algorithm to determine whether or not a permutation falls within the reordering constraints of LTGs.

The first two observations indicate that alignments of parts that carry the semantic information in sentences (i.e., predicates and semantic role fillers) do not warrant a highly expressive model. Further, since the evaluated models have an inherent bias towards generating these alignments, the constraints they enforce would be useful if one were

to automatically align and/or induce semantic role fillers from parallel sentences. It would be interesting to evaluate the performance of the alignments generated by using the semantic role fillers as anchors.

Our observation about the verb frame alternation patterns suggests that it might be possible to predict the target frame alternation pattern given a source frame alternation pattern which would be useful for aligning the verb frames. Although a qualitative evaluation indicated that source sentence construction and frame set definitions can affect target alternation pattern, further evaluation is needed in order to reliably identify features that affect alternation patterns.

As for future work, we think it is interesting to explore methods to incorporate semantic frames in generating robust alignments. It would also be interesting to see whether cross-lingual alternation patterns provide information about verb classes in the bilingual case similar to Schulte im Walde (2000).

Acknowledgments

This material is based upon work supported in part by the Defense Advanced Research Projects Agency (DARPA) under BOLT contract no. HR0011-12-C-0016, and GALE contract nos. HR0011-06-C-0022 and HR0011-06-C-0023; by the European Union under the FP7 grant agreement no. 287658; and by the Hong Kong Research Grants Council (RGC) research grants GRF621008, GRF612806, DAG03/04.EG09, RGC6256/00E, and RGC6083/99E. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the RGC, EU, or DARPA.

References

Chris Callison-Burch, Cameron Forgyce, Philipp Koehn, Christof Monz, and Josh Schroeder. (Meta-) evaluation of Machine Translation. In

- Proceedings of the 2nd Workshop on Statistical Machine Translation*, pages 136–158, 2007.
- Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. Further Meta-evaluation of Machine Translation. In *Proceedings of the 3rd Workshop on Statistical Machine Translation*, pages 70–106, 2008.
- Jesús Giménez and Lluís Màrquez. Linguistic Features for Automatic Evaluation of Heterogenous MT Systems. In *Proceedings of the 2nd Workshop on Statistical Machine Translation*, pages 256–264, Prague, Czech Republic, June 2007. Association for Computational Linguistics.
- Jesús Giménez and Lluís Màrquez. A Smorgasbord of Features for Automatic MT Evaluation. In *Proceedings of the 3rd Workshop on Statistical Machine Translation*, pages 195–198, Columbus, OH, June 2008. Association for Computational Linguistics.
- Adele E. Goldberg. *Constructions: A construction grammar approach to argument structure*. University of Chicago Press, 1995.
- Liang Huang, Hao Zhang, Daniel Gildea, and Kevin Knight. Binarization of synchronous context-free grammars. *Computational Linguistics*, 35(4):559–595, 2009.
- Chi-kiu Lo and Dekai Wu. MEANT: An Inexpensive, High-Accuracy, Semi-Automatic Metric for Evaluating Translation Utility based on Semantic Roles. In *Proceedings of the Joint conference of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies (ACL-HLT-11)*, 2011.
- Chi-kiu Lo and Dekai Wu. SMT vs. AI redux: How semantic frames evaluate MT more accurately. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI-11)*, 2011.
- Martha Palmer, Daniel Gildea, and Paul Kingsbury. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106, 2005.
- Sameer Pradhan, Wayne Ward, Kadri Hacioglu, James H. Martin, and Dan Jurafsky. Shallow Semantic Parsing Using Support Vector Machines. In *Proceedings of the 2004 Conference on Human Language Technology and the North American Chapter of the Association for Computational Linguistics (HLT-NAACL-04)*, 2004.
- Markus Saers, Dekai Wu, and Chris Quirk. On the expressivity of linear transductions. In *Proceedings of the 13th Machine Translation Summit (MT Summit XIII)*, 2011.
- Markus Saers. *Translation as linear transduction: Models and algorithms for efficient learning in statistical machine translation*. Acta Universitatis Upsaliensis., 2011.
- Sabine Schulte im Walde. Clustering verbs semantically according to their alternation behaviour. In *Proceedings of the 18th conference on Computational linguistics-Volume 2*, pages 747–753. Association for Computational Linguistics, 2000.
- Dekai Wu and David Chiang, editors. *Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation (SSST-3) at NAACL HLT 2009*. Association for Computational Linguistics, Boulder, Colorado, June 2009.
- Dekai Wu and Pascale Fung. Can Semantic Role Labeling Improve SMT? In *Proceedings of the 13th Annual Conference of the European Association for Machine Translation (EAMT-2009)*, pages 218–225, Barcelona, Spain, May 2009.
- Dekai Wu and Pascale Fung. Semantic Roles for SMT: A Hybrid Two-Pass Model. In *Proceedings of the 2009 Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL-HLT-09)*, pages 13–16, 2009.
- Dekai Wu. Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. *Computational Linguistics*, 23(3):377–403, 1997.
- Dekai Wu, editor. *Proceedings of the 4th Workshop on Syntax and Structure in Statistical Translation*. Coling 2010 Organizing Committee, Beijing, China, August 2010.
- Richard Zens and Hermann Ney. A comparative study on reordering constraints in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 144–151. Association for Computational Linguistics, 2003.
- Hao Zhang and Daniel Gildea. Syntax-based alignment: supervised or unsupervised? In *Proceedings of the 20th international conference on Computational Linguistics*, page 418. Association for Computational Linguistics, 2004.
- Hao Zhang and Daniel Gildea. Stochastic lexicalized inversion transduction grammar for alignment. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 475–482. Association for Computational Linguistics, 2005.

Oral Session 5 – Research Papers

Learning Machine Translation from In-domain and Out-of-domain Data

Marco Turchi

European Commission JRC,
IPSC - GlobeSec
Via Fermi 2749,
21020 Ispra (VA), Italy

marco.turchi@jrc.ec.europa.eu

Cyril Goutte

Interactive Language Tech.,
National Research Council Canada,
283 Boulevard Alexandre-Taché,
Gatineau QC J8X3X7, Canada

Cyril.Goutte@nrc.ca

Nello Cristianini

Intelligent Systems Lab.,
University of Bristol,
MVB, Woodland Rd,
BS8-1 UB, Bristol, UK

Nello@support-vector.net

Abstract

The performance of Phrase-Based Statistical Machine Translation (PBSMT) systems mostly depends on training data. Many papers have investigated how to create new resources in order to increase the size of the training corpus in an attempt to improve PBSMT performance. In this work, we analyse and characterize the way in which the in-domain and out-of-domain performance of PBSMT is impacted when the amount of training data increases. Two different PBSMT systems, Moses and Portage, two of the largest parallel corpora, Giga (French-English) and UN (Chinese-English) datasets and several in- and out-of-domain test sets were used to build high quality learning curves showing consistent logarithmic growth in performance. These results are stable across language pairs, PBSMT systems and domains. We also analyse the respective impact of additional training data for estimating the language and translation models. Our proposed model approximates learning curves very well and indicates the translation model contributes about 30% more to the performance gain than the language model.

1 Introduction

With the growing availability of bilingual parallel corpora, the past two decades saw the development and widespread adoption of *statistical* machine translation (SMT) models. Given a source (“foreign”) language sentence f and a target (“english”)

language translation e , the relationship between e and f is modelled using a statistical or probabilistic model which is estimated from a large amount of textual data, comprising bilingual and monolingual corpora. The most popular class of SMT systems is Phrase-Based SMT (PBSMT, (Koehn et al., 2003)).

In this paper, we are concerned with analyzing and characterizing the way in which the performance of PBSMT models evolves with increasing amounts of training data. In the SMT community, it is a common belief that learning curves follow logarithmic laws. However, there are few large-scale systematic analyses of the growth rate of the PBSMT performance. Early work (Al-Onaizan et al., 1999) used a relatively small training set and perplexity as evaluation metric. (Koehn et al., 2003) and (Suresh, 2010) show that BLEU score has a log-linear dependency with training corpus size, but this is limited to 350k training sentence pairs. Learning curves were also presented in order to motivate the use of active learning for MT (Bloodgood and Callison-Burch, 2010; Haffari et al., 2011). They attempt to address the challenge of “diminishing returns” in learning MT, although this is again done with small training corpora (<90k sentence pairs), and, on a log-scale, performance seems again to increase linearly. (Brants et al., 2007) produced a large-scale study, but focused on the language model training only, with billions of (monolingual) tokens.

The first complete and systematic analysis of PBSMT learning curves was obtained by (Turchi et al., 2008) using the Spanish-English Europarl, and recently extended to larger training data and more systems by (Turchi et al., 2011). In their work, accurate learning curves obtained over a large range of data sizes confirm that performance

grows linearly in the log domain.

The reason why relatively few systematic studies have been reported may be that producing accurate learning curves up to large data sizes with state-of-the-art systems requires the use of high performance computing in a carefully set up environment. This may seem dispensable when typical SMT research is usually focused on maximizing the performance that can be extracted from a given data set, rather than analysing how this performance evolves. However, we believe that the analysis and quantification of the way machine translation systems learn from data are important steps to identify critical situations which affect the overall translation performance. We also wish to characterize PBSMT performance up to data sizes more typical of current large-scale bilingual corpora.

In the following we pursue three purposes:

1. We confirm, in a systematic way, previous findings that PBSMT performance gains constant improvements for each doubling of the data. This holds across systems, language pairs and over a large range of data sizes.
2. We show that, somewhat surprisingly, this extends to out-of-domain data, although the growth is weaker in that case.
3. We analyse and quantify the relative importance of training data in language and translation model training, and show that the latter contributes about 30% more to the gains in performance.

In contrast with previous work, we build our learning curves using two of the largest available parallel training sets: the French-English Giga corpus and the Chinese-English UN corpus. In addition to being large corpora, these also cover two very distinct language pairs. We also use two PBSMT systems: Moses (Koehn et al., 2007) and Portage (Ueffing et al., 2007). Finally, we analyze in- and out-of-domain learning curves in order to better understand and investigate the growth rate.

The following section gives a quick overview of the models and systems we used in our experiments. We then briefly describe the experimental settings and data we used. Section 4 shows and analyzes the learning curves we obtained on French-English and on Chinese-English, and section 5 presents our results on the relative importance of LM and TM in the performance increase.

2 Translation Models and Systems

The standard phrase-based machine translation systems which we analyse here rely on a log-linear model and a set of baseline features functions. Translations of a source sentence \mathbf{e} is obtained by:

$$\hat{\mathbf{e}}(\mathbf{f}) = \operatorname{argmax}_{\mathbf{e}} \sum_i \lambda_i h_i(\mathbf{e}, \mathbf{f}).$$

where the $h_i(\mathbf{e}, \mathbf{a}, \mathbf{f})$ are *feature functions* involving both the source and target sentences, and the λ_i are the weights of those feature functions. Typical examples of feature functions that compose a basic phrase-based MT system are:

- phrase translation feature, e.g.:
 $h_T(\mathbf{e}, \mathbf{f}) = \sum_k \log p(f_k | e_k);$
- language model feature, e.g.:
 $h_L(\mathbf{e}, \mathbf{f}) = \sum_j \log p(w_j | w_{j-1}, \dots, w_1)$
- distortion feature, e.g.:
 $h_D(\mathbf{e}, \mathbf{f}) = \sum_k \|\text{start}(f_k) - \text{end}(f_{k-1}) - 1\|$
- Word penalty and/or phrase penalty features.

where e_k and f_k are contiguous subsequences of words in the source and target sentences and w_j are target words.

Parameter estimation is crucial for both the translation and language model features. Conditional probabilities are estimated from a large training corpus using empirical counts and various smoothing strategies. In addition, the weights λ_i are also estimated from a (usually disjoint) corpus of source and target sentence pairs. The size and composition of the training data will therefore have an influence on the quality of the predictions $\hat{\mathbf{e}}$ through the estimation of both the log-linear parameters and the feature functions.

