Building a Turkish ASR system with minimal resources

Arianna Bisazza and **Roberto Gretter**

Fondazione Bruno Kessler – Trento, Italy
bisazza@fbk.eu, gretter@fbk.eu

Abstract

We present an open-vocabulary Turkish news transcription system built with almost no language-specific resources. Our acoustic models are bootstrapped from those of a well trained source language (Italian), without using any Turkish transcribed data. For language modeling, we apply unsupervised word segmentation induced with a state-of-the-art technique (Creutz and Lagus, 2005) and we introduce a novel method to lexicalize suffixes and to recover their surface form in context without need of a morphological analyzer. Encouraging results obtained on a small test set are presented and discussed.

1. Introduction

Automatic Speech Recognition (ASR) systems are typically trained on manually transcribed speech recordings. Sometimes, however, this kind of corpora are either not available or too expensive for a given language, while it is pretty cheap to acquire untranscribed audio data, for instance from a TV channel. As regards language modeling (LM), only written text in the given language is required in principle. In reality, though, specific linguistic processings can be necessary to obtain reasonable performance in some languages. Turkish, with its agglutinative morphology and ubiquitous phonetic alternations, is generally classified as one of such languages. In this work, we investigate the possibility of building a Turkish ASR system with almost no language-specific resources. While this may seem an unrealistic scenario as more and more NLP tools and corpora are nowadays available for Turkish, we believe that our method may inspire further research on under-resourced languages with similar features, such as other Turkic languages or agglutinative languages in general.[1]

2. Unsupervised Acoustic Modeling

Acoustic modeling (AM) in state-of-the-art ASR systems is based on statistical engines capable to capture the basic sounds of a language, starting from an inventory of pairs ⟨utterance - transcription⟩. When only audio material is available, it can be processed in order to obtain some automatic transcription. Despite the fact that there will be transcription errors, it can be used to build a first set of suboptimal AMs, which can in turn be used to obtain better transcriptions in an iterative way.

2.1. Audio recordings

International news are acquired from a satellite TV channel broadcasting news in different languages, including Turkish. It broadcasts a cyclic schema that lasts about 30 minutes, and roughly consists of: main news of the day (politics, current events); music & commercials; specialized services (stock, technology, history, nature); music & commercials. From an ASR perspective, data are not easy to handle, as several phenomena take place: often, in case of interviews, some seconds of speech in the original language are played before the translation starts; commercials are often in English; there is the presence of music; sometimes a particular piece of news may contain the original audio, in another language. In this paper we use 108 hours of untranscribed recordings (1 hour per day within almost 4 months) of the Turkish channel. Moreover, a small amount of disjoint audio data, about 12 minutes, was manually transcribed in order to obtain a test set (*TurTest*) containing 1494 reference words.

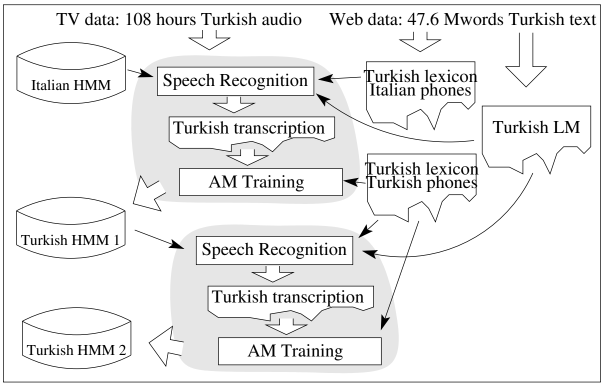

Figure 1: Block diagram of the procedure to bootstrap Turkish AMs from Italian ones.

2.2. Unsupervised acoustic training procedure

Figure 1 shows the unsupervised training procedure used for bootstrapping the phone Hidden Markov Models (HMMs) of a target language (Turkish) starting from those of a "well trained" source language (Italian) – for more details on this procedure see (Falavigna and Gretter, 2011). First we automatically transcribe the Turkish audio training data using a Turkish Language Model (LM), a lexicon expressed in terms of the Italian phones, and Italian HMMs. Then, a first set of Turkish HMMs (HMM 1 in Figure 1) is trained and used to re-transcribe the Turkish audio training data; this second transcription step makes use of a Turkish lexicon. A second set of Turkish HMMs (HMM 2 in Figure 1) is then trained using the new resulting transcriptions. Note that the procedure shown in Figure 1 could be iterated several times.

During the transcription stages, a Turkish LM was needed

[1] This work was partially funded by the European Union under FP7 grant agreement EU-BRIDGE, Project Number 287658.

