Subword and Phonetic Search for Detecting Out-of-Vocabulary Keywords

Damianos Karakos, Richard Schwartz

Raytheon BBN Technologies, Cambridge, MA

{dkarakos,schwartz}@bbn.com

Abstract

We compare several approaches, separately and together, for spotting of out-of-vocabulary (OOV) keywords, in terms of their ATWV scores. We considered three types of recognition units (whole words, syllables, and subwords of different lengths) and two basic search strategies (whole-unit, fuzzy phonetic search). In all cases, the search was performed by collapsing the recognition lattice into a consensus network, either in terms of the recognized whole units, or by first splitting the recognized units into phonemes. We ran experiments on five languages, for which the language model and vocabulary were derived from only 10 hours of transcriptions (70k-100k words of text), resulting in keyword OOV rates varying from 10% to 63% on new data, depending on the language. Our conclusions were that: 1) In all cases, the fuzzy phonetic search on phoneme-split lattices is better than searching for the whole units, 2) The syllable units are the best of the subword units for OOV keyword detection using fuzzy phonetic search, and 3) These methods combine very well, sometimes resulting in ATWV scores for OOV terms which are not too far below those of IV terms.

Index Terms: keyword spotting, speech recognition with subwords, limited language resources

1. Introduction

We consider the keyword spotting scenario in which the audio corpus is processed to produce an index and where the keyword spotting must be performed exclusively using the precomputed index.

The keyword terms are short phrases of one or more words. The lowest Word Error Rate (WER) of the 1-best answer is obtained by recognizing the speech with the known vocabulary, and we take it for granted that we need a word-based recognizer and indexer for spotting those keywords that consist entirely of known words. But of course, if the index is produced only in terms of whole words, then any keyword that contains an OOV word (a word that was not in the recognition lexicon) cannot be found. Therefore, there are several methods that have been proposed for finding keywords that contain OOV words. We assume that, if we are given a new word, we can use an automatic Grapheme-to-Phoneme (G2P) procedure to produce a plausible phonetic pronunciation for the new words. So if we could know the phonetic makeup of the audio data, we could search for the new words in terms of their phonemes. Some have proposed that the audio should be recognized using a phoneme recognizer instead of a word recognizer [1]. But even though this allows for recognition of any word, in principle, the recognition accuracy of phonetic recognizers is much worse, by comparison, for the known words and also quite poor for detection of OOV words. Therefore, we must either resort to (i) hybrid fuzzy phonetic search strategies, in which we perform recognition in terms of words, but we search the recognition output (lattice) for the phonetic strings, allowing for inexact matches [2, 3, 4], or, (ii) recognition in terms of shorter units [5, 6, 7, 8, 9] that have a higher chance of allowing a new word. The latter approach is the main focus of this paper.

The first strategy is to recognize in terms of whole words to produce a recognition lattice, compute posterior probabilities on all of the recognized arcs, split the recognized words into their corresponding phonemes according to the phonetic dictionary, and search the phonetic lattice for strings corresponding to the new word, allowing for an inexact match (substitutions, insertions, deletions) as in [10]. We can generalize the word lattice further by creating a consensus network (cnet) [11] from the phonetic lattice. This allows us to mix and match phonemes from different words in the lattice, thus resulting in higher recall and greater ability to distinguish correct from incorrect keywords [12].

The second strategy is to use shorter subwords for the recognition units, allowing for greater flexibility in the recognition. This tackles the problem of high phoneme error (PER) that arises when whole-word recognition is applied in regions corresponding to new words. Indeed, despite the fuzzy matching, the fact that the whole-word recognition is constrained to a (possibly limited) set of predefined words, makes it difficult to detect keywords whose phonetic representation is quite different from the phonetic representation of any sequence of vocabulary words. One method for choosing the set of subwords is to pick the set of a given size that provides the best coverage of the observed vocabulary, according to some criterion [13, 14]. A simple version of this is to pick the most frequent N (e.g., 4000) phonetic strings that occur in the training corpus [2]. But it has been shown that we get better performance by simply defining subwords as all of the phoneme substrings up to some length (e.g., from 1 to 3 phones) that occur in the training set [2]. These subword models provide lower PER in the regions corresponding to new words, thus resulting in better KWS performance.

