BUILDING AN ARABIC NEWS TRANSCRIPTION SYSTEM WITH WEB-CRAWLED RESOURCES

Arianna Bisazza and Roberto Gretter

HLT research unit, Fondazione Bruno Kessler, 38123 Povo (TN), Italy

(bisazza,gretter)@fbk.eu

Abstract

This paper describes our efforts to build an Arabic ASR system with web-crawled resources. We first describe the processing done to handle Arabic text in general and more particularly to cope with the high number of different phonetic transcriptions associated to a typical Arabic word. Then, we present our experiments to build acoustic models using only audio data found in the web, in particular on the Euronews portal. To transcribe the downloaded audio we compare two approaches: the first uses a baseline trained on manually transcribed Arabic corpora, while the second uses a universal ASR system trained on automatically transcribed speech data of 8 languages (not including Arabic). We demonstrate that with this approach we are able to obtain recognition performances comparable to the ones obtained with a fully supervised Arabic baseline.

Keywords: Arabic, ASR, Morphological segmentation, Lightly supervised training, Under-resourced languages

1. INTRODUCTION[1]

In this paper we describe the efforts made at FBK to build an Arabic ASR system. Two main topics are addressed: linguistic processing and Acoustic Model (AM) training using web-crawled audio data.

The Arabic language is well-known for its rich morphology and for its particular writing conventions that allow for the omission of short vowels and other graphemes. For our linguistic processing, we build on previous work by [1, 2] to address the rich morphology of Arabic with word segmentation. Besides, we choose to model short vowels explicitly, even though these are not strictly required in the ASR output, because this was shown by [3] and others to improve final ASR performance. We then follow an approach similar to that of [4] to generate the transcription lexicon starting from a non-vocalized corpus. Finally, we present a novel technique to reduce the size of the recognizer's Finite State Network (FSN), which embeds phonetic transcriptions into LM, by exploiting language regularities. In particular, we study the patterns of short vowel alternations that appear very often at the end of Arabic words and create a set of eight vowel metasymbols to represent them and obtain considerably smaller FSN's.

AM training, as for it, has been largely investigated by many researchers, who proposed approaches based on the usage of language specific [5, 6], language universal [7, 8] and language adaptive [9, 8] acoustic models. In general, the training of language specific acoustic models represents the best practice to adopt when a sufficient quantity of audio recordings (i.e. hundreds of hours) is available for a given language. On the contrary, when a reduced set of training data (i.e. tens of hours or less) is available for a language, two different approaches can be used: **(i)** cross-language bootstrap [5] of a *target* language's AM starting from that of a well trained *source* language. Bootstrap can be followed by training, or adaptation, using the available set of training data of the target language; **(ii)** training of a universal set of acoustic models using a mix of training data in many languages [7] [8] [9], also possibly followed by language dependent adaptation.

The main issue addressed in this paper is the use of cheaply available speech data to train AMs. In particular, we explore the usage of Euronews[2] data consisting of video news in several languages associated to short texts that often, but not always, provide a partial transcrition of the news. To transcribe the downloaded audio and generate data suitable for AM training we compare two approaches: the first uses a baseline ASR trained on manually transcribed Arabic speech corpora, while the second uses a universal ASR system trained on automatically transcribed speech data of 8 languages (not including Arabic). Because automatic transcriptions provide a cheap, but often noisy training material, we also design a technique for data selection that exploits the short summarizing text associated to each news on the Euronews portal.

This paper is organized as follows. § 2 describes the audio and text databases used in the paper, § 3 the linguistic processing for Arabic, § 4 the FBK ASR system, and § 5 the various AM training conditions. In § 6 we report and discuss experimental results and in § 7 we draw our conclusions and outline future work.

2. AUDIO AND TEXT DATA

2.1. Available databases

Arabic is not what is generally called an under-resourced language. Several speech corpora exist, including both orthographic and sometimes phonetic information; textual data is also available for instance in the Gigaword series. Concerning the linguistic side, the Buckwalter morphological analyzer[3] is a tool that analyzes Arabic tokens and returns linguistic and phonetic information. However, the cost of available databases for commercial purposes pushes towards the realization of ASR systems which don't use such databases, but are trained on cheaply available data.

