BALANCING FALSE ALARMS AND HITS IN SPOKEN TERM DETECTION

Carolina Parada [1]*, Abhinav Sethy* [2]*, Bhuvana Ramabhadran* [2]

[1] Human Language Technology Center of Excellence
and Center for Language and Speech Processing, Johns Hopkins University
3400 North Charles Street, Baltimore MD 21210, USA

[2] IBM T.J. Watson Research Center
Yorktown Heights, N.Y. 10568, USA

ABSTRACT

This paper presents methods to improve retrieval of Out-Of-Vocabulary (OOV) terms in a Spoken Term Detection (STD) system. We demonstrate that automated tagging of OOV regions helps to reduce false alarms while incorporating phonetic confusability increases the hits. Additional features that boost the probability of a hit in accordance with the number of neighboring hits for the same query and query-length normalization also improve the overall performance of the spoken-term detection system. We show that these methods can be combined effectively to provide a relative improvement of 21% in Average Term Weighted Value (ATWV) on a 100-hour corpus with 1290 OOV-only queries and 2% relative on the NIST 2006 STD task, where only 16 of the 1107 queries were OOV terms. Lastly, we present results to show that the proposed methods are general enough to work well in query-by-example based spoken-term detection, and in mismatched situations when the representation of the index being searched through and the queries are not generated by the same system.

Index Terms— Spoken Term Detection, OOV Detection

1. MOTIVATION

The most common approach to STD is the use of a large vocabulary continuous speech recognition (LVCSR) system to obtain word/subword/phonetic lattices that are subsequently indexed [1, 2, 3]. There are many challenges in finding a good operation point for a Spoken Term Detection (STD) system that balances false alarms and true hits, particularly when the queries are Out of Vocabulary (OOV) terms for the LVCSR system. In [2], we presented a Weighted Finite State Transducer (WFST) based indexing system modeled along the lines of [3, 4], which allows us to use the lattice representation of the audio directly as a query to the search system. This enabled us to compare the performance of the STD system when presented with textual queries and queries represented by sample audio from an existing index and conclude that a two-pass approach that uses the hits from text-based queries to refine search results can enhance the performance of a STD system. While increased N-best representation of queries did not translate to significant improvements in STD performance, modeling phonetic confusability, showed the potential to compensate for potential differences in deriving index and query representations. Such mismatches can be addressed by incorporating phone confusability. Table 1 illustrates a typical OOV query from our test-set which was completely missed by the baseline STD system, due to mismatch in some phones between the pronunciation used for the query and phonetic sequence present in the decoded lattice that was indexed. Representing the query by the top 6 pronunciations from the Letter to Sound models (L2S) does not match any of the phone representations found in the lattice (i.e. G AE V L [EH or AE] K). Including phone confusability allows for phone substitutions, deletions, and insertions which helps reduce these misses. In this example, allowing substitution of AA by EH in fifth position will be sufficient.

query	GAVLAK					
reference pron	G	AE	V	L	AA	K
L2S 6 best prons	G Y	AE AA AX EY	V	L	AA AY	K
index	**Candidate Hits**					
word decode	GET			LIKE		
hybrid decode	G_AE_V			L_EH_K		
phonetic lattice from hyb	G	AE	V	L	EH AE	K

Table 1. Example of various phonetic representations for an OOV query and potential hits in the phonetic indices

This paper addresses specific algorithmic improvements to the STD system to reduce the false alarm rate while increasing the number of hits in an STD system, taking advantage of the fact that the query terms are all OOVs to the LVCSR system. The rest of this paper is organized as follows. Sections 2 and 3 describe the corpora and the baseline STD system

used throughout the paper. Section 4 proposes features and methods to reduce false alarms while reducing misses. The paper concludes by summarizing the combined performance of the enhanced STD system in Section 5.

2. CORPORA AND ASR SYSTEM

For our experiments we use an 100 Hour spoken term detection corpus especially designed to emphasize OOV content [3]. The 1290 OOVs in the corpus were selected with a minimum of 5 acoustic instances per word, and common English words were filtered out to obtain meaningful OOVs, excluding short (less than 4 phones) queries.