One reasonable choice of subword for tackling the problem of OOV keywords is linguistically-motivated morphemes of the language. Morphemes represent the basic meaning-bearing units of a language and we know that segmenting words in terms of a good morphology can decrease the OOV rate significantly. We performed a study [8] in which we used a high-quality supervised morphology system to provide alternative segmentations for Turkish speech. As we found out, although we were able to represent many of the OOV keywords in terms of such morphemes, there were still a significant number of keywords (which consisted of "new" morphemes) that could not be segmented. This is, of course, a consequence of sparsity. If, instead, the morphemes were broken down into smaller pieces

(without necessarily adhering to linguistic rules) the generalization would have been better. This conclusion motivates us to consider alternatives (similar to those of [2]) which are able to represent virtually any possible OOV word, without necessarily going to the extreme of using a phonetic recognizer.

In this paper, we take the work of [2] a bit further by considering different lengths of phoneme substrings (consisting of at most 3, 4, 5 and "unlimited" number of phones) and then combining them together. We also consider *syllables* as an alternative form of (supervised) subword units that provide high coverage of the language with a small number of units. Performance is evaluated using the Average Term-Weighted Value (ATWV) metric [15], which is defined as a weighted combination of the miss and false alarm rates, averaged over all keywords:

$$\text{ATWV} = 1 - \frac{1}{K}\sum_{k=1}^{K}\left(\frac{\#\text{miss}(k)}{\#\text{ref}(k)} + \beta\frac{\#\text{fa}(k)}{T - \#\text{ref}(k)}\right),$$

where k is an index over the keywords, $\#\text{miss}(k)$ is the number of reference tokens of keyword k that are not detected, $\#\text{fa}(k)$ is the number of false detections of k, $\#\text{ref}(k)$ is the number of reference tokens of keyword k, T is the total duration of the audio in seconds and $\beta = 999.9$ is a constant.

In Section 2, we describe the various subwords and the whole-unit keyword spotting method. In Section 3, we discuss how we perform phonetic search from each of the subword types. In Section 4, we briefly review the score normalization and system combination method that we used to combine the results across multiple recognition and search methods. In Section 5, we present results on five languages, and, finally, we present concluding remarks in Section 6.

2. Subword Units

We discuss two types of subword units: syllables and subwords of 1-N phonemes.

I. Syllables
The phonetic spellings that we are provided with contain syllable boundaries. Note that the syllables are expressed in terms of sequences of phonemes—not letters. This allows us to represent each word in the training lexicon as a sequence of syllables, which are, in turn, represented as sequences of phonemes. For example, the English word TABLE with the phonetic spelling T-EY-#-B-AX-L (where '#' represents the syllable boundary) would now be represented as a compound word "T-EY_B-AX-L" where "_" signifies a compound word. In addition to the original words (represented as syllable compounds), the recognition lexicon is augmented with all syllables and all possible syllable compounds found within any word.

The acoustic model of context-dependent within-word and cross-word models was the same one that was estimated from the usual whole-word transcripts. To estimate the language model of syllables and compounds, we first segment the audio transcripts into space-separated syllables, e.g., "table" becomes "T-EY B-AX-L". When we estimate the language model from the transcripts, we provide the compound-syllable lexicon. The language model estimation program [2] constructs a lattice of all possible lexical sequences of single and compounded syllables and then estimates an exhaustive set of language model probabilities. Recognition is subsequently run, using this expanded set of compounded syllables in the lexicon. (Note that this is different from the use of syllables in [12], where only whole-word recognition was done, and syllables were used only for generating confusion networks of a finer granularity.) After recognition, we have a lattice of recognized units (single and compound syllables). We compute posterior probabilities on the lattice, and then we split any arc that contains multiple syllables into single syllables, preserving the same posterior probability. Then, we create a confusion network of recognized syllable units in the usual way.