In this paper, baseline acoustic models were trained using two manually transcribed speech corpora: *ELRA-S0219: NEMLAR Broadcast News Speech Corpus* and *ELRA-S0157: NetDC Arabic BNSC*. These contain Arabic speech together with fully vocalized transcriptions that are easy to convert into a phonetic sequence, unlike standard Arabic texts. Language Models (LMs)

[1] THIS WORK HAS BEEN PARTIALLY FUNDED BY THE EUROPEAN PROJECT EU-BRIDGE, UNDER THE CONTRACT FP7-287658.

[2] http://www.euronews.com

[3] LDC2004L02: Buckwalter Arabic Morphological Analyzer Version 2.0, http://www.qamus.org/morphology.htm

are trained on the *LDC2007T40: Arabic Gigaword Third Edition*. The resulting LMs were used by all the systems in test mode, in order to allow a fair comparison among different AMs.

2.2. Euronews audio data from the WEB

Euronews is a valuable source of multilingual data. It is a TV satellite channel which broadcasts news in several languages, whose number is growing over time. Its portal contains a collection of news in various languages, daily updated. Every news is composed by a video and an accompanying text, which is sometimes a summary of the news and sometimes a rather precise transcription of parts of the news.

At present, Euronews covers 13 languages (see Table 1). At FBK we download texts and videos on a daily basis, keeping where possible alignment among the same news in different languages: this means that a link may exist between, for instance, a news in Arabic and the same news in Portuguese. Note, however, that news in different languages are *not* exact translations of each other. To give an idea of the amount of available data, we show in Table 1 the data collected in one month, expressed in speech duration and number of words.

From an ASR perspective, data cannot be considered really clean because several phenomena take place: often in the case of interviews, some seconds of speech in the original language are played before the translation starts; there is the presence of music; sometimes the text associated to the video is not in the expected language.

After downloading, we extract the audio data from the video, and store it at 16kHz. Concerning the text, the HTML page is processed in order to extract only the relevant textual information, as described in [10]. Some other information like publication date, downloading date, original URL, cross lingual links, coding are also retained.

The Euronews corpus used in this work is composed of the Arabic data downloaded between January 2013 and May 2013, which amounts to 4406 files (news), 107.4 hours (88 seconds per news on average). The corresponding texts amount to 660,428 words (150 words per news on average).

2.3. Test data

As test data, we use the broadcast news section of the 2003 NIST Rich Transcription Evaluation Data (RT03-bn).[4] This benchmark is composed of about one hour audio coming from two different broadcasts and corresponding to 6.7K words.

3. LINGUISTIC PROCESSING FOR ARABIC

3.1. Text pre-processing

Pre-processing of the Arabic LM training data includes the following steps:

Cleaning and number processing. The corpus is filtered character by character and non-standard Arabic Unicode characters are replaced by their standard Arabic equivalent. Kashida/tatweel characters are removed and digits are normalized. All diacritics (including short vowels) are removed and the "alef" character is normalized. Finally, numerals are expanded.

Dictionary-based morphological segmentation. In this phase we follow the approach of [1, 2] and isolate common adfixes based on string matching. The list of adfixes includes the following items and their combinations for a total of 34 entries: conjunctions *w+, f+*, prepositions *l+, k+, b+*, future marker *s+* and pronominal suffixes *+k, +km, +h, +hm, ...* Note that the article *Al+* is not isolated because this was shown to cause many prefix deletion errors by [2]. As proposed in the same paper, we only segment infrequent words and leave the N most frequent word types of the dictionary in their complete form. 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. Similarly to [1], we split an adfix only if the resulting stem is contained in the original dictionary.