The Hybrid LVCSR system was built using the IBM Speech Recognition Toolkit [5] with utterances containing OOV words excluded. The acoustic models trained on 300 hours of HUB4 data. The language model for the LVCSR system was trained on 400M words with a lexicon of 83K words and 20K fragments derived using [6]. The excluded utterances (around 100 hours) were divided into development (5 hours) and test for parameter tuning in all STD experiments. We will refer to this set as **OOVCORP**. We also presents results on the NIST 2006 Spoken Term Detection Dev06 test set which we will refer to as **DEV06**. This set comprises of around 3 hours of speech and has 1107 query terms. The OOV rate on this set is low, with only 16 terms being OOV.

3. WFST-BASED SPOKEN TERM DETECTION SYSTEM

Our WFST based STD system is described in [3, 4]. We assume that the audio to be indexed has been processed with an hybrid LVCSR system and the corresponding word or sub-word lattices are available. Phonetic lattices are subsequently derived and used to build the indexes used in all of our experiments as described in [2].

At search time the textual queries are converted to their phonetic representation using the pronunciations obtained from the L2S system described in [7], which we refer to as *L2S*. We use as baseline the system presented in [2], which was evaluated on the same data set. Throughout this paper, we use as the baseline, the best performing STD system with an ATWV of 0.342 [2].

4. ALGORITHMIC IMPROVEMENTS

4.1. Reducing False Alarms

4.1.1. OOV-Detection

In this section we explore the benefits of incorporating automatic detection of OOV regions in the indexed audio. To identify these regions we employed the OOV detector introduced by [8], which uses a hybrid system combining words and data-driven variable length sub word units. This system combines the posterior probability of sub word units and the entropy of the subword unit in the region of interest to detect OOVs. The posterior probabilities of each unit in the confusion (consensus) networks of the indexed audio are used to detect OOV-segments and provide an associated confidence score given by the Equation 1:

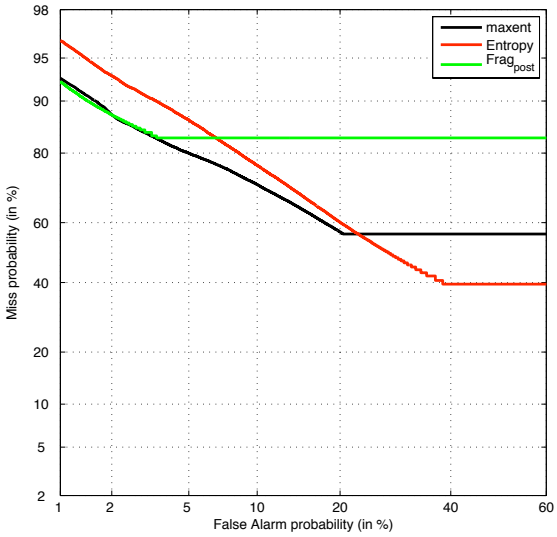

Fig. 1. OOV detection DET curve on OOVCORP-test

$$OOV_{scr}(\{t_j\}) = \sum_{f \in \{t_j\}} p(f|t_j) \quad (1)$$

where $\{t_j\}$ is the current region, and f is the sub-word (fragment) in that region. The sub-word entropy for each confusion bin [8], is combined with the OOV confidence score using a Maximum Entropy Classifier. We incorporate the OOV-detection (Figure 1 plots the DET curve) as a post-processing step to the baseline STD system described in Section 3. Let the score of an occurrence of a query-term Q at time interval Δ_t (returned by the baseline system) be denoted by $score_Q(\Delta_t)$. The updated score is given by the Equation below.

$$score_Q(\Delta_t, \gamma_o) = \begin{cases} score_Q(\Delta_t) & OOV_{scr}(\Delta_t) > 0 \\ score_Q(\Delta_t) \times \gamma_o & o/w \end{cases}$$

where γ_o is a parameter tuned on the development set, which penalizes the mismatch between query-type (OOV) and false alarms returned from In-Vocabulary (IV) regions.

4.1.2. Query Length Normalization

In order to further reduce False Alarms, we incorporated a normalization penalty based on the length of the query term.

Since hits with a longer duration are less likely to be false alarms, we adjust the score in a post-processing step as shown in Equation 2 below:

$$score_q(\Delta_t, \gamma_L) = score_q(\Delta_t)^{\frac{\gamma_L}{\Delta_{avg}(q)}} \quad (2)$$

where $\Delta_{avg}(q)$ is the average duration of all returned hits for the query-term q, and $\gamma_L \in [0, 1]$ is a parameter tuned on the development set. This approach is similar to the query-length normalization presented by [1], where the authors normalize the scores by a fixed factor $\gamma_L = 1/n$, where n is the number of words in the query phrase.

