SUBWORD-BASED MODELING FOR HANDLING OOV WORDS IN KEYWORD SPOTTING

Yanzhang He,[1] *Brian Hutchinson,*[2] *Peter Baumann,*[3]
Mari Ostendorf,[4] *Eric Fosler-Lussier,*[1] *Janet Pierrehumbert*[3]

[1]Dept. of Computer Science & Engineering, The Ohio State University, Columbus, OH, USA
[2]Dept. of Computer Science, Western Washington University, Bellingham, WA, USA
[3]Dept. of Linguistics, Northwestern University, Evanston, IL, USA
[4]Dept. of Electrical Engineering, University of Washington, Seattle, WA, USA

ABSTRACT

This work compares ASR decoding at different subword levels crossed with alternative keyword search strategies to handle the OOV issue for keyword spotting in the low-resource setting. We show that a morpheme-based subword modeling approach is effective in recovering OOV keywords within a Turkish low-resource keyword spotting task, where mixed word and morpheme decoding approach outperforms the traditional subword-based search from word-decoded lattices that are broken down to subword lattices. Furthermore, unsupervised learning of morphology works almost as well as a rule-based system designed for the language despite the low-resource condition. A staged keyword search strategy benefits from both methods of morphological analysis.

Index Terms— Automatic Speech Recognition, Keyword Spotting, Morphology

1. INTRODUCTION

Vocabulary growth is an important issue for automatic speech recognition, resulting in the twin problems of sparse language model training data and out-of-vocabulary (OOV) words, i.e., words that appear in the test data but are not seen in the training set and thus not represented in the recognizer vocabulary. These problems are particularly pronounced in highly inflective and aggutinative languages, but they can pose challenges for any language in the low-resource setting.

There are three types of applications that tend to have somewhat different approaches to handling OOVs, though all typically involve the use of sub-lexical or subword items in the recognizer vocabulary. For open vocabulary word transcription, subword items are chosen and represented in such a way that orthographic forms can be recovered from the sequence of recognized subwords. In human-computer interaction and voice search, subwords are leveraged to facilitate detection of OOVs. In keyword spotting or spoken term detection, subwords are used to handle search terms that are OOV. Particularly in the open vocabulary recognition and keyword spotting settings, the use of subwords can also help address the data sparsity problem in language model training. In this work, the focus is on handling OOVs in keyword spotting, but it is informed by work on open vocabulary recognition.

A variety of methods have been used for deriving subwords, which can be broadly classed as being based on phones or phone n-grams, graphones, syllables, and morphologically based units (possibly including bundles of morphemes) that we will refer to as "morphs." Graphones (coupled phonetic and orthographic sequences) [1] and morphs are particularly well suited to open vocabulary recognition. While some work has based the vocabulary entirely on morphs (see [2, 3, 4] and references therein), other studies obtain better results using a combination of morphs and words in Arabic [5] and German [6]. However, a mixed word and syllable vocabulary outperformed a mixed word and morph vocabulary for Polish [7]. A mixed word and graphone vocabulary has also been explored for English [8]. Morphs have the potential advantage of introducing more powerful constraints in language modeling, and several studies have investigated novel language model structures that take advantage of morphological features in a variety of languages [9, 6, 4, 7, 10, 2, 11, 12]. While these studies motivate our use of morphs in this work, only standard n-grams are used here since our focus will be primarily on the keyword search strategies that take advantage of a mixed word and morph vocabulary.

In keyword spotting, a standard approach for handling OOVs is to transform a word lattice into a phone lattice when searching for keywords [13]. Directly indexing the output of phone recognition tends to lead to much worse results, but in [14], it is shown that decoding with phone n-gram units outperforms the word lattice transformation approach for OOV

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense US Army Research Laboratory contract number W911NF-12-C-0014. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government.

terms when a flexible segmentation is used to incorporate different order n-grams. Since in-vocabulary terms are best recognized with a word-based model, a staged keyword search strategy (word-based model for in-vocabulary terms, subword model for OOVs) is typically used.