Note that alternate models such as hierarchical (Chiang, 2007) or syntax based (Zollman and Venugopal, 2006) have been developed and could also be studied. However their use on the large scale necessary for creating accurate learning curves would require solving a number of practical issues and we focus instead on the straight PBSMT approach, which has been shown in recent MT evaluations (Callison-Burch et al., 2009; Callison-Burch et al., 2011) to offer competitive performance.

2.1 PBSMT Software

Several software packages are available for training PBSMT systems. In this work, we use Moses (Koehn et al., 2007) and Portage (Ueffing et al., 2007), two state-of-the-art systems capable of learning translation tables, language models and decoding parameters from one or several parallel corpora. Moses is a complete open-source phrase-based translation toolkit available for academic purposes, while Portage is a similar package, available to partners of the National Research Council Canada.

Given a parallel training corpus, both perform basic preprocessing (tokenization, lowercasing, etc.) if necessary, and build the various components of the model. Both use standard external tools for training the language model, such as SRILM (Stolcke, 2002). Moses uses GIZA++ (Och and Ney, 2003) for word alignments, while Portage uses an in-house IBM model and HMM implementation. The parameters of the log-linear models are tuned using minimum error rate training (MERT, (Och, 2003)).

Earlier experiments performed on the Europarl corpus with both systems showed (Turchi et al., 2011) that despite small differences in observed performance, both systems produce very similar learning curves.

3 Experimental Setting

3.1 Corpora

We experiment with large corpora in two language pairs: French-English and Chinese-English.

For French-English, we use the Giga corpus (Callison-Burch et al., 2009) to provide the training, development and one in-domain test set. As out-of-domain test set, we use two different samples from the EMEA corpus (Tiedemann, 2009), which contains parallel documents from the European Medicines Agency, and two News test sets from the 2009 (Callison-Burch et al., 2009) and 2011 (Callison-Burch et al., 2011) editions of the Workshop on Statistical Machine Translation, containing news articles drawn from a variety of sources and languages in different periods and translated by human translators.

For Chinese-English, we use various parallel corpora obtained from the Linguistic Data Consortium for the NIST evaluations. The training, development and in-domain test sets are sampled from the United Nations corpus (UN,

src	Training Set	Sentences	Words
fr	Giga	18.276 M	482,744k
ch	UN	4.968 M	163,960k
Dev. Set			
fr	Giga	1,000	62k
ch	UN	2,000	32k
Test Set			
fr	Giga	3,000	109k
fr	Emea	3,051	45.4k
fr	Emea2	3,051	46.7k
fr	News 2009	2,489	70.7k
fr	News 2011	3,030	85.1k
ch	UN	10,000	332k
ch	HKH	5,000	153k
ch	NIST	1,357	42k
ch	News	10,317	320k

Table 1: Number of sentences and words (source side) for the training, dev and various test sets.

LDC2004E12). As out-of-domain test sets, we used a sample from the Hong-Kong Hansard (HKH, LDC2000T50), a corpus of Chinese News translations (LDC2005T06) and the NIST 2008 Chinese evaluation set (LDC2009E09). Basic statistics are given in Table 1.

In order to analyse the way MT performance evolves with increasing data, we subsample (without replacement) the training sets at various sizes, averaging performance (estimated by BLEU, cf. section 3.3) over several samples. Learning curves are then obtained by plotting the average BLEU score, with error bars, as training data sizes increases. The relatively large amount of sentences in most test sets will allow us to reduce the uncertainty on the estimated test error, therefore producing smaller error bars.

For the French-English data, we followed the methodology proposed in (Turchi et al., 2008) and sampled 20 different sizes representing 5%, 10%, etc. of the original training corpus. Due to the large size of the corpus, only three random subsets are sampled at each size. For the Chinese-English dataset, we sampled at sizes corresponding to one half, one quarter, etc. down to $1/512^{th}$ ($\sim 0.2\%$) of the full size. At each size we produced 10 random samples. Each random subsample produces a model (cf. below) which is used to translate the various test sets. The learning curves will therefore cover the range from around 900 thousand

to 18.3 million sentences for French-English, and from around 10 thousand to 5 million sentences for Chinese-English.

Note that the corpora, in addition to differing in language pair, also differ in domain and homogeneity. The UN data contains only material from the United Nations, covering a wide range of themes, but fairly homogeneous in terms of style and genre. The Giga corpus, on the other hand, was obtained through a targeted web crawl of bilingual web sites from the Canadian government, the European Union, the United Nations, and other international organizations. In addition to covering a wide range of themes, they also contain documents with different styles and genres. Moreover, we estimated in an independent study that the rate of misaligned sentence pairs in the Giga corpus is as high as 13%.

The choice of source languages is driven by the desire to analyze two very different languages and by the scarcity of large publicly available bilingual corpora, especially outside European languages. UN data is also available in Russian or Arabic, but by definition would be the same domain and homogeneity as the Chinese-English corpus.

3.2 PBSMT System Training

For both systems, Portage and Moses, we used the basic configuration and features: phrase extraction is done by aligning the corpus at the word level (IBM models 1, 2, 3 and 4 for Moses, HMM and IBM2 models for Portage), the parameters of the log-linear model are set using an implementation of Och's MERT algorithm (Och, 2003), n-gram language modelling uses Kneser-Ney smoothing (3-gram using SRILM for Moses and 4-gram for Portage) and the maximum phrase length is 7 tokens. In Portage, phrase pairs were filtered so that the top 30 translations for each source phrase were retained. In both systems, the MERT algorithm was independently run on each sampled training set for each experiment.

Note that we expect that there will be differences in the quality of the translation depending on the source language. However, we are not so much interested in the actual translation performance as in the way this performance evolves with increasing data under various conditions.

3.3 Evaluation metrics

We report performance in terms of BLEU score (Papineni et al., 2001), the well accepted and

widespread automatic MT metric. We are well aware that maximizing BLEU may neither be necessary for, nor guarantee good translation performance, and that automatic MT metrics may not tell the whole story as far as translation quality is concerned. However, our systematic study aims at characterizing the behaviour of PBSMT systems that are built by maximizing such metrics, and this maximization is part of the learning system we analyze. Deriving learning curves for human evaluations of translation quality would be interesting, but is clearly impractical at this point.

4 Learning Curve Analysis

We now present the results obtained under the general framework outlined above.

We stress that in these experiments, we focus on the growth rate of the learning curves. In particular we are interested in 1) confirming that learning curves have logarithmic growth, and 2) possible differences between domains, languages and systems. A common, but poorly supported belief in PBSMT is that each doubling of the data yields a more or less constant increase in performance. In order to analyze and support this belief, we show all learning curves on a log scale, where we can check if the curve has a linear behaviour.

Note that sampling without replacement results in an increasing overlap between samples as their sizes grow. The size of the error bars therefore decreases as the training set size grows, because the training sets, and therefore the resulting models, are not independent. This must be kept in mind, although we still believe that the presence of error bars helps to better understand the stability of the MT system's performance.

The resulting learning curves are shown in Figures 1 and 2 for the French-English and Chinese-English data, respectively. The plots show the performance, averaged over samples (marks, connected with dotted lines), the error bars (vertical lines) indicating the natural variance in the performance, and a least-squares linear fit of these points (dashed or solid line). It is very clear that the learning curves are almost exactly linear on the log scale in most cases (Chinese-English and most French-English curves). The EMEA 2 and News 2009 curves display a worse fit, but the empirical results are within error bars of the linear fit, showing that the deviation from linearity is not statistically significant. The instability in these last two curves

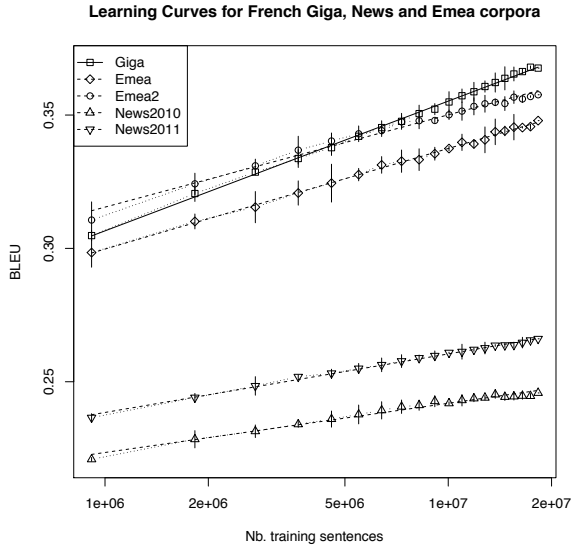


Figure 1: French-English learning curves obtained using the Giga corpus for training Moses on five test sets: one in-domain and four out-of-domain.

may actually be due to the fact mentioned earlier that the dependency between the performance estimates increases for large training sizes, which may lead to an increasing bias in the average.

These results confirm the findings of (Turchi et al., 2008) and extend them to more language pairs and much larger data sizes. These experiments supports the following claims:

- The increase in performance for PBSMT systems is essentially constant for each doubling of the data, over a wide range of training data sizes. Note that the growth does not seem to slow down as we near 20M training sentence pairs.
- A corollary of that first claim is that minor, even statistically significant increases in performance due to model “tweaking” are likely to be dwarfed by moderate increases in data sizes. For our Chinese system, for example, a 10% increase in data produces a 0.43 BLEU gain.
- On a linear scale, however, the addition of massive amounts of data from the same domain will result in diminishing improvements (“diminishing returns”) in the performance after an initial fast growth (Turchi et al., 2008; Bloodgood and Callison-Burch, 2010).
- Interestingly, the general shape of the learn-

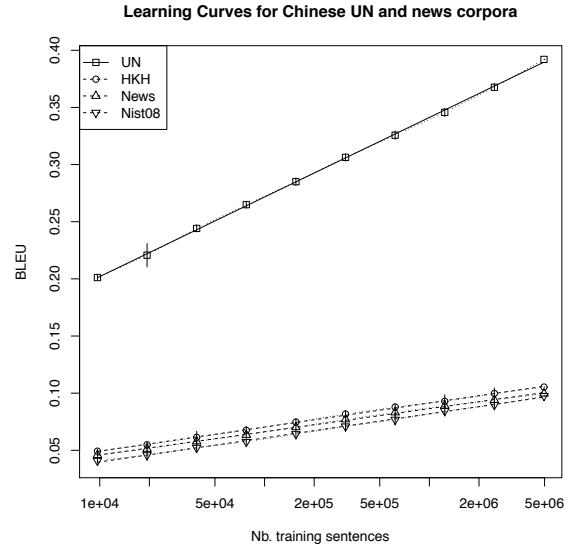


Figure 2: Chinese-English learning curves obtained using the UN corpus for training Portage on four test sets: one in-domain and three out-of-domain.

ing curves is essentially the same across different language pairs, different PBSMT systems, and also over different sources of test data (in-domain or out-of-domain).

- In particular, although the *performance* on out-of-domain data may greatly suffer (cf. Figure 2), the rate of increase is still linear in the log domain, up to large data sizes.

In order to quantify these findings, we estimate the gain per each doubling of the training set size by fitting a simple linear model on the learning curves in the log domain. For the Chinese-English data, each doubling of the data yields a gain of around 2.1 BLEU points on the in-domain data, and only 0.6 on the out-of-domain test sets. For the French-English data, the BLEU gain per training data doubling is around 1.5 points for the in-domain data, 1.1 for the EMEA test sets and 0.6 for the News test sets.

One may wonder why the out-of-domain EMEA test sets yield such high learning curves. Although the EMEA data comes from a European agency, we have verified that the sentences it contains are not contained in the Giga corpus. However, it turns out that the EMEA data is actually fairly easy to translate. The language is relatively constrained and repetitive, sentences are much shorter (on average ~ 15 words against more than 28 for the other

corpora), and the number of out-of-vocabulary words much lower than in the other test sets.

By contrast, all out-of-domain learning curves on Chinese-English are much lower than the in-domain curves (we have corroborated this with a dozen different test sets taken from various sources available for NIST evaluations, but omitted here for clarity). We believe this reflects differences between the sources of our training data. The UN corpus covers a number of topics but is very homogeneous and rather limited in genre. By contrast, the Giga corpus contains a wide range of documents covering many themes and genres. As a consequence, any test set that does not come from the UN data is distinctively different and “far” out-of-domain. On the other hand, it is not inconceivable that even for French text that does not come from the same sources, the larger and more diverse Giga corpus provides some measure of overlap in topics and genre.

