Morphological Decomposition for Arabic Broadcast News Transcription

Bing Xiang, †Kham Nguyen, Long Nguyen, Richard Schwartz, and John Makhoul

BBN Technologies, 10 Moulton St., Cambridge, MA, 02138, USA
†Northeastern University, 360 Huntington Ave., Boston MA 02115, USA

{bxiang,ln,knguyen,schwartz,makhoul}@bbn.com

Abstract

In this paper, we present a novel approach for morphological decomposition in large vocabulary Arabic speech recognition. It achieved low out-of-vocabulary (OOV) rate as well as high recognition accuracy in a state-of-the-art Arabic broadcast news transcription system. In this approach, the compound words are decomposed into roots and affixes in both language training and acoustic training data. The decomposed words in the recognition output are re-joined before scoring. Four algorithms are experimented and compared in this work. The best system achieved 1.7% absolute reduction (8.7% relative) in word error rate (WER) when compared to the 64K-word baseline. The recognition performance of this system is also comparable to a 200K-word recognition system trained on the normal words. In the meantime, the decomposed system is much faster in terms of speed and also needs less memory than the systems with larger than 64K vocabularies.

1. Introduction

In recent years, decomposition of compound words in morphologically complex languages has been addressed in a number of studies in areas such as information extraction and automatic speech recognition (ASR). The major problem for these languages, such as German, Dutch, and Arabic, is the huge number of compound words which result in high out-of-vocabulary (OOV) rate for ASR systems using finite-sized lexicons. A much larger recognition lexicon has to be used to achieve the similar OOV rate as for a non-compound language.

In [1], a German vocabulary was decomposed into morpheme-based units. The smaller dictionary resulted in a lower OOV rate and a 30% speed improvement during recognition. But the recognition accuracy degraded when compared to the word-based recognition system. A frequency-based approach, also in German speech recognition, was proposed in [2], where the best recognition result was achieved by not decomposing the compound words that occur more than 15 times in the training. A less than 1% relative improvement was obtained in recognition accuracy. A similar result was reported in a Dutch speech recognition system[3]. The top 5K to 20K most frequent words were kept in the recognition lexicon. The WER was reduced from 39.8% down to 39.1%. A different way of keeping frequent compound words was explored in [4] for German. Data-driven compound word splitting was followed by iterative recombination of high frequency combinations. Although the OOV rate was reduced by 35%, the recognition accuracy decreased by 5%.

Despite that Arabic is the sixth most widely spoken language in the world, there has been relatively little research on morphology-based Arabic speech recognition. Arabic nouns and verbs are derived from roots, then with templates applied and affixes attached.

For example, the word "wktAbhm" (written in Buckwalter format) consists of the prefix "w" ("and"), root "ktAb" ("book") and suffix "hm" ("their"). It is reported in the literature that the number of unique Arabic words (or surface forms) is as large as 6 x 10^{10} due to the morphological complexity [5]. So morphological decomposition is an appealing approach to lower the OOV rate and reduce the language training data sparsity in Arabic speech recognition. In [6] and [7], morphemes and a few other related features were used in factored language modeling within an N-best rescoring framework for Egyptian or Levantine Arabic telephone speech recognition. Small improvement (less than 2% relative) was obtained. A slightly larger improvement (3% relative) was reported in [8] when a similar morphology-based factored language model was used in all passes.

In this work, we concentrate on the transcription of Arabic broadcast news and utilize morphological decomposition in both acoustic and language modeling in our system. Four different algorithms of compound splitting were explored and compared to the baseline system trained on original Arabic words. Significant WER reduction as well as OOV rate reduction are achieved by the decomposition systems. Also, the best system is comparable to the systems trained on the normal words with larger than 64K recognition lexicons.

The paper is organized as follows. In Section 2, we describe the training and test data used in this work. Section 3 briefly introduces the text normalization and automatic vowelization procedure that is necessary for the decomposition. In Section 4, we present the four different algorithms. The recognizer and experimental results are presented in Sections 5 and 6. Section 7 concludes the paper.