In our study on keyword spotting, we investigate the use of mixed unit decoding to improve keyword spotting for OOVs, particularly focusing on the use of morph-based subword units. This is in contrast to previous study of morph-based keyword spotting [15], where they expand word-decoded lattices into morphs or do subword-only decoding instead of doing mixed unit decoding. We compare two methods for obtaining morph units in morph-only and mixed word-morph decoding, to a word-based decoding baseline, and we leverage different keyword search alternatives. Specifically, we look at stem and affix bundles identified by a rule-based system designed for our target language (Turkish) [16] and automatically-derived morphs identified via unsupervised learning using Morfessor [17].

This contrast is similar to some of the methods explored in [2] for open vocabulary recognition, and we confirm their finding that unsupervised morphology learning gives similar results compared to the rule-based system. Further, unlike this and other prior work which uses morphology in Turkish broadcast news transcription [18, 4, 19, 2], our study involves keyword spotting in conversational Turkish with minimal training resources (10 vs. roughly 200 hours).

2. SUBWORD-BASED DECODING

2.1. Morphological Analysis

There are many approaches for splitting words into subword units or morphs; of particular interest are rule-based morphological analyzers and unsupervised segmentation algorithms. Rule-based systems typically provide linguistically accurate morphological analyses for all words covered in the hand-crafted rule base. However, this high accuracy is often achieved by positing a large number of morphs; since many of these morphs are infrequent or acoustically confusable, they may have a detrimental effect on speech recognition performance [20].

Unsupervised segmentation algorithms are less accurate than rule-based systems, but they typically do not posit many rare morphs, while still providing good coverage of new words. However, many of the identified morphs may still be acoustically confusable.

In this paper, we compare analyses obtained from two different morphological systems, leveraging their respective advantages, while minimizing their drawbacks. For the rule-based system, we used the freely available finite-state morphological analyzer *TRmorph* [16], which achieves high accuracy, while also covering a large portion of the Turkish lexicon. In our training set, only 5.5% of word types could not be analyzed. In order to avoid the problem of over-segmentation and high acoustic confusability, we did not use *TRmorph*'s full morphological analysis, but only segment the word into a stem and the remaining affix bundle (*S+AB*). In addition to providing larger morphs, this stem and affix bundle segmentation also reduces the problem of morphological ambiguity: for 78.6% of the analyzed words the stem was uniquely determined, while unambiguous full morphological analyses would have been possible for only 13.9% of the analyzed words. In the remaining cases of ambiguity, the stem with the highest frequency across the whole training set was chosen.

Our unsupervised segmentation analysis comes from the *Morfessor* algorithm [17], which has become a benchmark for morphological segmentation. Morfessor's selection of word-internal segmentation is based on the minimum-description-length principle: it tries to find a lexicon of morphs that is both accurate and minimal. The desired degree of segmentation can be manipulated via Morfessor's perplexity threshold parameter, but the effect of this parameter depends strongly on the morphological structure of the language and the size of the training set. We used an exhaustive search over all possible parameter values, minimizing the percentage of low-frequency morphs.

2.2. Subword-based Vocabulary and Language Model

Mixed-unit vocabularies and language models are trained by considering multiple segmentations of the training text: the original word segmentation, a version with all words expanded into subword units, and one with some of the words expanded. Our vocabulary is simply the union of the units present in any segmentation of the training data. With this vocabulary we train trigrams on each of the segmentations of the training data and interpolate them to obtain the final language model. Tuning the interpolation weights requires a segmentation of the held out development data, but no fixed segmentation is clearly preferred, so we set the interpolation weights to be uniform. All of our language models are trigrams with modified-Kneser-Ney smoothing, and are trained with the SRILM toolkit [21]. In our preliminary experiment results, higher order n-grams beyond trigrams did not lead to reduced perplexity.