Given a search term (known or new), we could use only the correct segmentation into syllables. However, we divide it into all possible sequences of syllables, given the predicted phonetic string of the keyword. Then we search the confusion network for all of these possibilities, requiring an exact match. There is no "proper" method for combining the posteriors of successive links in the confusion network to produce the posterior probability of a sequence of links. So we tried using both the product and the geometric mean of the posteriors to produce "scores" which were used subsequently in downstream processing. Furthermore, similar to [2], we also performed fuzzy phonetic search on *phonetic* confusion networks that were derived after expanding the syllable-based lattices using phonetic pronunciations.

II. 1-N Phone Subwords
Following [2], we enumerated all possible phone sequences that occurred within words that appeared in the training transcripts. We ran experiments where N was set to 3, 4, or 5, as well as "unlimited" (i.e., all phonetic sequences within a word, including the whole word itself). Obviously, the number of such units increases rapidly with the value of N. Table 1 shows the size (and average phonetic length) of the vocabulary for N=3,4,5 and "unlimited" for the five development languages used in the second year of the IARPA Babel program. The process was somewhat different than for syllables, and followed that of [2]. We segmented the language model training into single phonemes and provided a list of subwords, represented as compound phonemes to the language model program so that it could estimate all possible n-gram probabilities among subwords. The lexicon contained only the complete set of single subwords. After recognition, we computed posteriors on the subword lattice and collapsed them into a confusion network of subwords. During search, as with syllables, we determined all possible subword sequences for each keyword and searched for all possibilities in the confusion network. And, as with syllables, we also performed fuzzy phonetic search on phonetic confusion networks.

3. Fuzzy Phonetic Search

This process has been described in [2], so we only review it briefly. Given the lattice of recognized units of any kind (words, syllables, subwords), after computing posteriors, we split the units into the corresponding phone sequences and then accumulate the phone arcs into a phonetic confusion network. Given a keyword, we search the confusion network for a close match, with penalties for substitutions, deletions, and insertions. Again, we used the product or geometric mean of successive posteriors to produce a total score for a keyword. Alternatively, we can search the lattices directly, as done in [16, 17]; however, as we have observed in our experiments, searching the confusion networks results in much better performance, possibly because of the introduction of new connections between words which occur close in time. A similar observation was made in [18].

Number of basic recognition units					
	As	Be	Ha	La	Zu
word	8.7K	9.4K	5.4K	4.0K	13.9K
syllable	2.0K	2.4K	2.2K	2.6K	1.7K
1-3ph	5.8K	7.2K	4.5K	9.1K	7.1K
1-4ph	16.7K	20.4K	11.1K	16.9K	26.0K
1-5ph	29.5K	34.0K	16.4K	23.6K	57.3K
1-∞ph	55.7K	56.3K	22.8K	35.4K	190.6K
Average phonetic length					
	As	Be	Ha	La	Zu
word	6.22	5.79	5.11	4.79	7.97
syllable	3.02	3.05	3.14	2.76	3.17
1-3ph	2.85	2.86	2.86	2.75	2.88
1-4ph	3.60	3.60	3.54	3.34	3.70
1-5ph	4.21	4.16	4.02	3.81	4.41
1-∞ph	5.65	5.35	4.83	5.02	6.86

Table 1: Number of basic recognition units, and their average phonetic length, for each of the five languages of the second year of the IARPA Babel program. Only single units, without compounding, are shown for syllables.

As	Be	Ha	La	Zu
60.8	63.1	51.3	52.7	69.0

Table 2: WERs of whole-word decodes.

4. Score Normalization and System Combination

We have previously described different score normalization methods [19, 20]. The ATWV metric [15] requires that scores of different keywords are commensurate. Therefore it is important that they be normalized appropriately. Here, we used the keyword-specific thresholds (with exponential normalization) method for normalization, as it was the fastest. It is instructive to compare the performance of each of the recognition and search methods. But it is also interesting to see whether they find the same information or their results are complementary. So we fused the results of the individual search systems to produce a combined result using Powell's method [21] for maximizing ATWV, as described in [19]. Note that, due to the quite diverse effect that the different search methods have on the IV and OOV keywords, we perform combinations separately on the two sets.

5. Experimental Results

The system we used was a fairly complex state-of-the-art training and recognition system that embodied a combination of several technologies [22]. The features were derived at Brno University of Technology [23], using deep multi-layer perceptrons (MLPs). These were trained on 10 hours of transcribed audio, plus another 90 hours of audio that was automatically transcribed for each language (semi-supervised learning). These features were shown to reduce the WER of the resulting systems by roughly 10% absolute relative to simple PLP features.