5 Relative Importance of TM and LM

In the previous Section, experiments have been run using the same training set size for language and translation models. However, there is a large difference in the cost of training data for language and translation models. The former can be trained using monolingual data only while the latter requires bilingual texts. In recent years, several parallel corpora have been produced, e.g. Europarl (Koehn, 2005), JRC Acquis (Steinberger et al., 2006), and others, but they are not comparable to the amount of freely available monolingual data.

(Brants et al., 2007) have shown that performance improves linearly with the log of the number of tokens in the language model training set when this quantity is huge (from billions to trillions of tokens). In this section, we are interested in understanding the trade-off between the training data size used to build language and translation models, as well as in how performance is affected by that difference. We propose a mathematical model to estimate the variation in BLEU score according to the size of the training data used by the language model vs. that use by the translation model. The previous section shows that the overall performance of a PBSMT system grows in the logarithm of the training data size. We therefore modelled this relation in the following way:

$$BLEU(d_{LM}, d_{TM}) = \alpha_{LM} * \log_2(d_{LM}) + \alpha_{TM} * \log_2(d_{TM}) + \epsilon$$

where d_{LM} is the amount of training data used to build the language model, d_{TM} is the amount of training data used to build the Translation Model. α_{LM} and α_{TM} are weighting factors that identify the contribution of language and translation training data to the BLEU score, and ϵ is the residual. Note that when $d_{LM} = d_{TM}$, we recover a simple logarithmic relationship between performance and data size, as illustrated in the previous section.

In order to evaluate the relation between the amount of training data used to build language and translation models we estimate α_{LM} and α_{TM} from data. We focus on the French-English data, and use the training data subsets at every 10% of the full data size (10%, 20%, etc.), using the same development and test sets as before. One instance of a PBSMT model is learned for each combination of language and translation training data sizes, and we compute the resulting BLEU on the test sets. We estimate the parameters α_{LM} and α_{TM} using multivariate linear regression based on least squares (Draper and Smith, 1981), with the BLEU scores as response variables and the log values of the LM and TM training sizes as explanatory variables. This is done for three French-English test sets: the in-domain Giga, Emea and News 2009. The Emea2 and News 2011 test sets were qualitatively very similar.

We estimated the weighting factors using all the data. The results in Table 2 empirically confirm the common belief that adding data to the translation model is more important than to the language model ($\alpha_{TM} > \alpha_{LM}$). The values of α_{LM} and α_{TM} vary across the test sets, and correspond to an increase of 1 to 1.3 BLEU point per doubling of the training data for the LM and 1.2 to 1.8 BLEU point per doubling for the TM. However, the ratio is rather stable, indicating that the relative importance of the TM w.r.t. the LM is stable across domains. Not surprisingly, the more similar the test set is to the training data, the larger is the BLEU point growth. Our results are qualitatively compatible with the observations reported in a tutorial by (Och, 2005), although the increments in BLEU with each doubling of the training data size are reported 0.5 and 2.5 points for the language and translation models, respectively, in the context of Arabic-English translation. The ratio we observed in our experiments is lot more favourable to the language model.

In order to validate this finding, we performed

Test Set	α_{LM}	α_{TM}	α_{TM}/α_{LM}
Giga	0.0133	0.0182	1.368
Emea	0.0134	0.0168	1.2563
News 2009	0.0097	0.0122	1.2532

Table 2: Empirical estimation of the contributions α_{LM} and α_{TM} of the LM and TM, respectively, (ϵ is smaller than 1×10^{-4}), in BLEU per \log_2 in size. Experiments have been performed independently on the three test sets.

two simple experiments where we added a fairly large, 10 million sentence corpus of monolingual data (not included in the Giga corpus) to our LM training data, starting with around 5 million sentence of bilingual data from the Giga corpus. This produced a 1.79 BLEU increase in performance on News 2009 and 1.38 BLEU increase on News 2011, which is roughly consistent with a tripling in LM training data size according to the rate estimated in Table 2 ($0.97 \times \log_2 3 \approx 1.54$).

6 Discussion

Although limited to two language pairs, our results investigate the behaviour of PBSMT as a learning system over a range of different conditions: very different language pairs, in-domain and out-of-domain data, differing level of corpus homogeneity. etc. We emphasize that obtaining systematic and accurate learning curves requires a significant effort, even with an high performance computing architecture (Figure 2 requires translating more than 3 million test sentences with 91 models).

The learning curves obtained here suggest that, on an absolute (linear) scale, performance gains per fixed amount of additional data decrease. The diminishing improvements in performance after an early fast growth was also reported by (Uszkoreit et al., 2010) who mined the Web to extract very large sets of parallel documents. Starting with two corpora (French/Spanish to English) similar in dimensions to the Giga training set and using the News 2009 test sets, they report that adding more than 4,800 M words from a different domain resulted in relative small performance gains (< 2 BLEU points).

On a log-scale, on the other hand, there is no sign that performance gains decrease as we keep doubling the training corpus size, at least up to 20M sentence pairs. Note that although usual

MT metrics have natural bounds (0 for error-based metrics such as TER, 1 for BLEU), this has little practical relevance to the results presented here. Indeed, assuming we could extrapolate the very stable growth rates observed here, taking the performance of the out-of-domain HKH test set to where the in-domain UN data starts (for 10k sentence pairs only) would require close to 180 *billion* sentence pairs. For all practical purpose, we would run out of data long before we reached even half of the theoretical maximum BLEU score.

Finally, the analysis of the relative importance of TM and LM estimation shows that the translation model contributes about 30% more to the increase in performance than the language model. Considering the crucial role of the phrase table in the translation process, this contribution is maybe less than one would expected. This means that the massive addition of training data to the language model has a substantial impact in terms of performance, as shown by (Brants et al., 2007). It is interesting that the ratio of α_{TM} and α_{LM} seems stable across different domains. The relation between the translation and language model contribution to the final BLEU score does not change whether we translate in- or out-of-domain data.

7 Conclusion

Using state-of-the-art Phrase-Based Statistical Machine Translation packages and large parallel corpora, we derived very accurate learning curves for a number of language pairs and domains. Our results suggest that performance, as measured by BLEU, increases by a constant factor for each doubling of the data. Although that factor varies depending on corpus and language pair, this result seems consistent over all experimental conditions we tried. Our findings confirm the results reported for example by (Brants et al., 2007) and (Och, 2005), and extend and complete the findings of (Turchi et al., 2008).

We propose a study of how performance is influenced by difference sizes of data used for training the language and translation models. Our model gives more importance to the translation model than the language model every doubling of training data, but we are lot more favourable to the language model compared to other reported results in the literature.

Even if we do not currently provide any result that is immediately actionable to improve current

PBSMT performance, we believe it is important to analyse and quantify the way Machine Translation systems learn. In addition, the markedly different rates of performance increase for in-domain and out-of-domain data may provide a clue to better characterise the suitability of a MT model to translate a given test set. Investigating features that help us differentiate out-of-domain from in-domain data may prove very useful to improve practical performance of PBSMT systems.

References

- Y. Al-Onaizan and J. Curin and M. Jahr and K. Knight et al. 1999. *Statistical Machine Translation: Final Report*. JHU 1999 Summer Workshop on Language Engineering, CSLP.
- M. Bloodgood and C. Callison-Burch. 2010. *Bucking the trend: large-scale cost-focused active learning for statistical machine translation*. 48th Meeting of the ACL, pp. 854–864.
- T. Brants and A. C. Papat and P. Xu and F. J. Och and J. Dean. 2007. *Large Language Models in Machine Translation*. Proc. EMNLP-CoNLL 2007, pp. 858–867.
- C. Callison-Burch, and P. Koehn and C. Monz and J. Schroeder. 2009. *Findings of the 2009 Workshop on Statistical Machine Translation*. Fourth Workshop on Statistical Machine Translation, pp. 1–28.
- C. Callison-Burch, and P. Koehn and C. Monz and O. Zaidan. 2011. *Findings of the 2011 Workshop on Statistical Machine Translation*. Sixth Workshop on Statistical Machine Translation, pp. 22–64.
- D. Chiang. 2007. *Hierarchical Phrase-Based Translation*. Computational Linguistics, 33(2):201–228.
- N.R. Draper and H. Smith. 1981. *Applied regression analysis*. Wiley, New York, USA.
- G. Haffari and M. Roy and A. Sarkar. 2009. *Active Learning for Statistical Phrase-based Machine Translation*. Proc. HLT-NAACL, pp. 415–423.
- P. Koehn. 2005. *Europarl: A Parallel Corpus for Statistical Machine Translation*. Proc. MT-Summit X, pp. 79–86.
- P. Koehn and H. Hoang and A. Birch and C. Callison-Burch et al. 2007. *Moses: Open source toolkit for statistical machine translation*. 45th Meeting of the ACL demo, pp. 177–180.
- P. Koehn and F. J. Och and D. Marcu. 2003. *Statistical phrase-based translation*. Proc. NAACL-HLT, pp. 48–54. Edmonton, Canada.
- F. J. Och 2005. *Statistical machine translation: Foundations and recent advances*. Proc. MT-Summit X tutorial.
- F. J. Och 2003. *Minimum error rate training in statistical machine translation*. 41st Meeting of the ACL, pp. 160–167.
- F. J. Och and H. Ney 2003. *A Systematic Comparison of Various Statistical Alignment Models*. Computational Linguistics, 29(1): pages 19–51. Sapporo, Japan.
- F. J. Och and H. Ney 2002. *Discriminative training and maximum entropy models for statistical machine translation*. 40th Meeting of the ACL, pp. 295–302.
- K. Papineni and S. Roukos and T. Ward and W. J. Zhu 2002. *BLEU: a method for automatic evaluation of machine translation*. 40th Meeting of the ACL, pp. 311–318.
- A. Stolcke. 2002. *SRILM – An extensible language modeling toolkit*. Intl. Conf. Spoken Language Processing.
- R. Steinberger and B. Pouliquen and A. Widiger and C. Ignat and T. Erjavec and D. Tufiş and D. Varga. 2006. *The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages*. 5th LREC, pp. 2142–2147.
- B. Suresh. 2010 *Inclusion of large input corpora in Statistical Machine Translation*. Technical report, Stanford University.
- J. Tiedemann. 2009. *News from OPUS—A Collection of Multilingual Parallel Corpora with Tools and Interfaces*. RANLP (vol V), pp. 237–248.
- M. Turchi and T. DeBie and N. Cristianini. 2008. *Learning Performance of a Machine Translation System: a Statistical and Computational Analysis*. Third Workshop on Statistical Machine Translation, pp. 35–43.
- M. Turchi and T. DeBie and C. Goutte and N. Cristianini. 2012. *Learning to Translate: a statistical and computational analysis*. Advances in Artificial Intelligence, in press.
- N. Ueffing and M. Simard and S. Larkin and J. Howard Johnson. 2007. *NRC’s PORTAGE system for WMT*. Second Workshop on Statistical Machine Translation, pp. 185–188.
- J. Uszkoreit and J.M. Ponte and A.C. Papat and M. Dubiner. 2010. *Large scale parallel document mining for machine translation*. 23rd COLING, pp. 1101–1109.
- A. Zollman and A. Venugopal. 2006. *Syntax augmented machine translation via chart parsing*. Proc. NAACL Workshop on Machine Translation.
- R. Zens and F. J. Och and H. Ney. 2002. *Phrase-Based Statistical Machine Translation*. Proc. KI ’02: Advances in Artificial Intelligence, pp. 18–32.