For the partially expanded version, we choose a set of words to decompose that satisfy three tunable criteria. First, all words that appear more than θ_1 times in the training data are left intact (i.e. excluded from expansion set). Leaving frequent words intact increases the effective context for tokens with a frequent word in their n-gram history, and due to their frequency, we assume that these are the whole word units that are easiest to model. Second, no word will be in the expansion set if any subword would appear fewer than θ_2 times in the expanded text. Lastly, no word will be in the expansion set if any subword appears in fewer than θ_3 expanded word types in the expanded text. The last two criteria are designed to limit unit expansion: a subword unit that appears few times in the expanded text will be difficult to model, while a subword unit that appears in few distinct types does not add much generalization. A simple iterative algorithm finds the set of words that satisfy all three criteria. The language models used for the mixed-unit decoding experiments reported in this paper are a three-way interpolation of the word-based, partially-expanded and fully-expanded models. For the partially-expanded models, we chose thresholds $\theta_2 = \theta_3 = 5$ to avoid introducing very infrequent morpheme units, and $\theta_1 = 500$ so that roughly half of the word tokens were left intact.

2.3. Pronunciation Modeling for Subword Units

Pronunciations for subwords are needed both for decoding within the ASR system and for the keyword search strategy. Since our subwords are generated from the lexicon, we have pronunciations for all of the source words, so the primary task is to associate parts of word pronunciations with the subwords. We have considered two approaches to this problem: the first is to train a grapheme-to-phoneme system to predict pronunciations. We follow a joint multigram approach utilized by the Phonetisaurus G2P toolkit [22]. Predicting subword pronunciations from G2P does have the disadvantage that the context of the subword within the word is lost – for example, ‘s’ word-finally in English can often be pronounced /z/ but this is rarely the case word initially.

In order to preserve context, we also investigated a technique where the graphone alignments in the joint multigram were mapped to the subword decompositions of each word, thus providing a range of pronunciations for each subword. We then took the most likely pronunciation of each subword and used that as the pronunciation in the subword lexicon.

3. KEYWORD SEARCH USING SUBWORDS

Searching keywords in the form of subwords is essential to recover OOVs. Each word can be represented by the word itself or a subword sequence if it can be segmented. For simplicity, we only consider one possible segmentation for each word. In a mixed-unit decoded system, we will search both the word and the subword sequence from the index. For multiword keywords, their representation would be the cross product of all the representations of each component word. In addition, we can consider using only the stem as another representation of a keyword. If the stem is also rare in the corpus, matching the stem is likely to match the OOV keyword, thus reducing the miss rates. Once the keyword representation is chosen, searching the unit sequence in the mixed-unit index is just like searching the multiword keywords in the word-based index, which is described below. Currently each representation for a keyword is equally weighted for simplicity.

Our keyword search algorithm is similar to that of [23]. We create a word-based index from the lattices, tracking all of the words that occur in the lattice, their start and end times, and their lattice posterior probabilities. For single word keywords, we return the list of all of the keyword occurrences, sorted by their posterior probabilities. For multiword keywords, we retrieve the individual words from the index in the correct order with respect to their start and end times but discard occurrences where the time gap between adjacent words is more than 0.5 seconds. All the hypotheses of a keyword form a posting list. The detection threshold in the list is determined separately for each keyword using an empirical estimate of each keyword’s term weighted value (TWV) [24]. The probabilities in each keyword’s posting list are adjusted by a keyword specific offset to enable a single, keyword independent, detection threshold.