language	number of news	duration in hours	avg. dur. in seconds	running words	avg. wrd. per file
Arabic	918	21.1	82.8	128,300	139.8
English	921	21.9	85.6	181,295	196.8
French	919	21.5	84.3	173,107	188.4
German	929	21.6	83.6	153,521	165.2
Greek	485	11.7	87.1	94,438	194.7
Italian	911	21.2	83.6	159,237	174.8
Persian	875	19.7	81.1	166,026	189.7
Polish	183	3.1	60.9	23,635	129.1
Portuguese	913	21.3	83.8	156,713	171.6
Russian	914	21.3	83.8	139,110	152.2
Spanish	927	21.5	83.5	172,062	185.6
Turkish	876	19.8	81.4	126,571	144.5
Ukrainian	895	20.1	81.0	131,978	147.5

Table 1. Data collected for all Euronews languages in one month (January 2013).

The segmentation procedure is illustrated in Table 2. Here, the first form is not segmented because of its high frequency, whereas the second is split into prefix and stem, the third into prefix, stem and suffix. Adfixes are marked with a '+' sign and reattached to their stem in the recognizer's output.

Full form	Segmented	Gloss
بدوره	بدوره	in his turn
بدورها	بـ + دورها	in her turn
بدورهم	بـ + دور + هم	in their turn

Table 2. Example of dictionary-based morphological segmentation.

3.2. Pronunciation lexicon

Generating the phonetic transcription of Arabic would be rather trivial if it wasn't for short vowels and consonant gemination. These, in fact, are only seldom marked in Arabic modern text[5] but are nevertheless important for AM modeling [3]. In order to vocalize the written data used to build the language model, we use the Buckwalter analyser, which returns all possible morphological analyses and corresponding vocalizations of any Arabic word. Given a fully vocalized and diacritized word form, grapheme-to-phoneme transcription is performed by a simple set of rules, including specific symbols for geminated consonants and long vowels.

Our choice is then to treat the different vocalized versions of a word as its multiple pronunciations, similarly to what was done in [4]. Table 3 shows three lexicon entries with their multiple vocalizations/pronunciations in Arabic SAMPA symbols.[6] Notice that different "pronunciations" can actually correspond to various grammatical cases but also to different lemmas etc. The

[4] LDC2007S10: http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp? catalogId=LDC2007S10

[5] Note instead that the speech corpora used to train the baseline AM are provided with fully vocalized and diacritized transcripts (cf. Section 2.1).

[6] http://www.phon.ucl.ac.uk/home/sampa/arabic.htm

Word Non-voc.	 Vocalized	Pronunciation	Gloss
بدوره	بِدَورِهِ	b i d a w r i h i	in his turn
	بِدُورِهِ	b i d u: r i h i	by his houses
	بُدُورَهُ	b u d u: r a h u	his full moon (*acc*)
	بُدُورِهِ	b u d u: r i h i	his full moon (*gen*)
	بُدُورُهُ	b u d u: r u h u	his full moon (*nom*)
لوكسمبورغ	–unk–	l u: k s m b u: r G	Luxembourg
لكسمبورغ	–unk–	l k s m b u: r G	Luxembourg

Table 3. Three lexicon entries with their multiple vocalizations/pronunciations in SAMPA.

average number of pronunciations per word in our Gigaword lexicon (containing all words with more than 50 occurrences) is 4.6. As suggested by the first entry of Table 3, it could be useful to filter out very unlikely interpretations of the non-vocalized form, but this is hard to implement in the absence of large vocalized corpora. In the future we would like to further investigate this problem.

As for the words that the Buckwalter could not analyze, we simply feed them, non-diacritized, to the grapheme-to-phoneme module. Although approximate, this is often a reasonable choice because foreign names are typically transliterated into Arabic using mostly long vowels. In the last two rows of Table 3, we can see two Arabic spellings of the word "Luxembourg" that were unknown to the Buckwalter analyzer and were transcribed without undergoing vocalization. In this case, the first spelling results in an almost perfect phonetic transcription, while the second is missing the first vowel.