Discriminative Reordering Extensions for Hierarchical Phrase-Based Machine Translation

Matthias Huck and Stephan Peitz and Markus Freitag and Hermann Ney

Human Language Technology and Pattern Recognition Group
Computer Science Department
RWTH Aachen University
D-52056 Aachen, Germany
<surname>@cs.rwth-aachen.de

Abstract

In this paper, we propose novel extensions of hierarchical phrase-based systems with a discriminative lexicalized reordering model. We compare different feature sets for the discriminative reordering model and investigate combinations with three types of non-lexicalized reordering rules which are added to the hierarchical grammar in order to allow for more reordering flexibility during decoding. All extensions are evaluated in standard hierarchical setups as well as in setups where the hierarchical recursion depth is restricted. We achieve improvements of up to +1.2 %BLEU on a large-scale Chinese→English translation task.

1 Introduction

Lexicalized reordering models are a common component of standard phrase-based machine translation systems. In hierarchical phrase-based machine translation, reordering is modeled implicitly as part of the translation model. Hierarchical phrase-based decoders conduct phrase reorderings based on a one-to-one relation between the non-terminals on source and target side within hierarchical translation rules. Non-terminals on source and target side are linked if they result from the same valid phrase being cut out at their position during phrase extraction. Usually neither explicit lexicalized reordering models nor additional mechanisms to perform reorderings that do not result from the application of hierarchical rules are integrated into hierarchical decoders.

In this work, we augment the grammar with more flexible reordering mechanisms based on additional non-lexicalized reordering rules and integrate a discriminative lexicalized reordering model. This kind of model has been shown to perform well when being added to the log-linear model combination of standard phrase-based systems. We present an extension of a hierarchical decoder with the discriminative reordering model and evaluate it in setups with the usual hierarchical grammar as well as in setups with a shallow hierarchical grammar. The shallow grammar restricts the depth of the hierarchical recursion. Two different feature sets for the discriminative reordering model are examined. We report experimental results on the large-scale NIST Chinese→English translation task. The best translation quality is achieved with combinations of the extensions with additional reordering rules and with the discriminative reordering model. The overall improvement over the respective baseline system is +1.2 %BLEU / -0.6 %TER absolute in the standard setup and +1.2 %BLEU / -0.5 %TER absolute in the shallow setup.

2 Related Work

Hierarchical phrase-based translation was proposed by Chiang (2005). Iglesias et al. (2009) and in a later journal publication Gispert et al. (2010) present a way to limit the recursion depth for hierarchical rules by means of a modification to the hierarchical grammar. Their work is of interest to us as a limitation of the recursion depth affects the search space and in particular the reordering capabilities of the system. It is therefore basically antipodal to some of the techniques presented in this paper, which allow for even more flexibility during the search process by extending the grammar with

specific non-lexicalized reordering rules. Combinations of both techniques are possible, though, and in fact Iglesias et al. (2009) also investigate a maximum phrase jump of 1 (MJ1) reordering model. In the MJ1 experiment, they include a swap rule, but simultaneously withdraw all hierarchical phrases.

Vilar et al. (2010) extend a hierarchical phrase-based system with non-lexicalized rules that permit jumps across whole blocks of symbols and report improvements on a German→English Europarl task. Their technique is inspired by conventional phrase-based IBM-style reordering (Zens et al., 2004). In an Arabic→English NIST setup, Huck et al. (2011) try a similar reordering extension, but conclude that it is less helpful for their task. Other groups attempt to attain superior modeling of reordering effects in their hierarchical systems by examining syntactic annotation, e.g. Gao et al. (2011).

He et al. (2010a) combine an additional BTG-style swap rule with a maximum entropy based lexicalized reordering model and achieve improvements on the Chinese→English NIST task. Their approach is comparable to ours, but their reordering model requires the training of different classifiers for different rule patterns (He et al., 2010b). Extracting training instances separately for several patterns of hierarchical rules yields a dependence on the phrase segmentation. In the more general approach we propose, the definition of the features is independent of the phrase boundaries on the source side.

In standard phrase-based systems, lexicalized reordering models are a commonly included component. A widely used variant is the orientation model as implemented in the Moses toolkit (Tillmann, 2004; Koehn et al., 2007) which distinguishes monotone, swap, and discontinuous phrase orientations. Galley and Manning (2008) suggest a refinement of the same model. A discriminatively trained lexicalized reordering model as the one employed by us has been examined in a standard phrase-based setting by Zens and Ney (2006).

3 Shallow-1 Grammar

Gispert et al. (2010) propose a limitation of the recursion depth for hierarchical rules with shallow- n grammars. The main benefit of the limitation is a gain in decoding efficiency. Moreover, the modification of the grammar to a shallow version re-

stricts the search space of the decoder and may be convenient to prevent overgeneration. We will investigate reordering extensions to both standard hierarchical systems and systems with a shallow-1 grammar, i.e. a grammar which limits the depth of the hierarchical recursion to one. We refer to this kind of rule set and the parses produced with such a grammar as *shallow*, in contrast to the standard rule set and parses which we denote as *deep*.

In a shallow-1 grammar, the generic non-terminal X of the standard hierarchical approach is replaced by two distinct non-terminals XH and XP . By changing the left-hand sides of the rules, lexical phrases are allowed to be derived from XP only, hierarchical phrases from XH only. On all right-hand sides of hierarchical rules, the X is replaced by XP . Gaps within hierarchical phrases can thus be filled with contiguous lexical phrases only, not with hierarchical phrases. The initial rule is substituted with

$$\begin{aligned} S &\rightarrow \langle XP^{\sim 0}, XP^{\sim 0} \rangle \\ S &\rightarrow \langle XH^{\sim 0}, XH^{\sim 0} \rangle, \end{aligned} \quad (1)$$

and the glue rule is substituted with

$$\begin{aligned} S &\rightarrow \langle S^{\sim 0} XP^{\sim 1}, S^{\sim 0} XP^{\sim 1} \rangle \\ S &\rightarrow \langle S^{\sim 0} XH^{\sim 1}, S^{\sim 0} XH^{\sim 1} \rangle. \end{aligned} \quad (2)$$

4 Reordering Rules

In this section we describe three types of reordering extensions to the hierarchical grammar. All of them add specific non-lexicalized reordering rules which facilitate a more flexible arrangement of phrases in the hypotheses. We first present a simple swap rule extension (Section 4.1), then we suggest two different extensions with several additional rules that allow for more complex jumps (Section 4.2) or very constrained jumps (Section 4.3). Furthermore, variants for deep and shallow grammars are proposed.

4.1 Swap Rule

4.1.1 Swap Rule for Deep Grammars

In a deep grammar, we can bring in more reordering capabilities by adding a single swap rule

$$X \rightarrow \langle X^{\sim 0} X^{\sim 1}, X^{\sim 1} X^{\sim 0} \rangle \quad (3)$$

supplementary to the standard initial rule and glue rule. The swap rule allows adjacent phrases to be transposed.

An alternative with a comparable effect would be to remove the standard glue rule and to add two rules instead, one of them being as in Equation (3) and the other a monotonic concatenation rule for the non-terminal X which is symmetric to the swap rule. The latter rule acts as a replacement for the glue rule. This is the approach He et al. (2010a) take. Our approach to keep the standard glue rule has however one advantage: We are still able to apply a maximum length constraint to X . The maximum length constraint restricts the length of the yield of a non-terminal. The lexical span covered by X is typically restricted to 10 to make decoding less demanding in terms of computational resources. We would still be able to add a monotonic concatenation rule to our grammar in addition to the standard glue rule. Its benefit is that it entails more symmetry in the grammar. In our variant, sub-derivations which result from applications of the swap rule can fill the gap within hierarchical phrases, while no mechanism to carry out the same in a monotonic manner is available. In the deep grammar, we refrain from adding a monotonic concatenation rule as recursive embeddings are possible anyway. We nevertheless tried the variant with the additional monotonic concatenation rule in a supplementary experiment (cf. Section 6.2.2) to make sure that our assumption that this rule is dispensable is correct. We were not able to obtain improvements over the setup with the swap rule only.

4.1.2 Swap Rule for Shallow Grammars

In a shallow grammar, several directions of integrating swaps are possible. We decided to add a swap rule and a monotonic concatenation rule

$$\begin{aligned} XP &\rightarrow \langle XP^{\sim 0} XP^{\sim 1}, XP^{\sim 1} XP^{\sim 0} \rangle \\ XP &\rightarrow \langle XP^{\sim 0} XP^{\sim 1}, XP^{\sim 0} XP^{\sim 1} \rangle \end{aligned} \quad (4)$$

supplementary to the standard shallow initial rules and glue rules. The swap rule allows adjacent lexical phrases to be transposed, but not hierarchical phrases. Here, we could as well have used XH as the left-hand side of the rules. As we chose XP and thus allow for embedding of sub-derivations resulting from applications of the swap rule into hierarchical phrases, which is not possible with sub-derivations resulting from applications of hierarchical rules in a shallow grammar, we also include the monotonic concatenation rule for symmetry reasons. A constraint can again be

applied to the number of terminals spanned by both XP and XH . With a length constraint, building sub-derivations of arbitrary length by applying the rules from Equation (4) is impossible.

4.2 Jump Rules, Variant 1

Instead of employing a swap rule that transposes adjacent phrases, we can adopt more complex extensions to the grammar that implement jumps across blocks of symbols. Our first jump rules variant is inspired by Vilar et al. (2010), but is a generalization that facilitates an arbitrary number of blocks per sentence to be jumped across.

4.2.1 Jump Rules for Deep Grammars

In a deep grammar, to enable block jumps, we include the rules

$$\begin{aligned} S &\rightarrow \langle B^{\sim 0} X^{\sim 1}, X^{\sim 1} B^{\sim 0} \rangle && \dagger \\ S &\rightarrow \langle S^{\sim 0} B^{\sim 1} X^{\sim 2}, S^{\sim 0} X^{\sim 2} B^{\sim 1} \rangle && \dagger \\ B &\rightarrow \langle X^{\sim 0}, X^{\sim 0} \rangle && \\ B &\rightarrow \langle B^{\sim 0} X^{\sim 1}, B^{\sim 0} X^{\sim 1} \rangle && \ddagger \end{aligned} \quad (5)$$

in addition to the standard initial rule and glue rule. The rules marked with \dagger are jump rules that put jumps across blocks (B) on source side into effect. The rules with B on their left-hand side enable blocks that are skipped by the jump rules to be translated, but without further jumps. Reordering within these windows is just possible with hierarchical rules. Note that our rule set keeps the convenient property of the standard hierarchical grammar that the initial symbol S needs to be expanded in the leftmost cells of the CYK chart only.

A binary jump feature for the two jump rules (\dagger) may be added to the log-linear model combination of the decoder, as well as a binary feature that fires for the rule that acts analogous to the glue rule, but within blocks that is being jumped across (\ddagger). A maximum jump width can be established by applying a length constraint to the non-terminal B . A distance-based distortion model can also easily be implemented by computing the span width of the non-terminal B on the right-hand side of the jump rules at each application of one of them.

4.2.2 Jump Rules for Shallow Grammars

In a shallow grammar, block jumps are realized in the same way as in a deep one, but the number of rules that are required is doubled.

We include

$$\begin{aligned}
S &\rightarrow \langle B^{\sim 0} X P^{\sim 1}, X P^{\sim 1} B^{\sim 0} \rangle && \dagger \\
S &\rightarrow \langle B^{\sim 0} X H^{\sim 1}, X H^{\sim 1} B^{\sim 0} \rangle && \dagger \\
S &\rightarrow \langle S^{\sim 0} B^{\sim 1} X P^{\sim 2}, S^{\sim 0} X P^{\sim 2} B^{\sim 1} \rangle && \dagger \\
S &\rightarrow \langle S^{\sim 0} B^{\sim 1} X H^{\sim 2}, S^{\sim 0} X H^{\sim 2} B^{\sim 1} \rangle && \dagger \\
B &\rightarrow \langle X P^{\sim 0}, X P^{\sim 0} \rangle && \\
B &\rightarrow \langle X H^{\sim 0}, X H^{\sim 0} \rangle && \\
B &\rightarrow \langle B^{\sim 0} X P^{\sim 1}, B^{\sim 0} X P^{\sim 1} \rangle && \ddagger \\
B &\rightarrow \langle B^{\sim 0} X H^{\sim 1}, B^{\sim 0} X H^{\sim 1} \rangle && \ddagger
\end{aligned} \tag{6}$$

in addition to the standard shallow initial rules and glue rules.