4. EXPERIMENTS

4.1. Evaluation Setup

We evaluate the effectiveness of various strategies for subword modeling in the task of OOV handling in keyword spotting (KWS). We conduct ASR and KWS experiments on systems trained with the 10-hour limited language pack (LimitedLP) of the Turkish IARPA Babel conversational telephone speech data (IARPA-babel105b-v0.4, [25]). “Actual term weighted value” (ATWV) [26] is the primary metric for the Babel program on the keyword spotting task. In this metric, the cost of a false alarm is relatively small and almost the same for each keyword, but the cost of a miss is one over the number of occurrences of the keyword, which is especially high for OOVs. We tune our parameters on the 10-hour development test set using the evaluation keyword set, and evaluate ATWV on the 5-hour eval-part1 test set for the same set of keywords. In all our experiments, we search only for OOV keywords (KW) using the subword-based system, while the in-vocabulary keywords are still searched by the word-based system for better performance. In addition to the overall ATWV, we also report the OOV-conditioned ATWV, where TWV for OOV keywords are averaged only within the OOV set, so that the scale is independent of OOV rates across different keyword sets. For the evaluation keyword set, 1685 keywords exist in the dev set data, 387 of which are OOVs (22.9%); 1625 keywords exist in the eval-part1 set, 452 of which are OOVs (27.8%). Around 60% of the OOV keywords can be fully recovered by the *S+AB* morphs in the training vocabulary; around 95% of the OOV keywords can be fully recovered by the Morfessor morphs.

Our ASR system models speech using a conventional cross-word triphone 3-state HMM system [24]. Observations are modeled by diagonal covariance GMMs. For the limited language pack, we build a relatively compact model using 12 mixtures in the GMMs and 1000 tied triphone states. The

Vocab / LM	Lattice Expansion	KW Representation	OOV ATWV	Overall ATWV	%OOV in Posting List
Word	-	Word	0	0.164	0%
Word	Phone	Phone	0.017	0.168	31.0%
Word	Morph$_{\mathrm{Morfessor}}$	Morph$_{\mathrm{Morfessor}}$	0.009	0.167	21.8%
Word	Morphs$_{\mathrm{S+AB}}$	Morphs$_{\mathrm{S+AB}}$	0.014	0.168	13.1%
Word	Morphs$_{\mathrm{S+AB}}$	Stem	0.021	0.169	34.7%
W+M$_{\mathrm{S+AB}}$	-	W+M$_{\mathrm{S+AB}}$	0.043	0.174	17.8%

Table 1. Word-decoding vs. mixed-unit-decoding using *regular* lattices on the *dev* set.

features include 13-d MFCC warped with speaker-dependent VTLN, pitch features, their deltas and accelerations, and bottleneck neural network features. The baseline language model is trigram with modified-Kneser-Ney smoothing and pruning. The recognition is generated by a second-pass decoder with speaker adaptation. The WER of the word-decoded system on the development test set is 74.6%.

4.2. Results and Discussion

The word-decoded baseline system achieves 0.164 overall ATWV, but does not handle OOVs. Following traditional subword-based methods, we decomposed word arcs into subword arcs of varying units (Table 1). Interestingly, converting word arcs into either phones or morphs provides a similar level of performance, although the phone-based KWS takes much longer to search. Unsupervised Morfessor morphs (Morph$_{\mathrm{Morfessor}}$) lead to slightly worse OOV ATWV than affix bundled morphs (Morphs$_{\mathrm{S+AB}}$), although the former almost doubles the OOV coverage in the posting list. Using only stems instead of full morph sequences (Morphs$_{\mathrm{S+AB}}$) as the keyword representation almost triples the OOV coverage in the posting list and leads to slightly better OOV ATWV. When we decode with the mixed-unit vocabulary and language model (W+M$_{\mathrm{S+AB}}$), performance outstrips all of the search methods in the word-decoded system. This is because we have more robust estimation of the units and thus better posteriors, even though the OOV coverage in the posting list is only half of the stem-based approach. Increasing the lattice density also helps especially for OOVs (Table 2).

Comparing different subword units for vocabulary design in Table 2, we find that mixed word-and-morph decoding performs better than mixed word-and-phone decoding as the phone unit estimate is not so robust. The performance with unsupervised morph units is not far from that with morph units learned from hand-crafted rules. This is partly because the smaller morph units have higher oracle coverage of OOV KWs (95% vs. 60%). Interestingly, staging of both morph systems does provide a small improvement over the individual systems (Table 3), as the larger units provide better keyword posterior estimates, while the smaller units complement with better coverage.

