A Real-World System for Simultaneous Translation of German Lectures

Eunah Cho[1], *Christian Fügen*[2], *Teresa Hermann*[1], *Kevin Kilgour*[1], *Mohammed Mediani*[1], *Christian Mohr*[1], *Jan Niehues*[1], *Kay Rottmann*[2], *Christian Saam*[1], *Sebastian Stüker*[1], *Alex Waibel*[1]

[1]International Center for Advanced Communication Technologies, Institute for Anthropomatics
Karlsruhe Institute of Technology, Germany
firstname.lastname@kit.edu
[2]Mobile Technologies GmbH, Germany
firstname.lastname@jibbigo.com

Abstract

We present a real-time automatic speech translation system for university lectures that can interpret several lectures in parallel. University lectures are characterized by a multitude of diverse topics and a large amount of technical terms. This poses specific challenges, e.g., a very specific vocabulary and language model are needed. In addition, in order to be able to translate simultaneously, i.e., to interpret the lectures, the components of the systems need special modifications.

The output of the system is delivered in the form or real-time subtitles via a web site that can be accessed by the students attending the lecture through mobile phones, tablet computers or laptops.

We evaluated the system on our German to English lecture translation task at the Karlsruhe Institute of Technology. The system is now being installed in several lecture halls at KIT and is able to provide the translation to the students in several parallel sessions.

Index Terms: speech translation, cloud computing

1. Introduction

Lectures at Karlsruhe Institute of Technology (KIT) are mainly taught in German. Therefore, foreign students that want to study at KIT need to learn German, and not only at a conversational level, but must be proficient enough to follow highly scientific and technical lectures carrying complex content. While foreign students often take a one year preparatory course that teaches them German, experience shows that even after that course, their German is not proficient enough to be able to follow German lectures and thus perform well.

Since the use of human interpreters for bridging the language barrier in lectures is too expensive, we want to solve this issue with the help of our automatic simultaneous lecture translation system. In this system we employ the technology of *spoken language translation* (SLT), which combines *automatic speech recognition* (ASR) and *machine translation* (MT) to build a system that simultaneously translates lectures from German to English.

Our system works with the help of a cloud based service infrastructure. The speech of the lecturer is recorded via a local client and sent to the service infrastructure. A service then manages the flow of the data through the ASR, MT, and other components. The final result is then made available as a website which continuously displays the result of the recognition and translation.

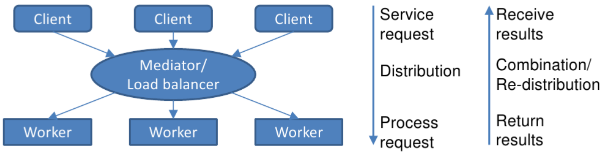


Figure 1: Schematic overview of the service architecture

2. Infrastructure

In order to make the system robust and performant enough to service several lectures held at KIT in parallel, using one central computation facility that is accessed through the university's network, we have improved the infrastructure developed in [1]. The service architecture allows server based recognition and translation of audio and text through a light-weight API. A schematic overview of the service architecture is given in Figure 1, a detailed description can be found in a companion paper [2]. The service architecture enables a connection-based communication with multiple service requests at the same time. A client connects to the mediator and the mediator connects the output media stream of the client with one or multiple workers in order to accomplish a specific service request. Clients are modules that allow users to access and use the service architecture, e.g., a recording application for the lecturer. Workers represent different core components such as speech recognition or machine translation.

The clients typically initiate the service request by specifying the type and language of the media stream that should be processed and by specifying the type and language to which the media stream should be converted. In case of a worker, each worker has to register at the service architecture with one or multiple services that the worker is able to handle. But each worker accepts only one incoming service request per connection.

The mediator is responsible for distributing the requests and load amongst several connected workers, and is also responsible for connecting several workers in order to fulfill more complicated requests such as a simultaneous translation of an audio input stream. For this type of a request, in our case, workers for speech recognition, text-processing, and translation have to be connected.

3. Training and Development Data

In order to develop the speech recognition and machine translation components of our lecture translation system, we needed in-domain data that allows for the adaptation of our models, as well as for the evaluation of the components and the whole system's performance. We therefore collected a corpus of KIT lectures. A detailed description of this corpus and the way we collected it can be found in [3]. University lectures are a challenging domain, due to the many different topics lectures can be held on. The training and test data needs to reflect this heterogeneity.