2. Training and Test Data

The acoustic training corpus used in this work consists of 150 hours of speech data. These include 28 hours of data from the FBIS corpus. The rest of the training data were automatically selected from two other corpora via light supervision [9]. Among them, 67 hours of data was selected from the TDT4 Arabic corpus available from LDC. And the remaining 55 hours of speech data was selected from an in-house broadcast news database that contains data from various sources.

Our language model training corpus is a pool of around 400 million-word text. It includes the data from the Gigaword Arabic corpus, TDT4 Arabic corpus, and a few other sources. We also downloaded some data from the website of Aljazeera. All the training data cover various time periods from 1994 to October 2003.

To evaluate the recognition performance, we used the BBN 2005

Arabic development set (Dev05) as the test set. It consists of 3.8 hours of data from 9 episodes broadcast by *Aljazeera*, *Dubai Television* and *Lebanese Broadcasting Corporation* in November 2003.

3. Text Normalization and Vowelization

Due to writing conventions in Arabic, sometimes the same word has different written forms. This is especially true for words that start with the letter "hamza". So a text normalization procedure is needed to minimize the impact from inconsistency. Also, the short vowels are usually missing from most Arabic corpora. Our previous work [10] showed that a vowelized system achieved significant improvement compared to a grapheme system. The vowelization procedure utilizes two resources available from LDC: the Buckwalter morphological analyzer and the Arabic Treebank corpus. The major steps in the text normalization and vowelization include:

- Text normalization: Given an un-normalized lexicon, map all different forms of "hamza" ("$<$", "$>$", and "|") at the beginning of the word, or after the popular Arabic prefixes "Al" and "w", to "alif" (letter "A"). Also, for certain frequent words map the "alif maksura" ("Y") at the end of the word to "yeh" ("y") or vice versa.

- Morphological analysis: Each word is passed to the Buckwalter morphological analyzer (version 2.0), and all the vowelizations corresponding to the output solutions are retained.

- Search in Treebank: If a word is not found in the previous step, it is searched in the Arabic Treebank dictionary. The found words are then merged with the dictionary created in last step.

- Phonetic transcription: Several phonological rules are applied to the pronunciations in the merged dictionary to create the final dictionary with vowelized pronunciations.

With the above procedure, the average number of pronunciations for each word in the vowelized dictionary is around 5. There are 38 phonemes in total in the phoneme set.

4. Morphological Decomposition

In this section we describe the affix set we used in this work, then propose four different algorithms for morphological decomposition.

4.1. Affix Set

Morphological analysis itself has been largely studied in the past [5]. Various techniques, including the one based on finite state transducers, were developed. In this work, we propose a simpler approach for the purpose of reducing OOV rate and also improving speech recognition performance. Instead of applying complicated linguistic rules, we start from a fixed set of prefixes and suffixes and then decompose the compound words into roots and affixes. This fixed set is a subset of the affixes used in [11]. Specifically, the affixes that we used are:

- 12 prefixes: Al, bAl, fAl, kAl, ll, wAl, b, f, k, l, s, w

- 34 suffixes: An, h, hA, hm, hmA, hn, k, km, kn, nA, ny, t, th, thA, thm, thmA, thn, tk, tkm, tm, tnA, tny, tynA, wA, wh, whA, whm, wk, wkm, wn, wnA, wny, y, yn

Then we made two additional changes. First, we created a variant for each of the prefixes ending with "Al" in order to distinguish the prefixes followed by a *sun* letter from those followed by other letters. This change resulted in 5 additional prefixes to increase the total number of prefixes to 17. Another change was to attach a tag, e.g. "_", at the end of each prefix and the beginning of each suffix to signify the affixes. When a recognition system outputs decomposed hypotheses, it is deterministic to join the affixes with their adjacent roots to get normal Arabic words.