4.3 Jump Rules, Variant 2

As a second jump rules variant, we try an approach that follows (Huck et al., 2011) and allows for very constrained reorderings. At most one contiguous block per sentence can be jumped across in this variant.

In a deep grammar, to enable constrained block jumps with at most one jump per sentence, we replace the initial and glue rule by the rules given in Equation (7):

$$\begin{aligned}
S &\rightarrow \langle M^{\sim 0}, M^{\sim 0} \rangle \\
S &\rightarrow \langle S^{\sim 0} M^{\sim 1}, S^{\sim 0} M^{\sim 1} \rangle && \ddagger \\
S &\rightarrow \langle B^{\sim 0} M^{\sim 1}, M^{\sim 1} B^{\sim 0} \rangle && \dagger \\
M &\rightarrow \langle X^{\sim 0}, X^{\sim 0} \rangle && \\
M &\rightarrow \langle M^{\sim 0} X^{\sim 1}, M^{\sim 0} X^{\sim 1} \rangle && \ddagger \\
B &\rightarrow \langle X^{\sim 0}, X^{\sim 0} \rangle && \\
B &\rightarrow \langle B^{\sim 0} X^{\sim 1}, B^{\sim 0} X^{\sim 1} \rangle && \ddagger
\end{aligned} \tag{7}$$

In these rules, the M non-terminal represents a block that will be translated in a monotonic way, and the B is a ‘‘back jump’’. We omit the exposition for shallow grammars as deducing the shallow from the deep version of the rules is straightforward from our previous explanations.

We add a binary feature that fires for the rules that act analogous to the glue rule (\ddagger). We further conform to the approach of Huck et al. (2011) by additionally including a distance-based distortion model (*dist. feature*) that is computed during decoding whenever the back jump rule (\dagger) is applied.

5 Discriminative Reordering Model

Our discriminative reordering extensions for hierarchical phrase-based machine translation systems integrate a discriminative reordering model

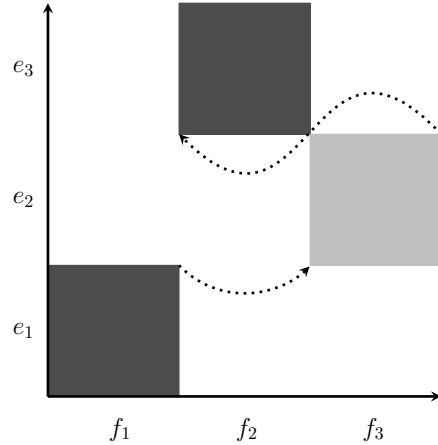


Figure 1: Illustration of an embedding of a lexical phrase (light) in a hierarchical phrase (dark), with orientations scored with the neighboring blocks.

that tries to predict the orientation of neighboring blocks. We use two orientation classes *left* and *right*, in the same manner as described by Zens and Ney (2006). The reordering model is applied at the phrase boundaries only, where words which are adjacent to gaps within hierarchical phrases are defined as boundary words as well. The orientation probability is modeled in a maximum entropy framework. We investigate two models that differ in the set of feature functions:

discrim. RO (src word) The feature set of this model consists of binary features based on the source word at the current source position.

discrim. RO (src+tgt word+class) The feature set of this model consists of binary features based on the source word and word class at the current source position and the target word and word class at the current target position.

Using features that depend on word classes provides generalization capabilities. We employ 100 automatically learned word classes which are obtained with the `mkcls` tool on both source and target side.¹ The reordering model is trained with the Generalized Iterative Scaling (GIS) algorithm with the maximum class posterior probability as training criterion, and it is smoothed with a gaussian prior.

For each rule application during hierarchical decoding, we apply the reordering model at all

¹`mkcls` is distributed along with the GIZA++ package: <http://code.google.com/p/giza-pp/>

boundaries where lexical blocks are placed side by side within the partial hypothesis. For this purpose, we need to access neighboring boundary words and their aligned source words and source positions. Note that, as hierarchical phrases are involved, several block joinings may take place at once during a single rule application. Figure 1 gives an illustration with an embedding of a lexical phrase (light) in a hierarchical phrase (dark). The gap in the hierarchical phrase $\langle f_1, f_2 X^{\sim 0}, e_1 X^{\sim 0} e_3 \rangle$ is filled with the lexical phrase $\langle f_3, e_2 \rangle$. The discriminative reordering model scores the orientation of the lexical phrase with regard to the neighboring block of the hierarchical phrase which precedes it within the target sequence (here: right orientation), and the block of the hierarchical phrase which succeeds the lexical phrase with regard to the latter (here: left orientation).

The way we interpret reordering in hierarchical phrase-based translation keeps our model simple. We are basically able to treat the orientation of contiguous lexical blocks in almost exactly the same way as the orientation of phrases in standard phrase-based translation. We avoid the usage of multiple reordering models for different source and target patterns of rules that is done by He et al. (2010b).

6 Experiments

We present empirical results obtained with the additional swap rule, the jump rules and the discriminative reordering model on the Chinese→English 2008 NIST task.²

6.1 Experimental Setup

We employ the freely available hierarchical translation toolkit Jane (Vilar et al., 2010) to set up our systems. In our experiments, we use the cube pruning algorithm (Huang and Chiang, 2007) to carry out the search. A maximum length constraint of 10 is applied to all non-terminals but the initial symbol S . We work with a parallel training corpus of 3.0M Chinese-English sentence pairs (77.5M Chinese / 81.0M English running words). Word alignments are created by aligning the data in both directions with GIZA++ and symmetrizing the two trained alignments (Och and Ney, 2003). The language model is a 4-gram with modified Kneser-

Ney smoothing which was trained with the SRILM toolkit (Stolcke, 2002).

Model weights are optimized against BLEU with Minimum Error Rate Training on 100-best lists. We employ MT06 as development set to tune the model weights, MT08 is used as unseen test set. The performance of the systems is evaluated using the two metrics BLEU and TER. The results on the test set are checked for statistical significance over the baseline. Confidence intervals have been computed using bootstrapping for BLEU and Cochran’s approximate ratio variance for TER (Leusch and Ney, 2009).

6.2 Experimental Results

The empirical evaluation of our reordering extensions is presented in Table 1. We report translation results on both the development and the test corpus. The figures with deep and with shallow rules are set side by side in separate columns to facilitate a direct comparison between them. All the setups given in separate rows exist in a deep and a shallow variant.

The shallow baseline is a bit worse than the deep baseline. Adding discriminative reordering models to the baselines without additional reordering rules results in an improvement of up to +0.6 %BLEU / -0.6 %TER (in the deep setup). The *src+tgt word+class* feature set for the discriminative reordering model altogether seems to perform slightly better than the *src word* feature set. Adding reordering rules in isolation can also improve the systems, in particular in the deep setup with the swap rule or the second jump rules variant. However, extensions with both reordering rules and discriminative lexicalized reordering model provide the best results, e.g. +1.0 %BLEU / -0.5 %TER with the system with deep grammar, swap rule, binary swap feature and discrim. RO (src+tgt word+class) and +1.2 %BLEU / -0.5 %TER with the system with shallow grammar, swap rule, binary swap feature and discrim. RO (src+tgt word+class). The second jump rules variant performs particularly well in combination with a deep grammar and the discrim. RO (src+tgt word+class) model, with an improvement of +1.2 %BLEU / -0.6 %TER absolute over the deep baseline. This system provides the best translation quality of all the setups investigated in this paper. With a shallow grammar, the combinations of the discrim. RO with the swap rule outperforms both

²<http://www.itl.nist.gov/iad/mig/tests/mt/2008/>

	MT06 (Dev)				MT08 (Test)			
	deep		shallow		deep		shallow	
	BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]
Baseline	32.6	61.2	31.4	61.8	25.2	66.6	24.9	66.6
+ discrim. RO (src word)	32.9	61.3	31.6	61.8	25.4	66.3	25.2	66.6
+ discrim. RO (src+tgt word+class)	33.0	61.3	31.6	61.6	25.8	66.0	25.1	66.3
+ swap rule	32.8	61.7	31.8	62.1	25.8	66.6	25.0	67.0
+ discrim. RO (src word)	33.0	61.2	32.5	61.4	25.8	66.1	26.0	66.2
+ discrim. RO (src+tgt word+class)	33.1	61.2	32.6	61.4	26.0	66.1	26.1	66.3
+ binary swap feature	33.2	61.0	32.1	61.8	25.9	66.2	25.7	66.5
+ discrim. RO (src word)	33.1	61.3	32.4	61.4	26.0	66.1	26.1	66.3
+ discrim. RO (src+tgt word+class)	33.2	61.3	32.9	61.0	26.2	66.1	26.1	66.1
+ jump rules, variant 1	32.9	61.3	32.1	62.4	25.6	66.4	25.1	67.5
+ discrim. RO (src word)	32.9	61.1	31.9	62.0	25.8	66.0	25.1	66.9
+ discrim. RO (src+tgt word+class)	33.2	61.0	32.1	62.0	25.9	66.1	25.6	66.5
+ binary jump feature	32.8	61.3	31.9	61.7	25.7	66.3	25.2	66.7
+ discrim. RO (src word)	32.8	61.3	32.2	61.9	25.8	66.1	25.2	66.7
+ discrim. RO (src+tgt word+class)	33.1	61.2	32.3	62.0	26.0	66.1	25.5	66.7
+ jump rules, variant 2 + dist. feature	33.0	61.5	31.5	62.0	25.8	66.5	25.3	66.3
+ discrim. RO (src word)	33.2	60.8	31.6	61.9	26.2	65.8	25.2	66.4
+ discrim. RO (src+tgt word+class)	33.2	61.0	31.7	62.1	26.4	66.0	25.5	66.3

Table 1: Experimental results for the NIST Chinese→English translation task (truecase). On the test set, bold font indicates results that are significantly better than the baseline ($p < .1$).

jump rules variants.

We proceed with discussing some supplementary results obtained with the deep grammar that are not included in Table 1. The results for Sections 6.2.2 through 6.2.4 can be found in Table 2.

6.2.1 Dropping Length Constraints

In order to find out if we lose performance by applying the maximum length constraint of 10 to all non-terminals but the initial symbol S during decoding, we optimized systems with no length constraints. When we drop the length constraint in the baseline setup, we observe no improvement on the dev set and +0.3 %BLEU improvement on the test set. Dropping the length constraint in the system with deep grammar, swap rule, discrim. RO (src+tgt word+class) and binary jump feature results in +0.2 %BLEU / -0.2 %TER on the dev set, but no improvement on the test set.

6.2.2 Monotonic Concatenation Rule

In this experiment, we add a monotonic concatenation rule

$$X \rightarrow \langle X^{\sim 0} X^{\sim 1}, X^{\sim 0} X^{\sim 1} \rangle \quad (8)$$

as discussed in Section 4.1.1 to the system with deep grammar, swap rule, binary swap feature and

discrim. RO (src+tgt word+class). As we presumed, the monotonic concatenation rule does not improve the performance of our system.

6.2.3 Distance-Based Distortion Feature

Our second jump rules variant includes a distance-based distortion feature (*dist. feature*). To make sure that the good performance of the jump rules variant 2 extension compared to jump rules variant 1 is not simply due to this feature, we also tested it in the best setup with our first jump rules variant. Adding the distance-based distortion feature does not yield an improvement over that setup. We tried such a feature with the swap rule as well by just computing the length of the yield of the left-hand side non-terminal at each swap rule application. Here again, adding the distance-based distortion feature does not yield an improvement.