We used the BYBLOS speech recognition system for our experiments. While there are different configurations, we used a fairly standard GMM-based HMM system. The acoustic models were trained on the same 10 hours of audio with transcriptions plus another 90 hours of untranscribed audio (semi-supervised learning). The speech recognition search uses multiple passes in order to provide recognition results efficiently. The first pass uses a forward pass based on a phonetic tree structure with an approximate bigram search. The second (backward) pass uses more complete models and is greatly sped up by the use of forward-backward pruning. At the same time, a lattice of all possible word ends is saved. The lattice is expanded for rescoring with a trigram language model and cross-word acoustic model rescoring. Finally, we compute the posterior probability of each arc in the lattice using a forward-backward pass on the lattice scores.

As	Be	Ha	La	Zu
28.5	31.5	16.3	10.2	62.8

Table 3: Percentage of keywords which are OOVs.

The audio corpora and keyword sets that we considered in our research were provided by the IARPA Babel program (LimitedLP releases). The languages studied and the corresponding releases were Assamese (IARPA-babel102b-v0.5a), Bengali (IARPA-babel103b-v0.4b), Haitian Creole (IARPA-babel201b-v0.2b), Lao (IARPA-babel203b-v3.1a) and Zulu (IARPA-babel206b-v0.1e). Each of these languages has different structure, phonetically, lexically, morphologically, and grammatically. The WERs obtained on the five languages are shown in Table 2. As can be seen, the corpora are quite difficult, with WERs around 60%. Each keyword set consisted of 2000 keywords, generated using artificial means by BBN [24] and supplied to all participants in the Babel program. The proportions of OOV keywords are shown in Table 3.

Table 4 shows the ATWV separately for the IV and OOV keywords and for all keywords, for each of the recognition and search methods we studied. For each search method, we show results obtained with the product of the posteriors, except for the whole-word search where we show the geometric mean for all 5 languages (was at most 1% better than the product). The last row of the table shows results obtained when combining the keyword list outputs across all of the decodings and search methods. The weights in the combination were chosen to maximize performance on the Dev set. The IV and OOV keywords were combined independently, to better leverage the strengths of the subwords and search methods used in each case. The absolute ATWV gain over using a single keyword list is about 1%. Also, the average Spearman correlation [21] between the sets of weights that resulted from Powell's method for the IV and OOV keywords was very low[1], approximately 0.13; this demonstrates that the usefulness of each system and search method varies significantly, depending on the keyword set of interest.

The first observation is that the phonetic search methods significantly outperform each of the corresponding whole-unit search methods for the OOV keywords, for all languages. For the IV keywords, whole-word search clearly outperforms all other search methods and vocabularies, possibly owing to the

[1]To compute the Spearman correlation, we sorted the systems according to the weights used in the combination for the IV keywords and compared with the corresponding sorting for the OOV keywords. If the two sortings were the same, Spearman correlation would have been equal to 1.

		Assamese			Bengali			Haitian Creole			Lao			Zulu		
rec-unit	search	IV	OOV	All	IV	OOV	All	IV	OOV	All	IV	OOV	All	IV	OOV	All
word	unit	**34**	0	25	**38**	0	26	**49**	1	41	**47**	0	42	**36**	0	13
word	phone	25	12	22	32	16	27	47	39	45	45	24	**43**	27	17	20
syllable	unit	21	7	17	33	10	26	47	20	43	38	8	35	22	13	17
syllable	phone	27	**26**	**27**	31	**29**	**31**	46	**52**	47	43	**32**	42	21	**30**	**27**
1-3ph	unit	17	10	15	25	12	21	41	28	39	36	11	34	18	16	17
1-3ph	phone	21	25	22	27	25	26	42	51	43	40	29	39	15	27	23
1-4ph	unit	17	10	15	26	14	22	41	29	39	36	10	34	20	18	18
1-4ph	phone	22	25	23	28	27	27	43	51	44	39	28	38	18	29	25
1-5ph	unit	18	11	16	27	14	23	42	30	40	36	11	34	19	19	19
1-5ph	phone	22	25	23	28	26	27	44	51	45	39	26	38	18	29	25
1-∞ph	unit	18	9	15	26	15	23	42	30	40	37	11	34	22	15	18
1-∞ph	phone	22	24	23	29	26	28	47	**52**	**48**	42	26	40	22	25	24
combination		38	33	36	40	35	39	57	61	57	56	39	54	37	38	38