4.2. Constraints on Compound Decomposition

During the compound decomposition, we applied some constraints to restrict the word splitting. In total, there are four constraints used by various algorithms described later. When the beginning or ending part of an Arabic word matches one of the prefixes or suffixes, one or more constraints listed below were applied.

1. The root must be at least two letters long.

2. The beginning or ending part of the pronunciations of the original word must match one of the pre-determined pronunciations of an affix.

3. The root must exist in the 784K-word master dictionary.

4. The original word does not belong to the top N most frequent decompoundable words.

The master dictionary in constraint 3 was generated from the vowelization of 1.3 million unique words occurring in the language training data. The list of the pre-determined affix pronunciations used in constraint 2 was obtained with the help of the Buckwalter morphological analyzer, which provided possible short vowels and other diacritics for each affix. After applying the same phonological rules as in the last step of the automatic vowelization procedure, we obtained a list of pronunciations for all affixes.

4.3. Algorithm I

In Algorithm I, we only apply constraint 1 to avoid short roots. This is also based on the observation that some of the frequent morphologically-atomic words would have been split if without this constraint. The decomposition results on three different lexicons are shown in Table 1.

Org Lex	Org OOV	Decomp Lex	Decomp OOV
64K	4.18	25K	1.84
180K	1.69	52K	0.87
220K	1.46	60K	0.79

Table 1: Decomposition results of Algorithm I (Org: original; Lex: lexicon; Decomp: decomposition)

We can see that a 220K lexicon can be reduced to 60K roots and affixes. A significant OOV rate reduction on the Dev05 test set was also achieved. The OOV rates shown in the table are normalized

as defined in Eq. (1).

$$OOV_{norm} = OOV * \frac{N_d}{N_{org}}, \qquad (1)$$

where N_d is the number of words in the decomposed data and N_{org} is the number of original words. The ratio between these two is 1.65 on the Dev05 data. The same compound splitting is applied to the data in both acoustic training and language training corpora.

4.4. Algorithm II

In Algorithm II, both constraints 1 and 2 are applied. It includes the following steps:

- Decompose all the words in the master dictionary first to get a word splitting list.
- Decompose all language model training data based on the splitting list.
- Collect all N-gram (N=1,2,3) counts and select the top N_1 most frequent unigrams and top N_2 most frequent bigrams and trigrams. A selected N-gram could contain parts of an original word or several parts of adjacent words. Each of these N-grams is converted into a new compound word.
- The pronunciations of the new compound words are created through concatenation of the pronunciations of each part in the corresponding N-grams. Then the pronunciations with low pronunciation probabilities are pruned to avoid huge number of pronunciations after concatenation.
- Compound the decomposed language training data with the list of new compound words.
- Decompose and re-compound the acoustic training data using the splitting and compounding list obtained above.

4.5. Algorithm III

Algorithm III is similar to Algorithm II, but with one more constraint, i.e. constraint 3, during the word splitting in the first step. Also, when selecting the most frequent N-grams, we only pick those existing in the original master dictionary. In this way, the most frequent compound words are retained in the lexicon.

4.6. Algorithm IV

In Algorithm IV, all four constraints are applied. Instead of decomposing all the language training data to find the most frequent N-grams as in Algorithm II and III, we start from an original lexicon with certain size, e.g. 95K, and decompose it into a lexicon with a smaller size. The top N most frequent decompoundable words are kept unchanged. Here a "decompoundable" word means the word would have been split into root and affixes if only the first three constraints were enforced. The effect of the fourth constraint is in fact similar to the additional restriction in Algorithm III. Both algorithms guarantee that all the words except affixes in the decomposed lexicon can be found in the original dictionary.