The phonetic transcription of the 34 adfixes used for dictionary-based morphological segmentation is produced manually. Following [4], two pronunciation variants are generated for each word beginning with *unstable hamza*.

3.3. Vowel metasymbols for FSN reduction

Due to its rich morphology, Arabic text is typically characterized by a high type/token ratio. While morphological segmentation is indeed helpful to reduce vocabulary size, we found in our initial experiments that our recognizer's Finite State Network (FSN) – which embeds phonetic transcriptions into LM – was still huge due to the many vocalizations associated to each word in the lexicon. As a result, we were obliged to reduce the order of the n-gram model or to consider a smaller dictionary.

To address this problem, we designed a technique that reduces the FSN size by exploiting linguistic regularities. In particular we observed that short vowel alternations often follow specific patterns corresponding, for instance, to a set of nominal cases. An example is given by the word الدور presented in Table 4. The last vowel of this word can be *a, i* or *u* according to its grammatical case: definite accusative, definite genitive or definite nominative, respectively. Since this pattern is extremely common in Arabic, we create a specific metasymbol for the class of vowels [aiu] and collapse the three corresponding vocalized forms into one.

Vocal. form	Pronunciation	Gloss	W/ metasymbols
اَلدُّورَ	a dd u: r a	the turn (*acc*)	a dd u: r **[aiu]**
اَلدُّورِ	a dd u: r i	the turn (*gen*)	
اَلدُّورُ	a dd u: r u	the turn (*nom*)	

Table 4. Example of metasymbol representing three definite noun cases.

This is the full list of metasymbols used by our system:

[KNaiu]	undefinite and definite masculine noun cases;
[FKNaiu]	undefinite and definite feminine noun cases;
[KNiu]	undefinite and definite feminine plural noun cases;
[aiu]	definite noun cases or past verb person markers;
[au]	verbal mood subjective and jussive markers;
[Ki]	definite and undefinite genitive case;
[iu]	definite feminine plural noun cases;
[ai]	2^{nd} pers. object pronoun and other minor patterns;

where F, K, N correspond to *fatha* /an/, *kasra* /in/, and *damma* /un/, respectively. The application of these metasymbols to our Gigaword lexicon brings down the average number of pronunciations per word from 4.6 to 3.2. The metasymbols are then expanded in the FSN used during ASR *after* its construction. As a difference from existing FSN minimization techniques that can be used to shrink an already compiled FSN, our method makes it possible to construct a smaller FSN in the first place and is therefore particularly valuable for system developers that have access to limited computational power.

We will show in Section 6.1 how this method allows us to use larger LM's and thus improve the final ASR performance.

4. FBK'S ASR SYSTEM

The FBK transcription system used in all the experiments is based on several processing stages, briefly described here:

Segmentation, classification and clustering. The speech signal is divided in segments, based on a voice activity detector. The segments are mapped to several classes by a GMM classifier, and grouped in homogeneous clusters according to a BIC criterion. The clusters are used by the following acoustic normalization procedures. **Acoustic features extraction.** From the waveform, a sequence of 52-dimensional feature vectors is extracted, including 13 mel-scaled cepstral coefficients and their first, second and third derivatives. **Unsupervised acoustic features normalization.** The feature vectors undergo a first stage of normalization, computing a specific CMLLR transform for each segment cluster, with respect to a 1024-Gaussians GMM trained on the whole training set. **HLDA projection.** The 52-dimensional normalized feature vectors are projected in a 39-dimensional space, by means of an HLDA transformation. **First decoding step.** A first decoding step is performed on the resulting acoustic features, applying an AM based on tied-states cross-word triphone HMMs and an n-gram LM. This hypothesized word sequence is used as a supervision for the following supervised normalization. **Supervised acoustic features normalization.** The feature vectors are processed to perform a further normalization based on CMLLR transforms, this time exploiting the approximate transcription output by the first decoding step and a set of tied-states cross-word triphone HMMs with a single Gaussian per state. **Second decoding step.** A second decoding is performed on the normalized features, providing the final output.