Table 4: ATWV Results (shown as percentages) on the Dev sets of the five languages of the second year of the IARPA program Babel using various subword models and search operations. We rounded to the nearest 1% because the 95% confidence intervals are greater than 1%. The table gives results for IV, OOV, and All keywords for each recognition unit and either whole unit ("unit") or fuzzy phonetic ("phone") search for each language. The best ATWV in each column (among the single-system results) is shown in **bold**. The OOV result for the whole-word decode for Haitian Creole is nonzero because a few OOV terms were *compounds* of known words.

use of a strong language model. (A similar observation was made in [2].)

The second observation is that the syllable models are almost always better than the 1-N phone subword models, when used with fuzzy phonetic search (only Haitian Creole seems to be an exception). This is somewhat surprising because the syllable models are a subset of the 1-N phone subwords. But obviously there is some interaction among the models. The syllable models provide a very compact set of units with a nominal coverage at least 90% of the OOV keywords. Despite the fact that the 1-N phone subwords cover 100% of the OOV keywords, the resulting posteriors are "weaker" when single-phone units are involved in the recognition. Search with syllables seems to do best (of the subwords) for the IV keywords.

The third observation is that the combined results are very much better than the best individual results in all cases, giving ATWV gains of at least 6% absolute on the OOV keywords. When compared to just using fuzzy phonetic search on top of whole-word decoding, the gain is almost 20% in most cases. Note that the basic acoustic model and features are the same in all of these experiments; only the recognition units and the search methods change. Observe that the final performance for the OOV terms is surprisingly close to that of the IV words, which is much better than where we started.

Furthermore, we have found that the ATWV gain from including the "unlimited" subword search in the combinations is at most 1%; given that it is expensive to run decoding with it (results in very dense lattices), one could omit it without significant loss in performance.

We also performed blind test experiments by holding out a subset of the Dev data, and the above qualitative conclusions were still valid (the gain from system combination on the OOV keywords was about 2% on average over the five languages, when compared to the best individual system. When compared to just fuzzy phonetic search on top of whole-word decoding, the gain was about 13.6% on average.)

6. Conclusions

In this paper we investigated the use of subword units (syllables, as well as phone sequences of variable lengths) for speech recognition with the goal of improving the detection of OOV keywords. We showed that the gain from using such units is very large (nominally 20% absolute), when compared to using only phonetic search on top of whole-word decoding. We also demonstrated the usefulness of treating the IV and OOV keywords independently; ranking of the systems and weights used in system combination differ significantly in the two cases. In future work we plan to investigate, in more detail, how the different search methods and subword units affect the performance of each keyword subset.

7. Acknowledgments

We would like to thank all members of the BBN Speech and Language group for useful discussions, especially those who work on the Babel project. Special thanks go to Shivesh Ranjan for providing acoustic models and whole-word results, Le Zhang for his contribution to the computational infrastructure, Jacob Devlin for his help with computational issues, and to Ivan Bulyko, Stavros Tsakalidis and Bing Zhang for useful discussions. We would also like to acknowledge the help and contribution of other partners of the BABELON team on the IARPA-funded Babel project.

This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense US Army Research Laboratory contract number W911NF-12-C-0013. 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.

8. References

[1] K. Ng and V. W. Zue, "Subword-based approaches for spoken document retrieval," *Speech Communication*, vol. 32, no. 3, pp. 157–186, 2000.

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

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

[4] J. Mamou, B. Ramabhadran, and O. Siohan, "Vocabulary independent spoken term detection," in *Proc. of SIGIR'07*, Amsterdam, The Netherlands, July 2007.

[5] F. Seide, P. Yu, C. Ma, and E. Chang, "Vocabulary-independent search in spontaneous speech," in *Proc. of ICASSP*, 2004.