5. Recognition System

5.1. Recognizer

The recognizer used in this work is similar to that described in [10]. The decoding stage is comprised of two decoding stages,

Lexicon	OOV	Set Bgr	Speed	Unadapt	Adapt
64K	4.18	104M	5.0xRT	24.6	19.5
100K	3.12	125M	6.6xRT	23.9	18.6
200K	1.70	156M	7.5xRT	23.1	17.7
300K	1.32	170M	7.9xRT	23.0	17.5

Table 2: Results from systems with various sizes of vocabularies

unadapted decoding and adapted decoding. In each stage, the decoding employs a multi-pass search strategy. The forward pass uses a simple acoustic models, State Tied Mixture (STM) model, and a bigram language model, and outputs the most likely words at each frame together with their scores. The backward pass then uses the output of the forward pass to guide a Viterbi beam search with a state clustered within-word quinphone acoustic model and a trigram language model. A lattice is also generated. Finally, we do lattice rescoring using a state clustered cross-word quinphone model. The top scoring hypothesis represents the system's recognition output.

All the acoustic models in this work were trained under the maximum likelihood criterion. The language models were back-off N-gram models estimated from either the original or decomposed language training data. Witten-Bell smoothing was applied during the training.

5.2. Vocabulary Size Beyond 64K

The decoding lexicon in the baseline system consists of 64K words selected based on the occurring frequency in the 400M-word language training corpus. 64K (or 2^{16}, to be exact) has been the typical limit for large vocabulary speech recognition systems. It is mainly due to the fact that each of the 64K words in the lexicon can be encoded in two bytes in a straight-forward implementation. To exceed this limit, more bytes have to be allocated. Due to the demand of reducing the OOV rate with larger vocabulary for Arabic speech recognition, we upgraded our software to accommodate lexicons larger than 64K by using 4 bytes to encode the word ID. Consequently, the memory requirement in the recognizer is larger than 4GB. Therefore, the recognizer can run comfortably only on machines using 64-bit processors and having more than 4GB of physical memory.

6. Experimental Results

6.1. Results from Large Vocabulary Systems

The experimental results from the systems trained on normal words are shown in Table 2. The 64K baseline is compared to three other systems with larger recognition lexicons. Table 2 listed the OOV rates, the number of set bigrams used in the forward pass, the system speed and the WERs in unadapted and adapted decoding. As expected, a larger vocabulary reduced the OOV rate significantly. The recognition accuracy was also improved. However, these systems need more memory for the language model and also more computation, especially in the forward pass during the search. The overall speed of the 300K system was about 60% slower than the baseline (from 5.0xRT to 7.9xRT). Also, these large systems can only run on 64-bit machines that we do not have many currently, which limited our efforts in the development.

6.2. Comparison of Four Decomposition Algorithms

As mentioned earlier, we experimented with four different algorithms for compound decomposition. They differ from each other in terms of various constraints. The results on the Dev05 test set are listed in Table 3. All the OOV rates in the table are the normalized ones as defined in Eq. (1). The numbers of roots and compound words in each lexicon are also listed. All decomposed lexicons have around 60K to 64K *words*. The system speed and language model sizes are similar to those in the 64K baseline.

Algorithm	#roots	#cpnds	OOV	Unadapt	Adapt
I	60k	0k	0.79	31.8	-
II	40k	24k	1.67	26.3	19.8
II	50k	14k	1.30	25.6	19.6
III	30k	34k	2.57	23.8	17.9
IV	40k	24k	2.95	23.7	17.8
IV	50k	14k	2.36	24.0	18.0
IV	63k	1k	1.77	25.4	-

Table 3: Comparison of four decomposition algorithms (cpnds: compound words)

As we can see, although Algorithm I achieved the lowest OOV rate, which is close to what we usually have in an English system, it provided the worst recognition performance. Since there was only one constraint applied in this algorithm, it is believed that the loss in the recognition accuracy is due to the aggressive splitting which resulted in too many possible pronunciations for each word (especially some suffixes) during acoustic training. The acoustic confusability was largely increased. Also, the language model was weak when trained on the fully decomposed words.