6.2.4 Discriminative Reordering for Reordering Rules Only

Instead of applying the discriminative reordering model at all rule applications, the model can as well be used to score the orientation of blocks only if they are placed side by side within the target sequence by selected rules. We conducted ex-

	deep			
	MT06 (Dev)		MT08 (Test)	
	BLEU [%]	TER [%]	BLEU [%]	TER [%]
Baseline	32.6	61.2	25.2	66.6
+ no length constraints	32.6	61.5	25.5	66.6
+ swap rule + bin. swap feat. + discrim. RO (src+tgt word+class)	33.2	61.3	26.2	66.1
+ no length constraints	33.4	61.1	26.2	66.3
+ monotonic concatenation rule	33.2	61.6	26.0	66.4
+ dist. feature	33.4	61.4	26.2	66.2
+ discrim. RO scoring restricted to swap rule	33.1	61.4	26.0	66.4
+ jump rules 1 + bin. jump feat. + discrim. RO (src+tgt word+class)	33.1	61.2	26.0	66.1
+ dist. feature	33.2	61.1	25.9	66.1
+ discrim. RO scoring restricted to jump rules	32.8	61.3	25.9	66.3

Table 2: Supplementary experimental results with the deep grammar (truecase).

	deep		shallow	
	Baseline	Best Swap System	Baseline	Best Swap System
used hierarchical phrases	25.8%	32.0%	17.8%	24.0%
used lexical phrases	45.8%	40.0%	47.6%	44.7%
used initial and glue rules	28.4%	26.8%	34.6%	29.5%
used swap rules	-	1.2%	-	1.8%
applied swap rule in sentences	-	295 (22%)	-	446 (33%)

Table 3: Statistics on the rule usage for the single best translation of the test set (MT08).

periments in which the discriminative reordering scoring is restricted to the swap rule or the explicit jump rules (marked as \dagger in Eq. 5), respectively. The result is in both setups slightly worse than the result with the discriminative reordering model applied to all rules.

6.3 Investigation of the Rule Usage

To figure out the influence of the swap rule on the usage of different types of rules in the translation process, we compare in Table 3 the baseline systems (deep and shallow) with the systems using the swap rule, binary swap feature and discrim. RO (denoted as *Best Swap System* in the table). As expected, the deep systems use in general more hierarchical phrases compared to the shallow setups. However, adding the swap rule causes an increased usage of hierarchical phrases and less applications of the glue rule. The swap rule by itself makes up the smallest part, but is employed in 22% (deep) and 33% (shallow) respectively of the 1357 test sentences.

6.4 Translation Examples

Figure 2 depicts a translation example along with

its decoding tree from our system with deep grammar, swap rule, binary swap feature and discrim. RO (src+tgt word+class). The example is taken from the MT08 set, with the four reference translations “*But it is actually very hard to do that.*”, “*However, it is indeed very difficult to achieve.*”, “*But to achieve this point is actually very difficult.*” and “*But to be truly frank is, in fact, very difficult.*”. The hypothesis does not match any of the references, but still is a fully convincing English translation. Note how the application of the swap rule affects the translation. Our baseline system with deep grammar translates the sentence as “*but to do this , it is in fact very difficult .*”.

7 Conclusion

We presented novel extensions of hierarchical phrase-based systems with a discriminative lexicalized reordering model. We investigated combinations with three variants of additional non-lexicalized reordering rules. Our approach shows significant improvements (up to +1.2 %BLEU) over the respective baselines with both a deep and a shallow-1 hierarchical grammar on a large-scale Chinese→English translation task.

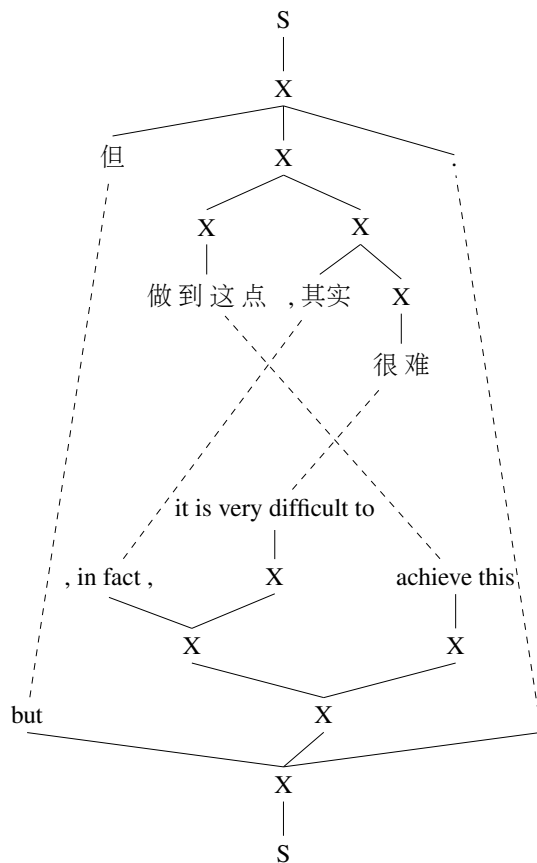


Figure 2: Translation example from the system with deep grammar, swap rule, binary swap feature and discrim. RO (src+tgt word+class).

Acknowledgments

This work was partly achieved as part of the Quaero Programme, funded by OSEO, French State agency for innovation, and partly funded by the European Union under the FP7 project T4ME Net, Contract No. 249119.

References

- Chiang, D. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *Proc. of the ACL*, pages 263–270, Ann Arbor, MI, June.
- Galley, M. and C. D. Manning. 2008. A Simple and Effective Hierarchical Phrase Reordering Model. In *Proc. of the EMNLP*, pages 847–855, Honolulu, Hawaii, October.
- Gao, Y., P. Koehn, and A. Birch. 2011. Soft Dependency Constraints for Reordering in Hierarchical Phrase-Based Translation. In *Proc. of the EMNLP*, pages 857–868, Edinburgh, Scotland, UK, July.
- Gispert, A. de, G. Iglesias, G. Blackwood, E. R. Banga, and W. Byrne. 2010. Hierarchical Phrase-Based Translation with Weighted Finite-State Transducers and Shallow-n Grammars. *Computational Linguistics*, 36(3):505–533.
- He, Z., Y. Meng, and H. Yu. 2010a. Extending the Hierarchical Phrase Based Model with Maximum Entropy Based BTG. In *Proc. of the AMTA*, Denver, CO, October/November.
- He, Z., Y. Meng, and H. Yu. 2010b. Maximum Entropy Based Phrase Reordering for Hierarchical Phrase-based Translation. In *Proc. of the EMNLP*, pages 555–563, October.
- Huang, L. and D. Chiang. 2007. Forest Rescoring: Faster Decoding with Integrated Language Models. In *Proc. of the ACL*, pages 144–151, Prague, Czech Republic, June.
- Huck, M., D. Vilar, D. Stein, and H. Ney. 2011. Advancements in Arabic-to-English Hierarchical Machine Translation. In *Proc. of the EAMT*, pages 273–280, Leuven, Belgium, May.
- Iglesias, G., A. de Gispert, E. R. Banga, and W. Byrne. 2009. Rule Filtering by Pattern for Efficient Hierarchical Translation. In *Proc. of the EACL*, pages 380–388, Athens, Greece, March.
- Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, et al. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proc. of the ACL*, pages 177–180, Prague, Czech Republic, June.
- Leusch, G. and H. Ney. 2009. Edit distances with block movements and error rate confidence estimates. *Machine Translation*, December.
- Och, F. J. and H. Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51, March.
- Stolcke, A. 2002. SRILM – an Extensible Language Modeling Toolkit. In *Proc. of the ICSLP*, Denver, CO, September.
- Tillmann, C. 2004. A Unigram Orientation Model for Statistical Machine Translation. In *Proc. of the HLT-NAACL: Short Papers*, pages 101–104.
- Vilar, D., D. Stein, M. Huck, and H. Ney. 2010. Jane: Open Source Hierarchical Translation, Extended with Reordering and Lexicon Models. In *Proc. of the ACL/WMT*, pages 262–270, Uppsala, Sweden, July.
- Zens, R. and H. Ney. 2006. Discriminative Reordering Models for Statistical Machine Translation. In *Proc. of the HLT-NAACL*, pages 55–63, New York City, June.
- Zens, R., H. Ney, T. Watanabe, and E. Sumita. 2004. Reordering Constraints for Phrase-Based Statistical Machine Translation. In *COLING '04: The 20th Int. Conf. on Computational Linguistics*, pages 205–211, Geneva, Switzerland, August.

Pivot-based Machine Translation between Statistical and Black Box systems

Antonio Toral

School of Computing
Dublin City University
Dublin, Ireland

atoral@computing.dcu.ie

Abstract

This paper presents a novel approach to pivot-based machine translation (MT): while the state-of-the-art uses two statistical systems, this proposal treats the second system as a black box. Our approach effectively provides pivot-based MT to target languages for which no suitable bilingual corpora are available to build statistical systems, as long as any other kind of MT system is available. We experiment with an algorithm that uses two features to find the best translation: the translation score provided by the first system and fluency of the final translation. Despite its simplicity, this approach yields significant improvements over the baseline, which translates the source sentences using the two MT systems sequentially. We have experimented with two scenarios, technical documentation in Romance languages and newswire in Slavic languages, obtaining 11.88% and 13.32% relative improvements in terms of BLEU, respectively.

1 Introduction

Pivot-based machine translation (MT) refers to the use of an intermediate language, called pivot language (PL), to translate from the source (SL) to the target language (TL). Unlike typical MT systems, which translate directly from SL to TL, pivot-based systems translate sequentially from SL to PL and then from PL to TL. The main motivation for building pivot-based MT systems is the lack of language resources for a language pair SL–TL, in

contrast with the availability of such resources for both language pairs SL–PL and PL–TL.

Much of the research carried out in pivot-based MT concentrates on a scenario where the translation both from SL to PL and from PL to TL is carried out by statistical machine translation systems (SMT). It is also often assumed that the developer has access not only to the output of the systems but also to their internal data structures. Hence, for these methods to work, bilingual corpora for both SL–PL and PL–TL are required in order to train the corresponding SMT systems.

Our research concentrates on pivot-based MT for cases where there is no access to the internals of the second system (PL to TL), i.e. we treat it as a black box: only the output translations produced by this system are available. Because of this our approach is applicable to a much broader set of scenarios than the current state-of-the-art; i.e. it can be applied when there is no access to the internals of the second system, which is the case for many online MT systems, or when the second system does not provide the required data (such as *n*-best lists), which is the case for many rule-based machine translation systems (RBMT).

The remainder of this paper is organised as follows. Section 2 presents an overview of the state-of-the-art for pivot-based MT. This is followed by the description of our methodology. Subsequently, we carry out the evaluation and present the results of the proposal. Finally, we conclude and outline lines of future work.

2 Related Work

Pivot-based strategies that use SMT systems can be classified into three categories (Wu and Wang, 2009): phrase table multiplication (also known as triangulation), transfer (also referred to as cascade)

and synthetic corpus.

Phrase table multiplication methods (Wu and Wang, 2007; Cohn and Lapata, 2007) induce a new SL–TL translation model by combining the corresponding translation probabilities of the translations models for SL–PL and PL–TL.

The transfer method (Utiyama and Isahara, 2007; Khalilov et al., 2008) translates the text in the SL to the PL using a SL–PL translation model and then to the TL using a PL–TL translation model. A source sentence s can be translated into n PL sentences. Each of these n sentences can then be translated into m TL sentences. Therefore we have $n \times m$ translation candidates which can be rescored using the translation scores from both the SL–PL and PL–TL models. The translation that gets the highest ranking is considered to be the best translation.

The synthetic corpus method (Gispert and Mariño, 2006; Bertoldi et al., 2008; Utiyama et al., 2008) obtains a SL–TL corpus using the SL–PL or the PL–TL corpora. One way to do this is to translate the PL sentences in the SL–PL corpus into TL with the PL–TL system. Another possibility is to translate the PL sentences in the PL–TL corpus into SL with the SL–PL system. Obviously, both methods could be applied and the two resulting synthetic corpora be merged into a single SL–TL corpus.

Wu and Wang (2009) compare the performance of the phrase table multiplication, transfer and synthetic corpus methods. They also present a hybrid method that combines RBMT and SMT to fill up the data gap, assuming the SL–PL and PL–TL corpora are independent. In this approach, RBMT systems are used to translate the PL sentences in the SL–PL or PL–TL corpus into TL or SL sentences, respectively. Then these synthetic corpora can be used to enrich the initial SL–PL and PL–TL corpora so that the SMT systems can take advantage of the availability of additional bilingual data.