REF:	ülkede**ki** **işçi sendikaları da** hükümetin **duyars**ız davrandığına **dikkati çekiyor**
HYP:	diğer i**ki** **işçi sendikaları da** internetten **duyar** **s**erdar arda **dikkati çekiyor**
REF:	**ülke çapında yapılan protesto gösterileriyle** **ma**denciler seslerini duyurmaya çalışırken
HYP:	**ülke çapında yapılan protesto gösterileri ile** **ma**vi jeans test edilmesi ve serkan

Table 1: Recognition of two Turkish utterances obtained with Italian acoustic models (first stage).

to drive the speech recognizer. It is coupled with a transcribed lexicon which provides the phonetic transcription of every word, expressed either in Italian phones (for the first iteration) or in Turkish phones (for the other iterations). Turkish phones which do not appear in the Italian inventory were mapped according to the following SAMPA table (http://www.phon.ucl.ac.uk/home/sampa/turkish.htm):

h:	h → ⟨sil⟩	*ı*:	1 → i	*ö*:	2 → o
ü:	y → u	*j*:	Z → dZ		

The collection of text data for training n-gram based LMs was carried out through web crawling. Since May 2009 we have downloaded, every day, text data from various sources, mainly newspapers in different languages including Turkish. A crucial task for LM training from web data is text cleaning and normalization: several processing steps are applied to each html page to extract the relevant information, as reported in (Girardi, 2007).

The LM for this stage was trained on 47.6 million words, which include the period of the audio recordings. Only number processing was applied at this stage. Perplexity (PP) on the small test set results very high (2508) while Out-of-Vocabulary (OOV) rate is reasonable (1.61%).

2.3. Convergence

Recognition on *TurTest* using the Italian AMs resulted in a 26.0% Word Accuracy (WA), corresponding to about 65% Phone Accuracy. Table 1 reports reference and ASR output for two samples, having 18 reference words and 14 ASR errors. Even if this corresponds to only 22.2% WA, phonetically more than half of the utterances are correct (highlighted in bold), resulting in a positive contribution to the AM training. The main causes of error at this stage were: acoustic mismatch, high perplexity and arbitrary phone mapping. However, despite the fact that 74.0% of the words are wrongly recognized, the second stage showed an encouraging 56.4% WA, which became 63.5% and 65.1% in the third and fourth stages.

3. Turkish Language Modeling

It is well known that morphologically rich languages present specific challenges to statistical language modeling. Agglutinative languages, in particular, are characterized by a very fast vocabulary growth. As shown for instance by Kurimo et al. (2006), the number of new words does not appear to level off even when very large amounts of training data are used. As a result, word segmentation appears as an important requirement for a Turkish ASR system. Two main approaches can be considered: rule-based and unsupervised. Rule-based segmentation is obtained from full morphological analysis, which for Turkish is typically produced by a two-level analyzer (Koskenniemi, 1984; Oflazer, 1994; Sak et al., 2008). On the other hand, unsupervised segmentation is generally learnt by algorithms based on the Minimum Description Length principle (Creutz and Lagus, 2005).

Another important feature of Turkish is rich suffix allomorphy caused by few but ubiquitous phonological processes. Vowel harmony is the most pervasive among these, causing the duplication or quadriplication of most suffixes' surface form. In this work we propose a novel, data-driven method to normalize (lexicalize) word endings and to subsequently predict their surface form in context. To our knowledge, this was only done by hand-written rules in past research.

3.1. Unsupervised Word Segmentation

Previous work (Arısoy et al., 2009) demonstrated that, for the purposes of ASR, unsupervised segmentation can be as good as, or even better than rule-based. Following these results, we adopt the unsupervised approach and, more specifically, the popular algorithm proposed by Creutz and Lagus (2005) and implemented in the Morfessor Categories-MAP software. The output of Morfessor for a given corpus is a unique segmentation of each word type into a sequence of morpheme-like units (*morphs*).

Instead of using each morph as a token, we follow a *'word ending'* (or *'half-word'*) approach, which was previously shown to improve recognition accuracy in Turkish (Erdoğan et al., 2005; Arısoy et al., 2009). In fact, while morphological segmentation clearly improves vocabulary coverage, it can result in too many small units that are hard to recognize at the acoustic level. As an intermediate solution between words and morphs, the sequence of non-initial morphs can be concatenated to form so-called *endings*. Note that the morphs do not necessarily correspond to linguistic morphemes and therefore a word ending can include a part of the actual stem.