Two configurations were experimented for Algorithm II, with either 40K or 50K roots in the recognition lexicon. The re-compounded words were selected based on the N-gram frequencies as described earlier. Both configurations resulted in lower OOV rates but small loss in recognition accuracy when compared to the 64K baseline. During investigation, we observed a large degradation in the forward pass. So the bigram language model was still weak although we had 14K or 24K compound words in the lexicon. Note that these compound words were not necessarily original words. For example, it could contain a suffix followed by a prefix of the next word.

In Algorithm III, all the selected frequent N-grams must be found in the original lexicon after being converted into a compound word. In this way, it ended up with 30K roots and 34K compound words. With an OOV rate 39% lower than that of the 64K baseline, it achieved 1.6% absolute gain after adapted decoding. So keeping frequent original compound words is important. The risk of increasing acoustic confusability was reduced, and also the language model was stronger than those obtained from Algorithm I and II.

Three systems were built with Algorithm IV. The original lexicons contain 95K, 120K and 165K words, respectively. Different numbers of most frequent compound words were retained in the recognition lexicon. The system with 40K roots and the top 24K

compound words achieved the best result. A 1.7% absolute WER reduction (8.7% relative) was achieved compared to the 64K baseline. This recognition performance is also comparable to that in the 200K normal system. Meanwhile, there is no significant increase in computation and memory usage. Based on all the results, it is believed that the four constraints we applied during the word splitting made major contributions to the significant improvement in recognition accuracy.

7. Conclusion

We have compared four morphological decomposition algorithms in our Arabic broadcast news transcription system. The best algorithm resulted in 1.7% absolute WER reduction when compared to the 64K baseline. The recognition performance is also comparable to that of a 200K-word system. In the meantime, there is no need for extra computation and memory usage, which makes the decomposition system quite appealing. It is believed that the system can be improved further in the future. An obvious way is to go beyond the limit of 64K words in the decomposition system so that more roots and frequent compound words can be included in order to lower the WER further. Different set of affixes and constraints can also be explored next.

References

1. P. Geutner, "Using morphology towards better large-vocabulary speech recognition systems," *Proc. ICASSP*, pp. 445-448, 1995.

2. A. Berton, P. Fetter and P. Regel-Brietzmann, "Compound words in large-vocabulary German speech recognition systems," *Proc. ICSLP*, 1996.

3. R. Ordelman, A. Hessen and F. Jong, "Compound decomposition in Dutch large vocabulary speech recognition," *Proc. Eurospeech*, 2003.

4. M. Larson, D. Willett, J. Kohler and G. Rigoll, "Compound splitting and lexical unit recombination for improved performance of a speech recognition system for German parliamentary speeches," *Proc. ICSLP*, 2000.

5. K. Darwish, "Building a shallow Arabic morphological analyzer in one day," *Proc. ACL workshop on computational approaches to semitic languages*, 2002.

6. K. Kirchhoff, et al., "Novel approaches to Arabic speech recognition: report from the 2002 Johns-Hopkins summer workshop," *Proc. ICASSP*, pp. 344-347, 2003.

7. D. Vergyri, K. Kirchhoff, R. Gadde, A. Stolcke and J. Zheng, "Development of a conversational telephone speech recognition for Levantine Arabic," *Proc. Eurospeech*, pp. 1613-1616, 2005.

8. D. Vergyri, K. Kirchhoff, K. Duh and A. Stolcke, "Morphology-based language modeling for Arabic speech recognition," *Proc. ICSLP*, 2004.

9. L. Nguyen and B. Xiang, "Light Supervision in Acoustic Model Training," *Proc. ICASSP*, Montreal, May 2004.

10. M. Afify, L. Nguyen, B. Xiang, S. Abdou and J. Makhoul, "Recent progress in Arabic broadcast news transcription at BBN," *Proc. Eurospeech*, 2005.

11. A. Ghaoui, F. Yvon, C. Mokbel and G. Chollet, "On the use of morphological constraints in N-gram statistical language model," *Proc. Eurospeech*, 2005.