System combination has also been exploited to improve pivot-based MT. Wu and Wang (2009) build systems following the three aforementioned approaches (phrase table multiplication, transfer and synthetic corpus) and combine the outputs produced by the different systems. Leusch et al. (2010) generate intermediate translations in several PLs, then translate them separately into the TL, and finally generate a consensus translation out of all of them.

The closest research strand to the work presented in this paper is the transfer method. The main difference is that the transfer method uses n -best lists and features from both systems and language pairs (SL–PL and PL–TL) in order to obtain the best translation while our proposal only has access to the n -best list and to internal features of the MT system for the language pair SL–PL. In our approach we treat the MT system for PL–TL as a black box. Because of this its application is wider: while the state-of-the-art requires access to the internals of this system, ours does not.

3 Methodology

In this section we introduce our methodology to perform pivot-based MT. We use a SMT system to translate from SL to PL (System1 from here onwards) and any kind of MT system to translate from PL to TL (System2).

For each source sentence we obtain the best n translations (n -best list) produced by System1 from SL to PL. Then we translate this n -best list from PL to TL using System2. Finally we select the best of these n translations in TL, using features from three different sources: (i) system internal features from System1, (ii) output from System1 (translations in PL) and (iii) output from System2 (translations in TL). In other words, we re-rank the n -best list of translations in PL produced by System1 based on features of this system (and the translations in PL) but also using features from the output of System2 (the translations in TL).

The method uses two features in order to perform re-ranking:

- $-ts$, the translation score assigned by System1 to translations from SL into PL. This is an internal confidence measure common in SMT decoders. It is a log probability in the range $[-\infty, 0]$. We take its negative (range $[0, \infty]$); the lower the value the better the translation is considered to be.
- $\log_2(\text{perp})$, the fluency of the translation produced by System2 in the TL. This is the logarithm of the perplexity given by a language model, in the range $[0, \infty]$. The lower the value, the better the fluency is considered to be.

The translations of the n -best list from PL to TL are scored using these two features according to

equation 1. The best translation (the one with the lowest score) is kept.

$$\text{score} = (-\text{ts}) \cdot \alpha + \log_2(\text{perp}) \cdot (1 - \alpha) \quad (1)$$

The parameter α , which can take any value in the interval $[0, 1]$, assigns complementary weights to the two features. An iterative process is followed in order to find the optimal value of α in the development set. The pseudocode of the algorithm is shown in Algorithm 1.

Algorithm 1 Find optimal α

```

scorebest ← 0
αbest ← 0.5
α ← 0.5
depth ← 1
max_depth ← 16
while depth < max_depth do
  α1 ← α +  $\frac{0.5}{2^{\text{depth}}}$ 
  α2 ← α -  $\frac{0.5}{2^{\text{depth}}}$ 
  score1 ← MT score at α1
  score2 ← MT score at α2
  if score1 = score2 then
    break
  end if
  α ← α of max(score1, score2)
  if scorebest < max(score1, score2) then
    scorebest ← max(score1, score2)
    αbest ← α
  end if
  depth = depth + 1
end while
return αbest

```

The procedure starts with $\alpha = 0.5$ (the average value in the range $[0, 1]$). At each step it calculates the scores of the translations selected when using $\alpha_1 = \alpha - \frac{0.5}{2^{\text{depth}}}$ and $\alpha_2 = \alpha + \frac{0.5}{2^{\text{depth}}}$, sets as new α the one for which the MT score is higher between the two, increments the value of *depth* and starts again. The procedure stops when the maximum value of *depth* is reached or when both MT scores at α_1 and α_2 are equal. The best value of α selected during the procedure is then used to select the translations for the test set.

4 Evaluation

4.1 Experimental Setting

The experiments have been carried out for two scenarios (involving different languages and do-

mains). The first scenario translates from Italian (SL) to Catalan (TL), passing through Spanish (PL). The test set consists of technical documentation data. We refer to this scenario as it-es-ca. The second scenario involves English as the SL, Bulgarian as the PL and Macedonian as the TL. The test set consists of newswire data. This scenario is referred to as en-bg-mk.

For System1 we use the phrase-based SMT Moses (Koehn et al., 2007)¹ in both scenarios. This system is trained and tuned on Europarl (Koehn, 2005)² Italian-Spanish for the first scenario and Europarl English-Bulgarian for the second. The corpora are tokenised and lower-cased, and sentences where the source or the target is longer than 40 words are discarded. From the sentences extracted, we set aside 1,000 as development set for parameter tuning using MERT (Och, 2003) and we use the rest for training, i.e. 1,278,411 sentences for Italian-Spanish and 196,113 for English-Bulgarian.

For each SL sentence we obtain the *n*-best (up to 3,000) PL translations. We ensure that all translations in the *n*-best list are different (using the Moses parameter *distinct*). In order to obtain different translations, Moses considers the best *n* · *m* translations (*m* = 200), therefore it is not guaranteed that *n* different translations will be found (in fact, for some sentences we obtain a number of translations slightly lower than *n*). Apart from this, we use Moses' default settings. The translations in PL are recased using Moses' built-in recaser trained on the target side of the SL-PL training data.

For System2 in both scenarios we use Apertium, a RBMT system that uses a shallow-transfer engine (Forcada et al., 2011).³ We use Apertium systems developed for Spanish-Catalan (Corbí-bellot et al., 2005) and Bulgarian-Macedonian (Rangelov, 2011).

The development and test sets for it-es-ca are extracted from the KDE4 multilingual documentation corpus in the OPUS project (Tiedemann, 2009).⁴ The Italian-Catalan bilingual corpus contains 146,372 sentence pairs. We discarded sentence pairs where the source or target side is

¹<http://www.statmt.org/moses/>

²<http://www.statmt.org/europarl/>

³<http://www.apertium.org/>

⁴<http://urd.let.rug.nl/tiedeman/OPUS/KDE4v2.php>

shorter than 10 words⁵ or longer than 30,⁶ where the difference of number of words between the source and target sentences is higher than 10% as well as sentences that contain URLs, Copyright notices and source code. This leads to a candidate set of 6,927 sentences, from it we randomly selected 1,000 sentences for development and 1,000 for test. The development set is used for the tuning procedure shown in Algorithm 1.

The development and test sets for en–bg–mk are taken from the SETimes multilingual corpus (Tyers and Alperen, 2010).⁷ The development set contains 1,000 sentences whilst the test set holds 1,003 sentences.

5-gram word-based Language Models (LMs) are built for the TL with the IRSTLM toolkit (Federico et al., 2008)⁸ using modified Kneser-Ney smoothing (Chen and Goodman, 1996). We use two monolingual corpora for the TL in the first scenario: one in-domain, from the KDE4 corpus, which consists of 53,776 sentences from the Catalan side which are not present in the aforementioned development nor test sets and one out-of-domain, consisting on up to 800,000 sentences gathered from news monolingual sources. A single monolingual corpus is used for the second scenario, in this case in-domain as it consists of sentences from the SETimes corpus. Up to 150,000 sentences are used.

Two automatic MT metrics are used to evaluate our approach, these are BLEU (Papineni et al., 2002) and NIST (Doddington, 2002). Statistical significance tests are carried out using paired bootstrap resampling (Koehn, 2004) with ARK’s code.⁹ Sentence-level scores for the oracles are computed with smoothed BLEU.¹⁰ BLEU is also used as the MT score to tune the procedure shown in Algorithm 1.

4.2 Experiments

4.2.1 Baseline and Oracles

First we establish the baseline, which consists of combining the two MT systems sequentially in a cascade fashion, i.e. for each source sentence this

⁵Those sentences are usually not fluent sentences but menu items, isolated terms, etc.

⁶Such long sentences are not ideal for potential tasks such as word alignment.

⁷<http://opus.lingfil.uu.se/SETIMES.php>

⁸<http://hlt.fbk.eu/en/irstlm>

⁹<http://www.ark.cs.cmu.edu/MT/>

¹⁰<ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13a.pl>

is the translation by System2 of the best translation obtained by System1.

In order to determine the margin for improvement that can be attained when taking into consideration all the translations in the n -best list, we have developed an oracle system which yields the maximum reachable score. The oracle is based on the one described in (Och et al., 2004) where only one reference translation is available; for each sentence, it translates all the translations in the n -best list in PL with System2 into TL, scores BLEU at sentence level, and picks the translation with the highest score. Finally it builds a set in the TL with the translation picked for each sentence and scores BLEU at document level.

Table 1 shows the BLEU and NIST scores for the baselines and the oracles for both scenarios and for different sizes of the PL n -best list (100, 1,000, 2,000 and 3,000). Apart from the absolute scores, for each metric and oracle we report its relative improvement over the baseline (in columns labelled $\Delta\%$).

For the first scenario, the oracle is almost 6 absolute points higher than the baseline (0.2878 vs 0.2289) according to BLEU with just 100 sentences in the n -best list. Incrementing the list to 1,000 sentences yields approximately 2.5 additional BLEU points (0.3133 vs 0.2878). This can be incremented by almost 3.5 further points by considering the top 2,000 translations (0.3476 vs 0.3133); this is 11.87 absolute points over the baseline (0.3476 vs 0.2289) or a 51.86% relative improvement. In comparison, having access to the best 3,000 translations brings about only modest further improvements: about half a BLEU point over the oracle that uses 2,000 translations (0.3525 vs 0.3476).

Sustained improvements are reported also by NIST, although they are lower (the maximum relative improvement is 24.29%).

A similar pattern is observed for the second scenario. In this case the relative improvements are even higher; 44.66% for 100-best, 68.02% for 1,000-best and almost 75% both for 2,000-best and 3,000-best in terms of BLEU.

The results clearly indicate that methods that exploit the n -best list to translate from PL to TL have potential to improve performance considerably over the baseline. Given the similar results obtained when using 2,000-best and 3,000-best lists and taking into consideration the computa-

Scenario	MT system	n -best size	BLEU	$\Delta\%$	NIST	$\Delta\%$
it-es-ca	Baseline	-	0.2289	0.00	5.6706	0.00
	Oracle	100	0.2878	25.73	6.2778	10.70
	Oracle	1,000	0.3133	36.87	6.5884	16.19
	Oracle	2,000	0.3476	51.86	6.9909	23.28
	Oracle	3,000	0.3525	54.00	7.0482	24.29
en-bg-mk	Baseline	-	0.1104	0.00	4.3274	0.00
	Oracle	100	0.1597	44.66	5.1126	18.14
	Oracle	1,000	0.1855	68.02	5.4734	26.48
	Oracle	2,000	0.1931	74.90	5.5646	28.59
	Oracle	3,000	0.1927	74.55	5.4222	25.30

Table 1: MT scores for the baselines and oracles

tional cost involved, we consider lists of 2,000-best sentences for the rest of the experiments.

4.2.2 Pivot Systems

We now turn to our pivot systems that rank translation output according to SL-PL translation score and TL perplexity (rather than oracle selection). We evaluate the pivot method using different LMs. For the first scenario there are four systems that use out-of-domain LMs (newswire) made up of a different number of sentences: News-100k (100,000), News-200k (200,000), News-500k (500,000) and News-800k (800,000). Finally there is a system that uses an in-domain LM, KDE-50k, derived from 50,000 sentences of the KDE corpus.

Regarding the second scenario, we have built three in-domain LMs, using 50,000 (SET-50k), 100,000 (SET-100k) and 150,000 (SET-150k) sentences from the SETimes corpus. The results obtained according to the BLEU and NIST metrics and the improvements over the baseline using 2,000-best lists are shown in Table 2.

The results obtained by the pivot systems using out-of-domain LMs are slightly higher than the baseline (except for the BLEU score for the system News-100K, which is slightly lower). However, only the NIST score for the system News-800K is significantly better than the baseline ($p = 0.05$).

Although using a much smaller LM, the system that uses an in-domain LM made up of 50,000 sentences from the KDE corpus reaches notably higher scores, achieving almost 3 absolute BLEU points over the baseline (0.2561 vs 0.2289, or 11.88% relative improvement). Both the BLEU and NIST scores are statistically significantly better than the baseline ($p = 0.01$). As the testset

comes from a very specific and technical domain, having a LM from that same domain (even if it is rather small) to re-rank the translations proves to be very useful.

