A Spoken Term Detection Framework for Recovering Out-of-Vocabulary Words Using the Web

Carolina Parada[1], *Abhinav Sethy*[2], *Mark Dredze*[1], *Frederick Jelinek*[1]

[1]Human Language Technology Center of Excellence
Center for Language and Speech Processing, Johns Hopkins University
3400 N Charles Street, Baltimore MD 21210
[2]IBM TJ Watson Research Center, New York, NY, 10598
carolinap@jhu.edu, asethy@us.ibm.com, mdredze@cs.jhu.edu, jelinek@jhu.edu

Abstract

Vocabulary restrictions in large vocabulary continuous speech recognition (LVCSR) systems mean that out-of-vocabulary (OOV) words are lost in the output. However, OOV words tend to be information rich terms (often named entities) and their omission from the transcript negatively affects both usability and downstream NLP technologies, such as machine translation or knowledge distillation. We propose a novel approach to OOV recovery that uses a spoken term detection (STD) framework. Given an identified OOV region in the LVCSR output, we recover the uttered OOVs by utilizing contextual information and the vast and constantly updated vocabulary on the Web. Discovered words are integrated into system output, recovering up to 40% of OOVs and resulting in a reduction in system error.
Index Terms: language modeling, data selection, spoken term detection, oov detection

1. Introduction

Large vocabulary continuous speech recognition (LVCSR) systems typically operate with a fixed decoding vocabulary so they encounter out-of-vocabulary (OOV) words, especially in new domains or genres. New words can be named entities, foreign, rare and invented words that are not in the system's vocabulary and therefore will be transcribed incorrectly. Additionally, OOVs typically contribute to recognition errors in surrounding words, and propagate to later processing stages such as machine translation or natural language understanding. Critically, OOVs are often information rich – miss-recognized OOVs can have a greater impact on understanding the transcript.

Considerable research interest has focused on OOVs in the context of spoken term detection (STD) [1, 2], which aims for open vocabulary search over large spoken document collections. In STD a typical approach for solving the OOV issue is to convert the speech to phonetic transcripts and represent queries as phone sequences. Such transcripts can be generated by expanding the word transcripts using the pronunciation dictionary of the LVCSR system or by the use of subword based LVCSR systems. Retrieval is based on searching the sequence of subwords representing the query in the subword transcripts.

In this paper, we present a novel approach for *recovering* the orthographic form of OOV terms using a STD framework where the STD queries are generated using the Web as a corpus. In the proposed approach, the audio is first processed by a LVCSR system producing both word and sub-word lattices. Our objective is to correct output errors due to OOVs to improve the quality of the automatic transcript. We assume that OOVs in the output lattices (OOV regions) have been identified using either an oracle or an automatic OOV detector [3, 4, 5].

For each utterance containing an OOV region, we generate queries for a Web search engine to find words relevant to the utterance topic, which are then incorporated into the recognition output using a STD system. For each candidate OOV (STD query), the system performs a search over the LVCSR lattices and retrieves all instances (hits) where the query matches the lattice's phonetic sequence. The STD hits replace the best LVCSR hypothesis in each OOV region without the need to re-decode or modify the LVCSR system. In a media monitoring/surveillance/browsing system it is impractical to re-decode and reindex large amounts of data. The proposed solution can be integrated naturally with the speech retrieval architecture with minimal cost.

Our approach for identifying relevant OOV terms from the Web is described in Section 2. Section 3 describes how the STD system is used for OOV recovery. Sections 4 and 5 describe the experimental setup and the results analysis respectively. We conclude with a summary of the key findings of this paper.

2. Identifying Relevant OOVs on the Web

Our OOV recovery approach exploits the lexical context of OOV regions to query the Web and retrieve content relevant to the utterance. For each utterance containing an OOV region, we select M relevant words in the utterance and submit the set of words as a single query to a search engine (Google). To identify keywords, we rank the decoded words using TF-IDF. This is a common information retrieval score used to evaluate how important word t_i is to document d_j in a collection of documents D, as given in the expression below:

$$\text{TF-IDF}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \log \frac{|D|}{|d : t_i \in d|} \quad (1)$$

where $n_{i,j}$ is the number of occurrences of term t_i in document (utterance) d_j, $|d : t_i \in d|$ is the number of documents (utterances) where the term t_i appears, and $|D|$ is the number of documents (utterances). We considered only words with confidence[1] larger than a threshold $T = 0.8$, and not present in a standard stop-list. We selected the top M keywords per utterance ($M = 5$) as a query to the search engine and retrieved the top 20 documents. Each retrieved document was scanned

[1]We used as confidence the posterior probability associated with each word in the output confusion networks.