5. AM AND LM TRAINING CONDITIONS

This section presents the various AMs and LMs used in the experiments. We recall that our final goal is to build acoustic models using only web-crawled audio data. To produce the transcription of the downloaded audio, which will be used for the final AM training, we compare two approaches: the first uses a supervised ASR baseline trained on manually transcribed Arabic corpora, while the second uses a universal ASR system trained on automatically transcribed speech data of 8 different languages.

5.1. Supervised Arabic AM (baseline)

Baseline acoustic models were trained using a subset of the corpora *ELRA-S0219* and *ELRA-S0157* introduced in Section 2.1. These databases include a total of about 60 hours of Standard Arabic news broadcasts, recorded from different radio stations. Both provide fully vocalized and diacritized transcriptions. Out of this, we selected 44.9 hours (32.2 from NEMLAR and 12.7 from NetDC) that were used to train AMs in a completely supervised way.

5.2. Universal AM

Our universal acoustic models were trained on audio material in 8 languages, automatically transcribed, and employ a set of 135 phones, many of which are shared among different languages. These Universal Phone Set (UPS) HMMs were trained on about 40 hours of speech equally divided among: English, Flemish, French, German, Italian, Russian, Spanish, and Turkish (i.e. Arabic is not included). This set of acoustic models is part of a Language IDentification (LID) system recently developed in FBK [11].

It is worth noting that not all the 69 phones composing the Arabic phone set are contained in the UPS. Thus, we manually mapped each missing phone into the most similar phone existing in the UPS, as shown in Table 5. For instance, the emphatic *t'* corresponding to the Arabic letter ط was mapped to the non-emphatic *t* (ت).

? → @bg	?? → @bg	?‘ → G	?‘?‘ → G
X → hh	XX → hh	X\ → hh	X\X\ → hh
q → k	qq → kk	D → dh	DD → dh
D‘→ dh	D‘D‘→ dh	G → R	GG → R
T → th	TT → th	ZZ → Z	d‘→ d
d‘d‘→ dd	s‘→ s	s‘s‘→ ss	t‘→ t
t‘t‘→ tt	ww → w	x → C	xx → C
zz → z	f‘→ f		

Table 5. Mapping between Arabic phones not in the UPS and similar existing phones. Arabic SAMPA symbols are used, while @bg means silence.

5.3. Gigaword LM

This LM was trained on the *Arabic Gigaword Third Edition (LDC2007T40)*, a comprehensive archive of newswire text data acquired from Arabic news sources by the LDC (about 2 billion words). We used a subset of these data amounting to 654M tokens, which resulted in about 257M trigrams. We use this LM both to automatically transcribe the training audio data and to perform recognition on the test set.

5.4. Euronews LM

This LM was trained on the summarizing text data associated to the audio data on the Euronews portal. It is a very small LM (660K tokens, 614K trigrams) which is focused on the collected audio material. We use this model only to automatically transcribe the training audio data.

5.5. Unsupervised training

AM training is based on statistical engines capable of capturing the basic sounds of a language, starting from an inventory of pairs ⟨utterance - transcription⟩. When only untranscribed audio material is available, this can be processed by an initial ASR to obtain automatic transcriptions. Despite the obvious presence of transcription errors, these automatic transcriptions 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. This procedure roughly corresponds to the first iteration of the approach described in [12] and consists of the following steps:

- automatic transcription of all the 107 hours of downloaded audio data, by a previously trained ASR system;
- training new AMs with the standard procedure on the automatically transcribed data;
- perform ASR on the test set, using the Gigaword LM and the newly created AMs.

According to [12], we can expect to improve performance by running other iterations with the newly created AMs, however in this work we only perform one iteration.