All the pivot systems for the second scenario obtain significantly better scores compared to the baseline ($p = 0.01$). The highest improvement is achieved by SET-150k (13.32% relative and 1.47 absolute in terms of BLEU).

For all the runs using out-of-domain LMs, the value of α is very high (the values range from 0.9453 to 0.9824), meaning that almost all the weight to choose the best translation is given to the feature that measures translation score in PL, while the one that measures fluency in the TL remains marginal ($1 - \alpha$, see equation 1). As the original n -best list is sorted by translation score, we can expect that in these runs most sentences selected are very near the top of this list; hence the results do not differ much from the baseline. Conversely, the value of α is considerably lower for the runs using in-domain LMs; 0.8125 for KDE-50k in the first scenario, even lower values for the second scenario, in the range [0.5390, 0.6250]. This suggests that the fluency in TL plays a more important role in the selection of translations from the n -best list when using an in-domain LM. More details on this are provided in Section 4.3, where the results are analysed.

4.3 Analysis

We provide an analysis of all the systems evaluated by looking at the distribution of the ranking positions of the sentences selected in the n -best lists. For each of the systems we report on the following statistical measures:

- Minimum (*min*), the rank of the highest sen-

Scenario	MT system	α	BLEU	$\Delta\%$	NIST	$\Delta\%$
it-es-ca	News-100k	0.9453	0.2283	-0.26	5.6739	0.05
	News-200k	0.9551	0.2301	0.52	5.6909	0.35
	News-500k	0.9824	0.2299	0.43	5.6844	0.24
	News-800k	0.9824	0.2300	0.48	5.6853	0.25
	KDE-50k	0.8125	0.2561	11.88	6.0130	6.03
en-bg-mk	SET-50k	0.6250	0.1238	12.14	4.4060	1.81
	SET-100k	0.5390	0.1245	12.77	4.4085	1.87
	SET-150k	0.5469	0.1251	13.32	4.4115	1.94

Table 2: MT scores for the pivot method

tence picked by the method.

- Maximum (*max*), the rank of the lowest sentence picked by the method.
- Mean, the average value of the ranking positions of the sentences chosen by the method.
- Standard deviation (*stddev*), the standard deviation from the average of the sentences selected.

Table 3 provides these values for the oracle systems over the different sizes of the *n*-best list (100, 1,000, 2,000 and 3,000). The high values of the *max* and *stddev* show that the oracles select sentences from the whole range of translations available in the *n*-best lists. At least one of the lowest ranked translations was taken for 100-best (*max* 99), while one very near the end of the list was selected from 1,000-best (*max* 999 and 998), 2,000-best (*max* 1,990 and 1,995) and 3,000-best (*max* 2,997 and 2,995).

	<i>n</i> -best size	min	max	mean	stddev
it-es-ca	100	0	99	17.53	27.34
	1,000	0	999	192.31	279.54
	2,000	0	1,990	472.34	579.35
	3,000	0	2,997	716.36	880.05
en-bg-mk	100	0	99	39.66	29.95
	1,000	0	998	400.10	297.25
	2,000	0	1,995	818.78	611.48
	3,000	0	2,995	1,002.88	862.01

Table 3: Statistics for oracles

Table 4 shows the statistics for the pivot systems. The previous hypothesis that systems using out-of-domain LMs select most sentences very near the top due to the very high value of α is

corroborated here by the statistical measures. Although the systems have access to 2,000 translations, the lowest ranked sentence picked by one of the systems using an out-of-domain LM (News-100k) is at position 133, while two of them (News-500k and News-800k) do not even select any translation beyond a rank as high as 9. The very low values of the mean, which range from 0.18 to 1.51, indicate that most translations are taken from the very highest ranked sentences.

The statistics are very different for the systems that use in-domain LMs. The values in this case resemble much more the pattern observed for the statistics shown for the oracles (Table 3). These systems do extract translations from all the range of ranks as indicated by the values of the lowest translation selected (1,990 for KDE-50k, 1,998 for systems using LMs built on SETimes), which are figures similar to those reported for the 2,000-best oracles (1,990 for es-it-ca and 1,995 for en-bg-mk). The high values of both the mean (214.65 for the first scenario and [615.25, 801.17] for the second) and the standard deviation (420.99 for the first scenario and [585.17, 593.20] for the second) confirm this trend.

	LM	min	max	mean	stddev
it-es-ca	News-100k	0	133	1.51	6.84
	News-200k	0	88	0.97	4.00
	News-500k	0	9	0.19	0.69
	News-800k	0	9	0.18	0.66
	KDE-50k	0	1,990	214.65	420.99
en-bg-mk	SET-50k	0	1,998	615.25	585.17
	SET-100k	0	1,998	801.17	593.20
	SET-150k	0	1,998	784.55	588.19

Table 4: Statistics for pivot systems

5 Conclusions and Future Work

This paper has presented, to the best of our knowledge, the first pivot-based MT methodology in which the second MT system is treated as a black box.

Compared to the state-of-the-art, our methodology is applicable to a broader set of scenarios, as no access to the internals of the second system is required. This opens the applicability of MT pivot-based approaches to target languages for which no suitable bilingual corpora to build PL–TL SMT systems are available, as long as there is any kind of PL–TL MT system available.

We have presented a method which exploits two types of features: internal of the system that translates from SL to PL, and from the output of the final TL translation. An algorithm that uses two features (translation score of the first system and perplexity of the final translation) is presented. Complementary weights are given to the features and the optimal values are tuned on the development set. The source code that implements this procedure is available under the GPL-v3 license.¹¹ The data used in the experiments is also available.

We have evaluated this approach comparing it to a baseline, which consists of translating the input sentences using the two MT systems sequentially. We have experimented with two scenarios that involve different language families and domains, technical documentation in Romance languages and newswire in Slavic languages, obtaining up to 11.88% and 13.32% relative improvements in terms of BLEU, respectively.

Using just two features yields significant improvements for both scenarios, but the scores obtained by the oracles indicate that there is still considerable room for improvement, e.g. for the 2,000-best configuration, the best pivot-based systems (KDE-50k and SET-150k) obtain 0.2561 and 0.1251 BLEU points, while the oracles yield 0.3476 and 0.1931 (over 9 absolute points better in the first case and nearly 7 in the second).

Therefore we plan to extend the methodology presented here in several ways. First, we will explore other possible features, looking for example at features successfully used in other MT-related tasks, such as (He et al., 2010). Second, we will experiment with other algorithms that allow us to combine an arbitrary number of features in order to

rescore the translations. Finally, the methodology will be applied to different MT systems, language pairs and domains in order to further validate the applicability of this approach.

Acknowledgements

We would like to thank Francis Tyers and Tihomir Rangelov for their help and ideas regarding the experiment that involves Bulgarian and Macedonian. This work has been funded by the European Association for Machine Translation through its 2011 sponsorship of activities program.

References

- Bertoldi, Nicola, Madalina Barbaiani, Marcello Federico, and Roldano Cattoni. 2008. Phrase-Based Statistical Machine Translation with Pivot Languages. In *Proc. of the International Workshop on Spoken Language Translation*, pages 143–149, Hawaii, USA.
- Chen, Stanley F. and Joshua Goodman. 1996. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, ACL '96, pages 310–318, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Cohn, Trevor and Mirella Lapata. 2007. Machine Translation by Triangulation: Making Effective Use of Multi-Parallel Corpora. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 728–735, Prague, Czech Republic.
- Corbí-bellot, Antonio M., Mikel L. Forcada, Sergio Ortiz-rojas, Juan Antonio Pérez, Gema Ramírez-sánchez, Felipe Sánchez-martínez, Iñaki Alegria, and Kepa Sarasola. 2005. An Open-Source Shallow-Transfer Machine Translation Engine for the Romance Languages of Spain. In *In Proceedings of the 10th European Association for Machine Translation Conference*, pages 79–86.
- Doddington, George. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*, pages 138–145, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- Federico, Marcello, Nicola Bertoldi, and Mauro Cettolo. 2008. IRSTLM: an open source toolkit for handling large scale language models. In *INTER-SPEECH*, pages 1618–1621. ISCA.
- Forcada, Mikel L., Mireia Ginestí-Rosell, Jacob Nordfalk, Jim O'Regan, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Gema Ramírez-Sánchez Felipe Sánchez-Martínez, and Francis M. Tyers. 2011.

¹¹<http://nclt.computing.dcu.ie/~atoral/#Resources>

- Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144. Special Issue: Free/Open-Source Machine Translation.
- Gispert, Adrià De and José B. Mariño. 2006. Statistical machine translation without parallel corpus: bridging through Spanish. In *Proceedings of LREC 5th Workshop on Strategies for developing Machine Translation for Minority Languages*, pages 65–68.
- He, Yifan, Yanjun Ma, Josef van Genabith, and Andy Way. 2010. Bridging SMT and TM with translation recommendation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 622–630, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Khalilov, M., Marta R. Costa-jussà, José A. R. Fonolosa, Rafael E. Banchs, B. Chen, M. Zhang, A. Aw, H. Li, José B. Mariño, Adolfo Hernández, and Carlos A. Henríquez Q. 2008. The TALP and I2R SMT Systems for IWSLT 2008. In *International Workshop on Spoken Language Translation (IWSLT 2008)*, pages 116–123, Hawaii, USA.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Prague, Czech Republic.
- Koehn, Philipp. 2004. Statistical significance tests for machine translation evaluation. In Lin, Dekang and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 388–395, Barcelona, Spain, July. Association for Computational Linguistics.
- Koehn, Philipp. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand. AAMT, AAMT.
- Leusch, Gregor, Aurélien Max, Josep M. Crego, and Hermann Ney. 2010. Multi-Pivot Translation by System Combination. In Federico, Marcello, Ian Lane, Michael Paul, and François Yvon, editors, *Proceedings of the seventh International Workshop on Spoken Language Translation (IWSLT)*, pages 299–306.
- Och, Franz Josef, Daniel Gildea, Sanjeev Khudanpur, Kenji Yamada, Alex Fraser, Shankar Kumar, David Smith, Katherine Eng, Viren Jain, Zhen Jin, and Dragomir Radev. 2004. Final report of Johns Hopkins 2003 summer workshop on syntax for statistical machine translation. Technical report.
- Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 160–167, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Philadelphia, Pennsylvania, USA.
- Rangelov, Tihomir. 2011. Rule-based machine translation between Bulgarian and Macedonian. In *Proceedings of the Second International Workshop on Free/Open-Source Rule-Based Machine Translation*, pages 53–60, Barcelona, Spain.
- Tiedemann, Jörg. 2009. News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In Nicolov, N., K. Bontcheva, G. Angelova, and R. Mitkov, editors, *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia, Borovets, Bulgaria.
- Tyers, Francis M. and Murat Serdar Alperen. 2010. South-East European Times: A parallel corpus of the Balkan languages. In *Proceedings of the Workshop on Exploitation of multilingual resources and tools for Central and (South) Eastern European Languages*, LREC 2010.
- Utiyama, Masao and Hitoshi Isahara. 2007. A Comparison of Pivot Methods for Phrase-Based Statistical Machine Translation. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 484–491, Rochester, New York, April. Association for Computational Linguistics.
- Utiyama, Masao, Andrew Finch, Hideo Okuma, Michael Paul, Hailong Cao, Hirofumi Yamamoto, Keiji Yasuda, and Eiichiro Sumita. 2008. The NICT/ATR Speech Translation System for IWSLT 2008. In *Proc. of the International Workshop on Spoken Language Translation*, pages 77–84, Hawaii, USA.
- Wu, Hua and Haifeng Wang. 2007. Pivot Language Approach for Phrase-Based Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 856–863, Prague, Czech Republic, June. Association for Computational Linguistics.
- Wu, Hua and Haifeng Wang. 2009. Revisiting Pivot Language Approach for Machine Translation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 154–162, Suntec, Singapore, August. Association for Computational Linguistics.