Vocab / LM / KW	OOV	Overall	%OOV in Posting List
W+M$_{\mathrm{S+AB}}$	0.053	0.177	27.9%
W+M$_{\mathrm{Morfessor}}$	0.046	0.175	42.6%
W+Phone	0.017	0.168	57.5%

Table 2. OOV/Overall ATWV with different size of subword units using *dense* lattices on the *dev* test set.

System	Dev Set		Eval-part1 Set	
	OOV	Overall	OOV	Overall
(1) Word	0	0.164	0	0.163
(2) W+M$_{\mathrm{S+AB}}$	0.053	0.177	0.074	0.184
(3) W+M$_{\mathrm{Morfessor}}$	0.046	0.175	0.062	0.181
Staged (1-2-3)	0.061	0.178	0.084	0.187

Table 3. ATWV on *dev* and *eval-part1* test sets with *dense* lattices. Note: (2) and (3), as in all above experiments, use the word system (1) as a first stage.

The results on the dev and eval-part1 test sets are shown in Table 3. The word-and-bundled-morph decoded system has achieved more than 2% absolute gain in ATWV for eval-part1 over the original word-based system. In a subsequent run with improved acoustic models, we noted an improvement in our staged system (0.196 ATWV) over a word baseline (0.170).

5. CONCLUSIONS

We have shown that morph-based subword modeling is useful for handling the OOV issue for keyword spotting in the low-resource setting for a morphologically rich language, and that including subwords in the decoding process can be more effective than the traditional method of breaking down word-decoded lattices into subword lattices. In this work, a mixed-level vocabulary design and the staged keyword search strategy are used to balance the confusability and coverage. Future work may benefit from more sophisticated morphological feature-based approaches in language modeling which would allow us to better model long-span subword dependencies, and better rescoring of putative subword hits in keyword posting lists. In addition, we can also apply our subword modeling approach to search in-vocabulary words as a complement to the word-based search approach.

Acknowledgements: We thank Steven Wegmann for very useful discussions and his help on the KWS system. We also thank our collaborators at ICSI providing us the ASR system.

6. REFERENCES

[1] M. Bisani and H. Ney, "Open vocabulary speech recognition with flat hybrid models," in *Proc. Interspeech*, 2005, pp. 726–729.

[2] H. Sak, M. Saraclar, and T. Güngör, "Morpholexical and discriminative language models for Turkish automatic speech recognition," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 20, no. 8, pp. 2341–2351, 2012.

[3] T. Hirsimäki, J. Pylkkonen, and M. Kurimo, "Importance of high-order n-gram models in morph-based speech recognition," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 724–732, 2009.

[4] E. Arisoy, M. Saraclar, B. Roark, and I. Shafran, "Syntactic and sub-lexical features for Turkish discriminative language models," in *Proc. Interspeech*, 2010, pp. 5538–5541.

[5] A. El-Desoky, C. Gollan, D. Rybach, R. Schluter, and H. Ney, "Investigating the use of morphological decomposition and diacritization for improving Arabic LVCSR," in *Proc. Interspeech*, 2009, pp. 2679–2682.

[6] A.E.-D. Mousa, M.A.B. Shaik, R. Schluter, and H. Ney, "Sublexical language models for German LVCSR," in *Proc. IEEE Spoken Language Technology Workshop*, 2010, pp. 171–176.

[7] M.A.B. Shaik, A.E.-D. Mousa, R. Schluter, and H. Ney, "Using morpheme and syllable based sub-words for Polish LVCSR," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2011, pp. 4680–4683.

[8] M.A.B. Shaik, D. Rybach, S. Hahn, R. Schluter, and H. Ney, "Hierarchical hybrid language models for open vocabulary continuous speech recognition using WFST," in *Proc. Workshop on Statistical and Perceptual Audition*, 2012, pp. 46–51.