Some examples are provided in Table 2. The segmentation of the first word (*saatlerinde*) is linguistically correct. On the contrary, in *çocukların*, the actual stem *çocuk* got truncated probably because the letter *k* is often recognized as a verbal suffix. The third word, *düşünüyorum*, is in reality composed of a verbal root (*düşün-*, 'to think') a tense/aspect suffix (*-üyor-*) and a person marker (*-um*). In this case, Morfessor included in the stem a part of the verbal tense suffix and oversplit the rest of the word. Finally, *diliyorum* was not segmented at all, despite being morphologically similar to the previous word. In any case we recall that detecting proper linguistic morphemes is not our goal and it is possible that statistically motivated segmentation be more suitable for the purpose of n-gram modeling.

The Morfessor Categories-MAP algorithm has an important parameter, the perplexity threshold (PPth), that regulates the level of segmentation: lower PPth values mean more aggressive segmentation. As pointed out by the software authors, the choice of this threshold depends on sev-

Word	Morfessor Annotation	Stem+Ending	Stem+Lex.Ending	Meaning
saatlerinde	saat/STM + ler/SUF + in/SUF + de/SUF	saat+ +lerinde	saat+ +lArHnDA	*in the hours of*
çocukların	çocu/STM + k/SUF + lar/SUF + ın/SUF	çocu+ +kların	çocu+ +KlArHn	*of the children*
düşünüyorum	düşünüyo/STM + r/SUF + u/SUF + m/SUF	düşünüyo+ +rum	düşünüyo+ +rHm	*I think*
diliyorum	diliyorum	diliyorum	diliyorum	*I wish*

Table 2: Chain of morphological processing on four training words. Morfessor annotation obtained with PPth=200.

eral factors, among which the size of the corpus. We then decided to experiment with various settings, namely PPth={100, 200, 300, 500}. Results will be given in Section 4. Morfessor was run on the whole training corpus dictionary, from which we only removed singleton entries.

3.2. Data-driven Morphophonemics

Vowel harmony and other phonological processes cause systematic variations in the surface form of Turkish suffixes, i.e. allomorphy[2]. For example, the possessive suffix *-(I)m* 'my' can have four different surface forms depending on the last vowel of the word it attaches to (ex.1-4), plus one if attached to a word that ends with vowel (ex.5):

1) *saç* + *(I)m* -> *saç**ım*** 'my hair'
2) *el* + *(I)m* -> *el**im*** 'my hand'
3) *kol* + *(I)m* -> *kol**um*** 'my arm'
4) *göz* + *(I)m* -> *göz**üm*** 'my eye'
5) *kafa* + *(I)m* -> *kafa**m*** 'my head'

As suffixes belong to close classes, we do not expect these phenomena to be the main cause of vocabulary growth. Nevertheless, we hypothesize that normalizing suffixes – or word endings in our case – may simplify the task of the LM and lead to more robusts models. Since the surface realization of a suffix depends only on its immediate context, we can leave its prediction to a post-processing phase.

In (Erdoğan et al., 2005) vowel harmony is enforced *inside* the LM by means of a weighted finite state machine built on manually written rules and exception word lists. More recently Arısoy et al. (2007) addressed the same problem by training the LM on lexicalized suffixes and then recovering the surface forms in the ASR output. This technique too required the use of a rule-based morphological analyzer and generator. On the contrary, we propose to handle suffix allomorphy in a data-driven manner. The idea is to define a few *letter equivalence classes* that cover a large part of the morphophonemic processes observed in the language. In our experiments we use the following classes:

$$A=\{a,e\} \quad H=\{ı,i,u,ü\}$$
$$D=\{d,t\} \quad K=\{k,ğ\} \quad C=\{c,ç\}$$

The first two classes address vowel harmony, while the others describe consonant changes frequently occurring between attaching morphemes. Note that defining the classes is the only manual linguistic effort needed by our technique. In the lexicalization phase, the letters of interest are deterministically mapped to their class, regardless of their context (see column 'Stem+Lex.Ending' in Table 2).

At the same time, a reverse index $\mathcal{I}$ is built to store surface forms that were mapped to a lexical form (very unlikely surface forms are discarded by threshold pruning). The LM is subsequently trained on text containing lexicalized endings and $\mathcal{I}$ is used to provide the possible pronunciation variants of each ending in the transcribed lexicon. After recognition, $\mathcal{I}$ is employed to generate the possible surface forms, which are then ranked by two statistical models assigning probabilities to ending surface forms in context. We assume that predicting the first 3 letters of an ending is enough to guess its complete surface form. As for conditioning variable, we use the full stem preceding the lexical ending if frequently observed, or else its last 3 letters only. This results in two models that are linearly combined: the *Stem Model* and the *Stem End Model*, respectively. The intuition behind this is that frequent exceptions to the generic phonological rules can be captured by looking at the whole stem, while for most of other cases knowing a small context is enough to determine an ending's surface form. Here is an example:

Stem Model		Stem End Model
p(+**lar**|kural)=.894	p(+lar|santral)=.026	p(+**lar**|*ral)=.242
p(+ler|kural)<.001	p(+**ler**|santral)=.308	p(+ler|*ral)=.200

Combination weights were set to $\langle 0.8,0.2 \rangle$ to give priority to the larger-context model (*Stem Model*). During post-processing, each lexical ending is assigned the surface form with the highest probability, among those provided by $\mathcal{I}$.