For training and testing the ASR component of the SLT system, large amounts of in-domain audio data are needed that are transcribed at sentence level. For the MT component of the system, data is needed that consists of parallel sentences in the required domain in all languages between which the system is supposed to translate.

Since lectures provide such a diverse set of topics, the traditional approach of training systems on a fixed set of data and then deploying them, will not be sufficient. Reasonable performance can only be reached by systems that are able to flexibly and autonomously adapt themselves to varying topics of lectures. In order to facilitate this adaptation process, the presence of verbose meta-data, such as the name of the lecturer, his field of expertise, the title of the lecture, or the slides used by him, is very valuable. The corpus collected by us reflects those needs and is thus also intended as a tool for conducting research to advance the state-of-the-art in autonomous and unsupervised adaptation for SLT systems.

While in the beginning we only collected lectures from the computer science department, we later expanded our collection to lectures from all faculties at KIT. Whenever possible we tried to collect not only single lectures from a class, but rather as many lectures from a single class as possible. However, often lecturers only agreed to have one or few lectures recorded, as they thought the recording process to be too interruptive.

The collected lectures were then carefully transcribed and translated into English with the help of trained part time students. From the collected data we created a development set of six test speakers and their lectures.

4. ASR System

The ASR components that we used for the lecture translation system were realized with the help of the *Janus Recognition Toolkit* (JRTk) which features the IBIS single pass decoder [4]. For that we extended the JRTk to be able to act as a worker in the infrastructure described in Section 2.

4.1. Front-End

The front-end of our ASR systems is based on the warped minimum variance distortionless response (MVDR) [5]. The preprocessing provided features every 10 ms, we used an MVDR model order of 22. Vocal tract length normalization (VTLN) [6] was applied in the warped frequency domain. The mean and variance of the cepstral coefficients were normalized on a per-utterance basis. The resultung 20 cepstral coefficients were combined with the seven adjacent frames to a single 300 dimensional feature vector that was reduced to 40 dimensions using linear discriminant analysis (LDA).

4.2. Accoustic Model

We used a context dependent quinphone setup with three states per phoneme, and a left-to-right topology without skip states. We trained a speaker independent model for speakers for whom we didn't have much or no data, as well as speaker dependent models for the 5 speakers for whom we had sufficient data. All models use 4,000 distributions and codebooks and were trained using *incremental splitting of Gaussians* training, followed by *semi-tied covariance* training and 2 iterations of Viterbi training.

For the speaker dependent models the data used in the Viterbi training was restricted to the particular speaker's data. We performed discriminative training using *boosted MMIE* to improve the performance of the speaker independent system.

4.3. Language Model and Test Dictionary

For training the language model of our system we collected training texts from various sources like web dumps, newspapers and transcripts. The resulting 28 text corpora range in size from about 5 MB to just over 6 GB. Our tuning set was randomly selected from the acoustic model training data transcripts. The baseline 300k vocabulary was selected by building a Witten-Bell smoothed unigram language model using the union of all the text sources' vocabulary as the language model's vocabulary (global vocabulary). With the help of the maximum likelihood count estimation method described in [7] we found the best mixture weights for representing the tuning set's vocabulary as a weighted mixture of the sources' word counts, thereby giving us a ranking of all the words in the global vocabulary by their relevance to the tuning set.

4.3.1. Sub-Word Vocabulary

German, our input language to the translation system, is well known for the frequent use of compounds, which makes it difficult to define a static vocabulary containing all words which will be used. We addressed this problem by using a sub-word vocabulary. In order to select it we first performed compound splitting on all the text corpora and tagged the split compounds. Initial experiments showed that only tagging the head of a compound performs best. Linking morphemes are attached to the preceding word. *Wirtschaftsdelegationsmitglieder* is, for example, split into *Wirtschafts+ Delegations+ Mitglieder (eng: members of the economic delegation).*

Our compound splitting algorithm requires a set of valid sub-words and selects the best split from all possible splits by maximizing the sum of the squares of all sub-word lengths [8].

As a set of valid sub-words we selected the top n words from the ranked baseline word-list. The same maximum likelihood vocabulary selection method used to generate the baseline vocabulary was used to select the best vocabulary from this split corpora resulting a ranked vocabulary containing both full words and sub-words.