[9] K. Kirchhoff, D. Vergyri, J. Bilmes, K. Duh, and A. Stolcke, "Morphology-based language modeling for conversational Arabic speech recognition," *Computer Speech and Language*, vol. 20, no. 4, pp. 589–608, 2006.

[10] A.E.-D. Mousa, R. Schluter, and H. Ney, "Investigations on the use of morpheme level features in language models for Arabic LVCSR," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2012, pp. 5021–5024.

[11] A.E.-D. Mousa, H.-K. J. Kuo, L. Mangu, and H. Soltau, "Morpheme-based feature-rich language models using deep neural networks for LVCSR of Egyptian Arabic," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 8435–8439.

[12] E.W.D. Whittaker and P.C. Woodland, "Particle-based language modelling," in *Proc. Interspeech*, 2000, pp. 170–173.

[13] M. Saraclar and R. Sproat, "Lattice-based search for spoken utterance retrieval," in *HLT-NAACL 2004: Main Proceedings*, D. Marcu S. Dumais and S. Roukos, Eds., Boston, Massachusetts, USA, May 2 - May 7 2004, pp. 129–136, Association for Computational Linguistics.

[14] I. Bulyko, J. Herrero, C. Mihelich, and O. Kimball, "Subword speech recognition for detection of unseen words," in *Proc. Interspeech*, 2012.

[15] Siddika Parlak and Murat Saraclar, "Performance analysis and improvement of Turkish broadcast news retrieval," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 731–741, 2012.

[16] Ç. Çöltekin, "A freely available morphological analyzer for Turkish," in *Proc. International Conference on Language Resources and Evaluation*, Valletta, Malta, May 2010.

[17] M. Creutz and K. Lagus, "Inducing the morphological lexicon of a natural language from unannotated text," in *Proc. International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05)*, Espoo, Finland, June 2005.

[18] E. Arisoy and M. Saraclar, "Lattice extension and vocabulary adaptation for Turkish LVCSR," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 1, pp. 163–173, 2009.

[19] H. Sak, M. Saraclar, and T. Güngör, "Morphology-based and sub-word language modeling for Turkish speech recognition," in *Proc. IEEE International Conference on Acoustics Speech and Signal Processing*, 2010, pp. 5402–5405.

[20] K. Carki, P. Geutner, and T. Schultz, "Turkish LVCSR: towards better speech recognition for agglutinative languages," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing,*. IEEE, 2000, vol. 3, pp. 1563–1566.

[21] A. Stolcke, "SRILM — an extensible language modeling toolkit," in *Proc. Int'l Conf. on Spoken Language Processing*, Denver, Colorado, 2002.

[22] J. Novak, N. Minematsu, and K. Hirose, "WFST-based grapheme-to-phoneme conversion: Open source tools for alignment, model-building and decoding," in *Proc. International Workshop on Finite State Methods and Natural Language Processing*, Donostia–San Sebastian, 2012, pp. 45–49.

[23] D. Miller, M. Kleber, C.-L. Kao, O. Kimball, T. Colthurst, S. Lowe, R. Schwartz, and H. Gish, "Rapid and accurate spoken term detection," in *Proc. Interspeech*, 2007, pp. 314–317.

[24] S. Wegmann, A. Faria, A. Janin, K. Riedhammer, and N. Morgan, "The tao of ATWV: Probing the mysteries of keyword search performance," in *Proc. IEEE Workshop on Speech Recognition and Understanding*, 2013.

[25] "IARPA Babel Program - Broad Agency Announcement (BAA)," http://www.iarpa.gov/Programs/ia/Babel/solicitation_babel.html, 2011.

[26] J. G. Fiscus, J. Ajot, J. S. Garofolo, and G. Doddingtion, "Results of the 2006 spoken term detection evaluation," in *Proc. ACM SIGIR Workshop on Searching Spontaneous Conversational*. Citeseer, 2007, pp. 51–55.