5.6. Acoustic data selection

As explained in Section 2.2, our web-crawled Euronews corpus contains summarizing texts that can be a partial transcription of the speech content. Using this form of supervision, we can select only those speech segments for which an accurate transcription was produced by the ASR. More specifically, we detect those portions in which the supposed transcription (we will call it *reference*) and an automatic transcription agree. To this end, the output of the ASR is aligned with the reference transcriptions, and only the matching segments are selected retaining ASR time markers.

This procedure allows us to extract reliable speech data, at the cost of loosing part of the material. Note that the amount of "lost" data depends on the ASR perfomance - in particular, on the AM and LM characteristics - but also on the accuracy of the reference transcription. For instance, by inspecting part of the material, we found that the reference was often an accurate transcription of only part of the news, and that caused the loss of data which had been perfectly transcribed. Still, we can count on this procedure to obtain a small but very high-quality training corpus out of large and noisy data.

6. EXPERIMENTS

In this section we evaluate our technique for FSN reduction based on vowel metasymbols. Then, we measure the performance on the test set in different AM training conditions.

6.1. Vowel metasymbols

The size of the LM, and consequently that of the recognizer's FSN, can be limited in several ways. One of these consists in reducing the dictionary size by ignoring words that occurred rarely in the LM training corpus. In our preliminary experiments with the Gigaword corpus, we had to set the word minimum frequency to 400, resulting in a lexicon of 71K word (386K pronunciations/vocalizations), because larger dictionaries led to extremely large FSN's and compilation failures.

However, considering the rich morphology of Arabic and the finding of [4] for instance, we expected that using a larger dictionary would be very beneficial for the final ASR performance.

Indeed, by applying the vowel metasymbols described in Section 3.3, we were able to prove this. As shown in Table 6, the metasymbols allow for a 18%–25% reduction of the FSN size (from 102M to 84M states, from 201M to 150M transitions), without any loss in WA. Using this technique we could set the minimum frequency to 50 and compile the FSN with a 212K-word lexicon, which led to a notably higher WA: that is, from 79.3% to 80.1%. We then decided to include the vowel metasymbols in all the remaining experiments.

System	lexicon			FSN size		WA
	m.freq	#words	#pron.	#states	#trans.	
baseline	400	71K	386K	102M	201M	79.3%
	50	212K	983K	*— failed —*		*– NA –*
with vowel metasymbols	400	71K	248K	84M	150M	79.3%
	50	212K	686K	91M	163M	80.1%

Table 6. Effect of vowel metasymbols on FSN size (states and transitions) and ASR performance (WA on test). All systems were trained on ELRA speech and Gigaword text corpora.

6.2. Training AM on automatically transcribed data

In the remaining experiments, the FSN used to perform recognition is the one with vowel metasymbols and word minimum frequency set to 50, which correspond to 80.1% WA using the Baseline AMs in recognition.

Table 7 shows results in terms of WA when using different methods to generate the AM training data. In each row we report the experiment ID, the AM and LM used to perform ASR on the training data, the amount of speech data retained for the final AM training, and the WA obtained by the newly trained AM. Even when data selection wasn't applied, the amount of retained speech was less than the initial 107.4 hours because music and noisy segments had to be discarded.

First we note that the best result overall (80.7%, B-E) is obtained when using the Baseline AM and the small focused Euronews LM to transcribe the training data. This result is even better than the one achieved by the fully supervised baseline on the test (80.1%). Using the bigger LM does not seem to help (80.3%, B-G). On the contrary, bootstrapping AMs with a poor AM (Universal) significantly decreases performance (74.4%, U-E). We then consider the impact of data selection (DS). When starting from a high WA, our DS procedure slightly degrades performances (from 80.7% to 80.3% B-E-DS), or simply leaves it unchanged (from 80.3% to 80.3% B-G-DS) while reducing the training material by almost a half (e.g 99.9 to 46.0 hours). However, DS appears to be extremely helpful when starting from a low WA (from 74.4% to 79.0% U-E-DS). Notice that, in this last scenario, only *less than a third* of the available training material was used (29.5 hours). These results suggest that our DS procedure is effective both for discarding useless data and speed up the final AM training process, and for discarding extremely noisy data that could otherwise deteriorate the models.