[6] A. Rastrow, A. Sethy, and B. Ramabhadran, "A new method for OOV detection using hybrid word/fragment system," in *Proc. of ICASSP*, 2009.

[7] A. Rastrow, A. Sethy, B. Ramabhadran, and F. Jelinek, "Towards using hybrid word and fragment units for vocabulary independent LVCSR systems," in *Proc. of Interspeech*, 2009.

[8] K. Narasimhan, D. Karakos, R. Schwartz, S. Tsakalidis, and R. Barzilay, "Does morphological segmentation help keyword spotting?" unpublished.

[9] P. Baumann, H. Fang, R. He, B. Hutchinson, A. Jaech, E. Fosler-Lussier, M. Ostendorf, J. Pierrehumbert, A. Janin, and S. Wegmann, "Leveraging morphology for dealing with data sparsity," presented at the IARPA Babel PI Meeting, January 2014.

[10] J. Mamou and B. Ramabhadran, "Phonetic query expansion for spoken document retrieval," in *Proc. of Interspeech*, 2008, pp. 2106–2109.

[11] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech and Language*, vol. 14, no. 4, pp. 373–400, 2000.

[12] I. Bulyko, O. Kimball, M.-H. Siu, J. Herrero, and D. Blum, "Detection of unseen words in conversational Mandarin," in *Proc. of ICASSP*, Kyoto, Japan, Mar 2012.

[13] I. Szoke, L. Burget, J. Cernocký, and M. Fapo, "Sub-word modeling of out of vocabulary words in spoken term detection," in *IEEE Workshop on Spoken Language Technology*, India, 2008.

[14] W. Hartmann, L. Lamel, and J.-L. Gauvain, "Cross-word subword units for low-resource keyword spotting," in *Proc. of SLTU*, 2014.

[15] NIST, "OpenKWS13 keyword search evaluation plan," http://www.nist.gov/itl/iad/mig/upload/OpenKWS13-EvalPlan.pdf, 2013.

[16] G. Chen, O. Yilmaz, J. Trmal, D. Povey, and S. Khudanpur, "Using proxies for OOV keywords in the keyword search task," in *Proc. of ASRU*, 2013.

[17] M. Saraclar, A. Sethy, B. Ramabhadran, L. Mangu, J. Cui, X. Cui, B. Kingsbury, and J. Mamou, "An empirical study of confusion modeling in keyword search for low resource languages," in *Proc. of ASRU*, 2013.

[18] L. Mangu, B. Kingsbury, H. Soltau, H.-K. Kuo, and M. Picheny, "Efficient spoken term detection using confusion networks," in *Proc. of ICASSP*, 2014.

[19] D. Karakos, R. Schwartz, S. Tsakalidis, L. Zhang, S. Ranjan, T. Ng, R. Hsiao, G. Saikumar, I. Bulyko, L. Nguyen, J. Makhoul, F. Grezl, M. Hannemann, M. Karafiat, I. Szoke, K. Vesely, L. Lamel, and V.-B. Le, "Score normalization and system combination for improved keyword spotting," in *Proc. of ASRU*, Olomouc, Czech Republic, 2013.

[20] D. Karakos, I. Bulyko, R. Schwartz, S. Tsakalidis, L. Nguyen, and J. Makhoul, "Normalization of phonetic keyword search scores," in *Proc. of ICASSP*, Florence, Italy, 2014.

[21] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The art of Scientific Computing*. Cambridge University Press, 2007.

[22] S. Tsakalidis, R. Hsiao, D. Karakos, T. Ng, S. Ranjan, G. Saikumar, L. Zhang, L. Nguyen, R. Schwartz, and J. Makhoul, "The 2013 BBN Vietnamese telephone speech keyword spotting system," in *Proc. of ICASSP*, Florence, Italy, 2014.

[23] M. Karafiát, F. Grézl, M. Hannemann, K. Veselý, and J. H. Černocký, "BUT Babel system for spontaneous Cantonese," in *Proc. of Interspeech*, 2013.

[24] D. Karakos and R. Schwartz, "Automatic keyword selection," presented at the IARPA Babel PI Meeting, January 2014.