4.2. Increasing Hits

4.2.1. Cache feature

Certain content words, especially rare words, tend to appear in bursts. Inspired by the Cache Language Model [9], which adapts probabilities of ngrams based on the recently encountered n-grams, we adapt the probability of a potential hit, based on the number of neighboring hits for the same query. Specifically, we boost the score of each hit as shown below:

$$score_Q(\Delta_t, \delta) = score_Q(\Delta_t)^{1/\#hits \in \Delta_{t \pm \delta}} \quad (3)$$

where , $\delta \in [0, 1000] secs$ is again tuned on the development set.

4.2.2. Incorporating Phonetic confusions

In assessing the match of indexed lattices with search queries, recognition errors must be accounted for. One method relies on modeling typical confusions between phones. In this work, we derive this confusion matrix from broadcast news development data. We also present results with a phonetic confusion transducer (P2P) generated from a phonetic confusion matrix based on a neural network based acoustic model [10]. Similar results were obtained with an alternate approach to compute a confusion matrix using string alignments between reference and decoded phonetic representations. To augment the query with phonetic confusions we compose the WFSA representation of the query with the P2P transducer and generate n-bests. More formally, for any string s, let $I(s)$ be the transducer that maps each character is s to itself , let L2S be the transducer that maps any letter string to a set of possible pronunciations, and P2P be the transducer that maps phones to phones, where the weights are obtained from the P2P system explained above. The new query representation is then given by:

$$qfst \quad = \quad \text{bestpathN}(I(q) \circ L2S \circ P2P) \quad (4)$$

where q is the textual query, $\circ$ and $bestpath$ correspond to the standard composition and shortest-path operations (in *Tropical* semiring) for WFSTs.

5. EXPERIMENTAL RESULTS

This section presents results to illustrate the impact of each feature proposed in Section 4. Tables 2 and 4 present a summary of the results on the two corpora, DEV06 and OOV-CORP, respectively. The tables represent the features used in each experiment with a "X" mark against their respective columns.

While all features provide an incremental gain in ATWV, on the DEV06 corpus (See Table 2), it can be seen that use of the *cache feature*, i.e. proximity to a potential hit, provides the maximum gain in ATWV by increasing the number of hits. Although very small, the use of an OOV-detector does reduce the false alarms without impacting the number of hits. An oracle experiment that uses true OOV-regions from the reference can reduce the false alarms from 388 to 293, suggesting that more improvements in ATWV can be obtained with better OOV detection.

oovdet	length-norm	cache	Hits	FAs	ATWV
			4752	388	0.849
x			4752	383	0.8497
	x		4845	427	0.8520
		x	4907	400	0.8551
x	x	x	5011	452	**0.8597**

Table 2. DEV06 Results using Automatic OOV-detector

Table 3 illustrates the relative improvement over the baseline system for the OOVCORP corpus, obtained when incorporating a P2P transducer in deriving several n-best query representations. In all cases, a significant improvement in ATWV performance is seen for OOVCORP. For Dev06, we did not see any improvements with P2P since there were only 16 OOV terms and the In-Vocabulary terms were most likely captured by the word portion of the hybrid LVCSR system.

P2P-Nbest	none	10best	20best	100best
ATWV	0.342	0.368	0.384	**0.398**
%rel improv	-	7.6%	12.3%	**16.4%**

Table 3. Phone-to-phone transducer for N-best query representation (OOVCORP)

The queries for the experiments on OOVCORP use the 100-best representations generated after composition with the P2P transducer which has a best result of 0.398 in Table 3. Using the OOV detector we were able to reduce false alarms significantly, but the hits were also reduced leading to no improvement in ATWV. The cumulative improvement over the baseline with no P2P (with ATWV 0.342) is 21% relative.

Next, we present results in a Query by Example (QbyE) setup where the queries are specified as cuts or splices of an audio stream [2]. In this case the query transducer is a cut or a section of the lattices generated by the decoder. Table

oovdet	length-norm	cache	Hits	FAs	ATWV
			9027	28472	0.398
x			8611	24378	0.399
	x		10053	25630	0.412
		x	9027	28472	0.398
	x	x	10053	25630	0.412
x	x	x	10320	35811	**0.415**

Table 4. OOVCORP Results using Automatic OOV-detector, Length-normalization, and Cache Features.