Exp. ID	ASR system used to transcribe the training		DS	resulting training hours	WA
	AM	LM			
fully sup.	– trained on manual transcripts (44.9 h) –				80.1%
B-E	Arabic sup.	Euronews	-	99.7	80.7%
B-E-DS	Arabic sup.	Euronews	+	46.6	80.3%
B-G	Arabic sup.	Gigaword	-	99.9	80.3%
B-G-DS	Arabic sup.	Gigaword	+	46.0	80.3%
U-E	Universal	Euronews	-	98.9	74.4%
U-E-DS	Universal	Euronews	+	29.5	79.0%

Table 7. WA on the test set with different setups to get automatic transcriptions of the AM training data. DS stands for data selection. U-* rows don't use commercial corpora at all.

7. CONCLUSIONS

We have presented an approach to build an Arabic news transcription system with web-crawled resources. In particular, we have shown that very high-quality AM training material can be obtained by automatically transcribing downloaded audio data with a supervised ASR baseline. This material can then be used to build a new ASR system that makes no use of costly manual transcription corpora. More importantly, we have shown that the use of manual transcriptions in the targeted language can be totally bypassed by using a language universal ASR system trained only on other languages.

On the language-specific side, we have presented a novel method, called *vowel metasymbols*, to cope with the large number of different phonetic transcriptions typically associated to non-diacritized Arabic words. Despite the large previous work done on Arabic ASR, we believe this research direction should be further explored, for instance to design a way to automatically filter out unlikely vocalizations from the pronunciation lexicon.

8. REFERENCES

[1] B. Xiang, K. Nguyen, L. Nguyen, R. Schwartz, and J. Makhoul, "Morphological decomposition for arabic broadcast news transcription," in *ICASSP*, 2006, pp. 1089–1092.

[2] L. Lamel, A. Messaoudi, and J-L. Gauvain, "Investigating morphological decomposition for transcription of arabic broadcast news and broadcast conversation data," in *Interspeech*, 2008, pp. 1429–1432.

[3] M. Afify, L. Nguyen, B. Xiang, S. Abdou, and J. Makhoul, "Recent progress in Arabic broadcast news transcription at BBN," in *Interspeech*, 2005, pp. 1637–1640.

[4] A. Messaoudi, J-L. Gauvain, and L. Lamel, "Arabic broadcast news transcription using a one million word vocalized vocabulary," in *ICASSP*, 2006, pp.1093-1096.

[5] T. Schultz and A. Waibel, "Fast Bootstrapping of LVCSR Systems with Multilingual Phoneme sets," in *Eurospeech*, 1997, pp. 371–374.

[6] T. Schultz D. Koll and A. Waibel, "Japanese LVCSR On the Spontaneous Scheduling Task with JANUS-3," in *Eurospeech*, 1997, pp. 367–370.

[7] J. Kohler, "Multilingual Phoneme Recognition Exploiting Acoustic-Phonetic Similarities of Sounds," in *ICSLP*, 1996, pp. 2195–2198.

[8] H. Lin, L. Deng, D. Yu, Y. Gong, A. Acero, and C.H. Lee, "A Study on Multilingual Acoustic Modeling for Large Vocabulary ASR," in *ICASSP*, 2009, pp. 4333–4336.

[9] T. Schultz and A. Waibel, "Language independent and language adaptive acoustic modeling for speech recognition," *Speech Communication*, vol. 35, pp. 31–51, 2001.

[10] C. Girardi, "HtmlCleaner: Extracting Relevant Text from Web Pages," in *Proceedings of WAC3*, 2007, pp. 15–16.

[11] D. Giuliani and R. Gretter, "Esperimenti di identificazione della lingua parlata in ambito giornalistico," in *AISV*, 2013.

[12] D. Falavigna and R. Gretter, "Cheap bootstrap of multi-lingual hidden markov models," in *Interspeech*, 2011, pp. 2325–2328.