4. Experiments

Two text corpora were defined: *TurTrain* and *TurDev*. Both of them have been collected via web crawling, over two distinct periods (*TurTrain*: Jan 1, 2010 - Feb 15, 2012 and *TurDev*: Feb 16, 2012 - Feb 28, 2012). The same basic cleaning procedures were applied, in particular numbers were expanded (e.g. *2012* → *iki bin on iki*) and punctuation was removed. *TurTrain* resulted in 129.9M words (lexicon size: 837K), while *TurDev* resulted in 3.2M words (99K).

4.1. Language Model Coverage and Perplexity

To evaluate the language modeling component of our ASR system, we measure OOV and PP on *TurDev* and on the reference transcription of *TurTest*, our ASR benchmark. In Table 3 the baseline word-level LM is compared with a series of LMs trained on *'word ending'* segmented data obtained with different PPth values. We recall that lower PPth means more aggressive segmentation by Morfessor. Note that perplexities are not directly comparable with one another, as the number of test tokens changes across settings.

4.2. Morphophonemic Normalization

With PPth equal to 200, the reverse index built on *TurTrain* contains 4355 ambiguous entries, i.e. lexicalized word endings with more than one surface form, and the average number of surface forms per entry is 2.3. To compute the accuracy of the surface form generator, we first lexicalize

[2] In this work we do not directly address stem allomorphy.

		TurTrain		TurDev		TurTest	
Preproc.	PPth	#tokens	lex.size	PP	OOV	PP	OOV
baseline		129.9M	837K	501	1.97	1442	0.94
morph	500	154.2M	733K	184	1.66	365	0.76
morph	300	161.4M	688K	148	1.59	260	0.72
morph	200	170.2M	636K	114	1.51	186	0.68
morph	100	173.5M	605K	105	1.48	169	0.66

Table 3: Impact of unsupervised *word ending* segmentation on number of training tokens and lexicon size; PP and OOV obtained on test sets by the corresponding 5-gram LMs.

the endings found in the development set, then we recover their surface forms in context by applying the models described in Section 3.2. Finally, we compare results with the original version of the text. We find that 27% of the tokens in *TurDev* are ambiguous lexicalized endings, and that 99.7% of them are assigned the correct surface form by our model. From a manual analysis, it also appears that some mismatches are actually due to the presence of wrong surface forms in the original text. In fact, misspellings are extremely common in web-crawled text (e.g. the non-Latin character *'ı'* replaced by *'i'*).

Given the very good performance reported, we integrate the model into our ASR system and measure its impact on language modeling. The OOV rate remains unchanged, but this is not surprising as lexicalization does not concern stems, which are the main responsible for vocabulary growth. Unfortunately, as shown in the row "lex" of Table 4, the effect on perplexity is also negligible.

	TurDev		
	4-gram	5-gram	6-gram
morph200	112.0	114.4	115.1
morph200,lex	112.1	114.0	114.6

Table 4: Effect of data-driven lexicalization on perplexity.

4.3. Speech Recognition

Speech recognition experiments were performed over *TurTest* using the same AMs described in Section 2. Table 5 reports results in terms of WA and, for the morphological case, Half-Word Accuracy (HWA). The latter simply corresponds to measuring WA *before* joining the half-words, which are the true output of the ASR system.

As a first observation, performance is reasonable and close to the state of the art, at least on our small test set. This is an important result, given that no language-specific resources were used on either the acoustic or language modeling side. Secondly, we compare the word-based approach (baseline) with the morphological approaches described above: WA improves from 71.55% to 73.69% (+2.14%) in the best experimental setting, that is 5-grams and PPth=200. In general we see that tuning the value of PPth is important as recognition accuracy varies significantly with it. Indeed, the intermediate values (300 and 200) yield the best performance overall. To our knowledge, previous work did not investigate this point but only used the default setting provided in the tool's distribution. Looking at HWA, trends are somehow different. However, it should be noted that here the number of reference units changes across settings, making values in different rows not directly comparable with one another. As regards the n-gram order, HWA figures confirm the trends observed on WA: 5-grams are better than both 4-grams and 6-grams.