4.3.2. Query-Based Vocabulary Selection

Due to its technical nature the lecture test set has a very high OOV rate. [9] attempts to solve this problem by generating a vocabulary from the results of queries derived from lecture slides. The data downloaded to build the query vocabulary can also be used to adapt the language model. We applied this method to 4 lecturers for which German lecture slides were available, extracting over 4000 queries per lecture.

Both this method and the proposed sub-word vocabulary reduce the OOV rate significantly, from 2.25% to 0.75% for a 300k vocabulary.

Lecturer	Lecturer 1	Lecturer 2	Lecturer 3	Lecturer 4	Lecturer 5	Lecturer 6
Speaker Independent AM WER	34.79%	21.08 %	28.44%	22.85%	22.73%	18.97%
Speaker Dependent AM WER	—	18.87%	27.63%	22.63%	21.52%	17.84%
Adapted LM +Vocab	23.87%	17.31%	—	18.07%	—	15.39%

Table 1: WER for our six test speakers for the speaker-independent AMs, speaker adapted AMs and LMs adapted on the slides

4.4. ASR Performance

We evaluated our systems on our development set of six lecturers. Table 1 shows the results on these speakers. You can see that the speaker dependent models improve performance for the five speakers for which sufficient amounts of training data were available. Similarly, the performance improved with the LM adapted from the slides for those four speakers, for which we did have German slides.

4.5. Punctuation Prediction and ASR Post-Processing

The output of our ASR system is a continuous stream of words without segment boundaries and punctuation. It is thus hard to read and can cause problems for the machine translation, which translates whole sentences. Our punctuation prediction setup can detect both full stops and commas. Long pauses force a full stop and short pauses increase the probability of a punctuation mark computed by a 4-gram language model.

After punctiuation prediction all numbers are normalized so that they appear as digits. Common symbols like (%, , ...) are used instead of text and simple equations like $P(x_i) = x_1 - x_2$ are converted into their proper math from.

5. Machine Translation

For the lecture translation system we use a phrase-based statistical machine translation system. The system was trained on the EPPS corpus, News Commentary corpus, BTEC corpus, TED corpus and the data collected internally at the KIT. We performed specific pre-processing to better match the characteristics of speech translation. Furthermore, the system has been adapted to the task using the internally collected lecture data. We also used additional resources like Wikipedia to be able to translate domain specific terms. Finally, we modified the system to enable it to perform the simultaneous translation in the lecture translation system.

5.1. Pre-Processing

Before training the training texts were pre-processed. Besides the usual normalization we performed smart casing, as well as compound splitting for the German side and treated numbers.

The two-step compound splitting described for the German ASR is applied to the source side of the training data, in order to be consistent with the ASR output, that will be the input to the MT system.

Also, for the sake of consistency between the ASR and the MT system, we apply a rule-based handling for numbers.

In order to avoid that person names are compound-split and translated into multiple words, we use a named entity tagger. By using a list of titles and a list of names, we tag sequences of names and titles. Names tagged this way will not be translated and the order between the title and the name is kept fixed.

5.2. Training

We applied the Discriminative Word Alignment approach described in [10]. This alignment model is trained on a small corpus of hand-aligned data and uses the lexical probability as well as the fertilities generated by the PGIZA++ Toolkit (http://www.cs.cmu.edu/˜qing/) and POS information.

To model reordering we first learn probabilistic rules from the POS tags of the words in the training corpus and the alignment information. Continuous reordering rules are extracted as described in [11] to model short-range reorderings. We apply a modified reordering model with non-continuous rules to cover also long-range reorderings [12]. The reordering rules are applied to the source text and the original order of words and the reordered sentence variants generated by the rules are encoded in a word lattice which is used as input to the decoder. For the test sentences, the POS-based reordering allows us to change the word order in the source sentence so that the sentence can be translated more easily. By applying this also to the training sentences, we were able to extract the phrase pairs for originally discontinuous phrases and can apply them during translation of reordered test sentences. Therefore, we built reordering lattices for all training sentences and then extracted phrase pairs from the monotone source path as well as from the reordered paths.

The 4-gram language model is trained on the target side of the parallel data. To have source side context in addition to the target side context information, we used a bilingual language model as described in [13].

Scores for the test sets from six speakers mentioned earlier are shown in Table 2. The scores are reported in case-insensitive BLEU.