5 presents results after applying a phonetic confusion transducer to the splice section of the lattice and generating n-best representation. For QbyE since the query is a transducer excised from the output of the system that generated the index, it is likely that the query transducer already models the confusions generated while producing the index. Adding additional confusability with a phone confusion transducer does not benefit in QbyE. However, for the case where the query is generated by an out-of-domain system it is conceivable that the phone confusion transducer will still provide gains. To test this hypothesis we used a conversational telephony system to generate query transducers. The query transducers were then used to search the index generated by the broadcast news sytem. The results are presented in Table 6. It can be seen that in the case where the system that generates the query is mismatched we get modest gains with a phone confusion transducer. It should be noted that for these experiments we used as query terms, the lattice cuts which had the minimal phone error rate [2]. The baseline performance for QbyE with these query cuts is 0.481 as compared to 0.398 (with 100 P2P) for textual queries (Table 3).

N-best	1	5	10	20
QbyE	0.481	0.454	-	-
QbyE+P2P	0.481	0.451	0.433	0.424

Table 5. Query by Example results incorporating P2P with n-best representations

N-best	1	5	10	20
QbyE	0.200	0.207	0.201	0.196
QbyE+P2P	0.200	0.210	0.210	0.211

Table 6. Cross domain QbyE incorporating P2P with n-best representations

6. CONCLUSIONS

In this paper, we have explored features to increase hits and reduce false alarms in spoken term detection systems on two different corpora. All features collectively provide an additive gain on both corpora, resulting in a relative 21% improvement in ATWV on a 100-hour corpus with 1290 OOV-only queries and 2% relative on the NIST 2006 STD task, where only 16 of the 1107 queries were OOV terms. However, no single feature by itself provides significant improvement in terms of ATWV. The cache feature which defines the proximity to a potential hit provides an expected increase in number of hits, and normalizing the query length not only reduces false alarms but reduces misses as well. The performance of the OOV detector was rather disappointing, suggesting that future work needs to focus on this intuitive method to reduce false alarms for OOV queries. Lastly, incorporating phone confusability provides significant improvements for textual queries and for cross domain Query-by-Example.

7. REFERENCES

[1] Jonathan Mamou, Bhuvana Ramabhadran, and Olivier Siohan, "Vocabulary independent spoken term detection," in *Proceedings of SIGIR*, 2007.

[2] Carolina Parada, Abhinav Sethy, and Bhuvana Ramabhadran, "Query-by-example spoken term detection for oov terms," in *ASRU*, 2009.

[3] Dogan Can, Erica Cooper, Abhinav Sethy, Chris White, Bhuvana Ramabhadran, and Murat Saraclar, "Effect of pronounciations on OOV queries in spoken term detection," *Proceedings of ICASSP*, 2009.

[4] Cyril Allauzen, Mehryar Mohri, and Murat, "General indexation of weighted automata - application to spoken utterance retrieval," in *Proceedings of HLT*, 2004.

[5] H. Soltau, B. Kingsbury, L. Mangu, D. Povey, G. Saon, and G. Zweig, "The ibm 2004 conversational telephony system for rich transcription," in *ICASSP*, 2005.

[6] Ariya Rastrow, Abhinav Sethy, Bhuvana Ramabhadran, and Fred Jelinek, "Towards using hybrid, word, and fragment units for vocabulary independent LVCSR systems," *INTERSPEECH*, 2009.

[7] Stanley F. Chen, "Conditional and joint models for grapheme-to-phoneme conversion," in *Eurospeech*, 2003, pp. 2033–2036.

[8] Ariya Rastrow, Abhinav Sethy, and Bhuvana Ramabhadran, "A new method for OOV detection using hybrid word/fragment system," *Proceedings of ICASSP*, 2009.

[9] F. Jelinek, B. Merialdo, S. Roukos, and M. Strauss, "A dynamic language model for speech recognition," in *Proceedings of HLT*, 1991.

[10] Bhuvana Ramabhadran, Abhinav Sethy, Jonathan Mamou, Brian Kingsbury, and Upendra Chaudhari, "Fast decoding for open vocabulary spoken term detection," in *Proceedings of HLT*, June 2009.