From the last row of Table 5 we see that morphophonemic normalization has a negative effect on accuracy. This is in contrast with the improvements achieved by Arısoy et al. (2007) when applying a similar technique built on a rule-based morphological analyzer. Interestingly, though, the best result in the last row is obtained by the 6-grams, while in all other settings 5-grams are better. In future work we would like to investigate whether normalization can have a positive impact on 7-grams or even higher-order LMs.

	TurTest		
	4-gram	5-gram	6-gram
baseline	71.15| –	**71.55**| –	71.29| –
morph500	71.95|73.30	72.69|74.23	72.49|73.95
morph300	72.89|74.28	**73.69**|75.05	72.69|74.18
morph200	72.36|75.19	**73.69**|76.40	73.49|76.40
morph100	72.56|75.69	73.36|**76.87**	73.23|76.49
morph200,lex	71.69|74.42	72.09|74.86	**73.23**|**76.06**

Table 5: Recognition results in percentage word accuracy and half-word accuracy (WA|HWA).

So far we did not limit the vocabulary size. However, this is a parameter that tends to grow indefinitely with the size of the text corpus, and in our case reached 837K entries. Thus, we only keep the most frequent N entries, and test the effect on three parameters: WA, PP and OOV. Figure 2 reports the results, which highlight how the morphological approach is more robust to this effect as expected.

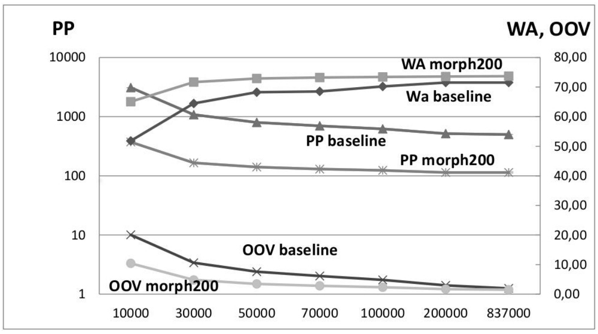

Figure 2: Results depending on lexicon size (*x*-axis).

5. Conclusions

We have shown how a Turkish ASR system with reasonable performance can be built without using language-specific resources: AMs were bootstrapped from those of a well-trained language, while unsupervised segmentation was applied to LM training data. The whole development cycle required only few minor interventions by an expert of the language. Experiments show that word-segmented models are more accurate and robust wrt lexicon size variations. Besides, WA appears to be notably affected by the degree of word segmentation. We have further presented a novel method to perform phonetic normalization of word endings. Intrinsic evaluation is very positive, however the effect on ASR is rather negative. While we plan to further investigate this effect, we hope that our work will inspire further research in under-resourced agglutinative languages.

6. References

Ebru Arısoy, Haşim Sak, and Murat Saraçlar. 2007. Language modeling for automatic turkish broadcast news transcription. In *Proceedings of INTERSPEECH*.

Ebru Arısoy, Doğan Can, Sıddıka Parlak, Haşim Sak, and Murat Saraçlar. 2009. Turkish broadcast news transcription and retrieval. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(5):874 –883.

Mathias Creutz and Krista Lagus. 2005. Inducing the morphological lexicon of a natural language from unannotated text. In *International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning*.

H. Erdoğan, O. Büyük, and K. Oflazer. 2005. Incorporating language constraints in sub-word based speech recognition. In *Automatic Speech Recognition and Understanding, 2005 IEEE Workshop on*, pages 98–103.

Daniele Falavigna and Roberto Gretter. 2011. Cheap bootstrap of multi-lingual hidden markov models. In *Proceedings of INTERSPEECH*, pages 2325–2328.

Christian Girardi. 2007. Htmcleaner: Extracting relevant text from web. In *3rd Web as Corpus workshop (WAC3)*, pages 141–143.

Kimmo Koskenniemi. 1984. A general computational model for word-form recognition and production. In *Proceedings of ACL*, pages 178–181.

Mikko Kurimo, Antti Puurula, Ebru Arısoy, Vesa Siivola, Teemu Hirsimäki, Janne Pylkkönen, Tanel Alumäe, and Murat Saraçlar. 2006. Unlimited vocabulary speech recognition for agglutinative languages. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 487–494.

Kemal Oflazer. 1994. Two-level description of Turkish morphology. *Literary and Linguistic Computing*, 9(2):137–148.

Haşim Sak, Tunga Güngör, and Murat Saraçlar. 2008. Turkish language resources: Morphological parser, morphological disambiguator and web corpus. In *GoTAL 2008*, volume 5221 of *LNCS*, pages 417–427. Springer.