Lecturer	BLEU
Lecturer 1	13.80
Lecturer 2	22.58
Lecturer 3	14.24
Lecturer 4	20.83
Lecturer 5	24.50
Lecturer 6	24.13

Table 2: Offline test scores for six speakers

5.3. Adaptation

We adapted the language model as well as the translation model to the lecture domain to improve the performance on this task. For the translation model adaptation, first, a large model was trained on all the available data. Then, a separate in-domain model was trained on the in-domain data only reusing the same alignment from the large model. The two models are then combined using a log-linear combination to achieve the adaptation towards the target domain. The newly created translation model uses the four scores from the general model as well as the two smoothed relative frequencies of both directions from the small in-domain model. If the phrase pair does not occur in the in-

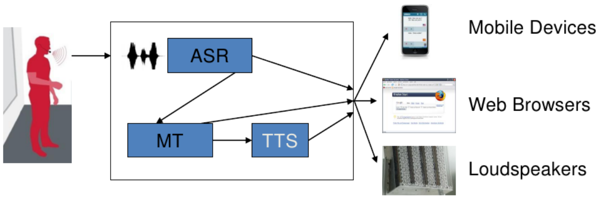


Figure 2: Schematic overview of the simultaneous lecture translation system.

domain part, a default score is used instead of a relative frequency. In our case, we used the lowest probability.

We also adapted our system by log-linear combination of the big language model trained on all data with one trained on the lecture data. In addition, we use a third language model trained on the TED corpus, since this is more similar to the target domain than the other out-of-domain data.

5.4. Special Terms

One problem when building the machine translation system is to acquire translations for domain specific terms. For example, if we want to translate computer science lectures, we need also to learn translations for terms such as *sampling* or *quantisation*. We tried to get these translations from Wikipedia, which provides articles on very specific topics in many different languages as described in [14]. To extract translations for the domain specific terms, we used the inter-language links of Wikipedia. Using these links we can align the articles in source and target language. Although the articles are no translations of each other and cannot be used directly in the translation system, the titles themselves tend to be translations of each other. We trained a phrase table on this additional corpus and use this phrase table only for the OOV words of the original phrase table.

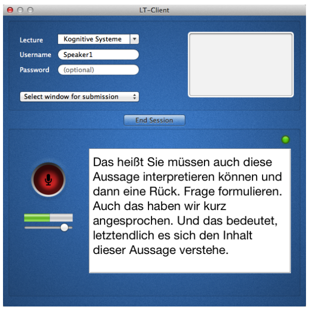
Since only the word lemmas occure in the titles of Wikipedia, we learn quasi-morphologic operations form the parallel data to generate translations for other word forms from the lemmas occuring in the wikipedia titles.

To increase our vocabulary even further, we also use the resource of wiktionary[1] to learn additional translation. Here, the entries for one word in a language is also linked to the translation is a different language. Since we have no statistics about which translation to choose, we also choose the first mentioned translation.

5.5. Online System

Since in the lecture translation system we do not know the test data beforehand, we use the ASR vocabulary to filter and generate the final phrase table of the MT system. We only keep phrase pairs which source phrase includes only words from the ASR vocabulary.

Since the Tree Tagger that we used during training is too time consuming to be used for POS tagging in the live translation system, we used a simplified tagger in the interpretation system. This tagger tags each entering word with the most frequent tag for the word in the training data.



Lecture: Kognitive Systeme
Username: Speaker1
Password: (optional)
Select window for submission

Das heißt Sie müssen auch diese Aussage interpretieren können und dann eine Rück. Frage formulieren. Auch das haben wir kurz angesprochen. Und das bedeutet, letztendlich es sich den Inhalt dieser Aussage verstehe.

Figure 3: Prototype implementation of a client.

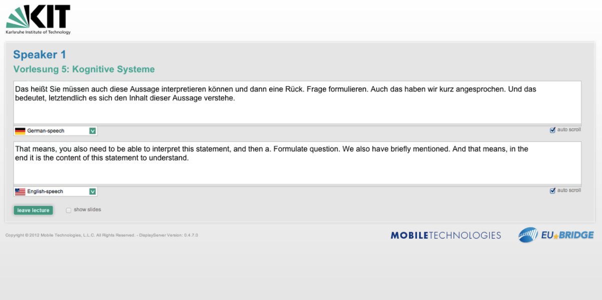
6. Interface

Figure 2 gives a schematic overview of the simultaneous translation system. A client was implemented that connects to a microphone worn by the speaker, captures the slide currently presented, and transmits both information as separate output streams to the mediator for processing. In order for the service architecture to be able to handle the audio and slides correctly, both streams are annotated with additional meta information, such as the type of the stream, the identity of the speaker, and the identity of the lecture being recorded and streamed. The client also provides feedback about the quality of the recording which is influenced, e.g., by the gain level and positioning of the microphone. Figure 3 shows a prototype implementation of a client for Mac OS X.

On the other hand, the result of the translation, but also optionally the result of the speech recognition, is delivered to the users via a web-site. The creation and serving of this web-site is the job of the display server. Using the display server, students can log into a specific lecture that is currently given independent from their current location. The web-site is also comfortably viewable by a wide range of devices, from a classical laptop to smart-phones and tablet computers. Figure 4 shows a screenshot of the display server during use.

7. Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n 287658 — 'Bridges Across the Language Divide' (EU-BRIDGE). 'Research Group 3-01' received financial support by the 'Concept for the Future' of Karlsruhe Institute of Technology within the framework of the German Excellence Initiative.



Speaker 1
Vorlesung 5: Kognitive Systeme

Das heißt Sie müssen auch diese Aussage interpretieren können und dann eine Rück. Frage formulieren. Auch das haben wir kurz angesprochen. Und das bedeutet, letztendlich es sich den Inhalt dieser Aussage verstehe.

German-speech

That means, you also need to be able to interpret this statement, and then a. Formulate question. We also have briefly mentioned. And that means, in the end it is the content of this statement to understand.

English-speech

Copyright © 2012 Mobile Technologies, L.L.C. All Rights Reserved. - DisplayServer Version: 0.4.7.0

Figure 4: Display Server.

[1] http://www.wiktionary.org/

8. References

[1] C. Fügen, “A system for simultaneous translation of lectures and speeches,” Ph.D. dissertation, Universiät Karlsruhe (TH), November 2008.

[2] K. Rottmann, C. Fügen, and A. Waibel, “Network Infrastructure of the KIT Lecture Translation System,” in *submitted to Proceedings of the Interspeech 2013*, Lyon, France, 2013.

[3] S. Stüker, F. Kraft, C. Mohr, T. Herrmann, E. Cho, and A. Waibel, “The kit lecture corpus for speech translation,” in *Proceedings of LREC 2012*, Istanbul, Turkey, May 2012.

[4] H. Soltau, F. Metze, C. Fügen, and A. Waibel, “A one-pass decoder based on polymorphic linguistic context assignment,” in *Automatic Speech Recognition and Understanding, 2001. ASRU ’01. IEEE Workshop on*, 2001, pp. 214–217.

[5] M. Wölfel, J. McDonough, and A. Waibel, “Minimum variance distortionless response on a warped frequency scale,” in *Eurospeech 2003*, 2003.

[6] P. Zhan and A. Waibel, “Vocal tract length normalization for large vocabulary continuous speech recognition,” DTIC Document, Tech. Rep., 1997.

[7] A. Venkataraman and W. Wang, “Techniques for effective vocabulary selection,” *Arxiv preprint cs/0306022*, 2003.

[8] T. Marek, “Analysis of german compounds using weighted finite state transducers,” *Bachelor thesis, University of Tübingen*, 2006.

[9] P. Maergner, K. Kilgour, I. Lane, and A. Waibel, “Unsupervised vocabulary selection for simultaneous lecture translation,” 2011.

[10] J. Niehues and S. Vogel, “Discriminative Word Alignment via Alignment Matrix Modeling.” in *Proc. of Third ACL Workshop on Statistical Machine Translation*, Columbus, USA, 2008.

[11] K. Rottmann and S. Vogel, “Word Reordering in Statistical Machine Translation with a POS-Based Distortion Model,” in *TMI*, Skövde, Sweden, 2007.

[12] J. Niehues and M. Kolss, “A POS-Based Model for Long-Range Reorderings in SMT,” in *Fourth Workshop on Statistical Machine Translation (WMT 2009)*, Athens, Greece, 2009.

[13] J. Niehues, T. Herrmann, S. Vogel, and A. Waibel, “Wider Context by Using Bilingual Language Models in Machine Translation,” in *Sixth Workshop on Statistical Machine Translation (WMT 2011)*, Edinburgh, UK, 2011.

[14] J. Niehues and A. Waibel, “Using wikipedia to translate domain-specific terms in smt,” in *Proceedings of the eight International Workshop on Spoken Language Translation (IWSLT)*, 2011.