Window Size	Recall	List Size
1	7.86	36K
3	22.78	105K
5	38.15	170K
10	71.24	323K
Paragraph	91.55	1.4M

Table 1: Recall versus candidate list set size.

to create a list of "potential OOVs" containing all OOV words within a fixed window around the query terms. Some sample queries which we will refer to in our results (Section 5) include "MEDIATORS UNDERMINE EXTREME NATIONALIST BOSNIAN " which helped us to recover the OOV word *Milosevic* and the query "POLLS REFERENDUM SHARON ISRAELIS WEDNESDAY" which helped recover *Netanyahu*.

Table 1 shows for various window sizes around the query terms the tradeoff between the size of the retrieved word set versus recall. In our experiments we used a window of 10 words. Note that in this work we consider a global list of potential OOVs rather than an utterance specific list. In our experiments we found that limiting the set of potential OOVs to the words retrieved using the current utterance had a significantly lower recall (30%) than creating a global list. This could be due to the fact that many utterances refer to the same topic, while their automatic transcription performance might vary significantly, yielding noisy or irrelevant keywords.

To reduce the size of the candidate list we remove bad candidates, such as misspelled words, typos, and invented words, by filtering words which appear less than K times in the retrieved documents. Figure 1 shows the recall of true OOVs versus candidate list size for the test set (Section 4). We selected $K = 4$ for our experiments based on development set tuning. This setting reduces the list to 29K while still retrieving 53% of the missing OOVs. We tried several other approaches to further reduce the candidate list, including: scoring candidates with a character letter N-gram model trained on In-Vocabulary (IV) words and excluding concatenated IVs (commonly found on the Web). There was no significant improvement over the simple strategy of shortlisting by Web counts.

Acquiring Web data using automatically generated queries similar to the approach described in this section has been of considerable interest as a means of increasing the amount of language model training data and incorporating new words into the LVCSR vocabulary [6, 7, 8]. In this paper we focus on using the OOV word list acquired from the Web as a source of query terms for a spoken term detection system, which can then help to identify and incorporate the OOV terms in LVCSR output without re-decoding or modifying the LVCSR system.

3. STD for OOV Recovery

In the previous section, we described a method for obtaining potential orthographic forms for uttered OOVs using contextual queries and the Web. With this candidate list a spoken term detection system is used to identify matches in the LVCSR output.

We use a weighted finite state transducer (WFST) based STD system described in [1, 9]. To build the index, we process the audio with a hybrid LVCSR system [10] to obtain the corresponding phonetic lattices. The resulting phonetic index is used in all of our experiments. At search time each textual query is converted to its phonetic representation using the pronunciations obtained from a letter-to-sound (L2S) system [11].

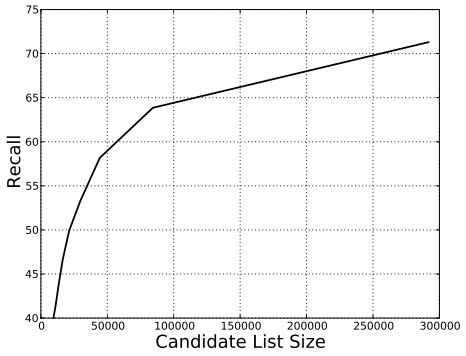

Figure 1: Unique OOV recall versus the size of the candidate list for different term frequency thresholds (window of 10).

3.1. Decision Thresholds for OOV Recovery

The performance of our OOV recovery approach depends on what candidate hits are accepted from the output of the STD system. [12] presented an optimal decision theoretic approach for selecting a term specific threshold (TST) on STD scores for deciding whether a postulated hit for a term should be included in the search output. The threshold was optimized for the NIST ATWV metric and has been widely adopted in the research community. Here we present an extension of TST targeted towards improving WER by OOV recovery.

Motivated by [12] we aim to select a threshold that leads to a reduction of WER. Given a hit for term t in region r with confidence score $P(t,r)$[2], we accept a hit if the benefit of correctly retrieving it (B) is larger than the cost of incorrectly retrieving it (C), as shown in Equation 2.

$$P(t,r)B - [1 - P(t,r)]C > 0 \tag{2}$$

Let T_{OOV} be all missing OOVs, R_{OOV} the set of all OOV regions, then in terms of WER, the benefit of correctly retrieving a missing OOV is $B = 1$ if $r \in R_{OOV}$ and $t \in T_{OOV}$, and cost $C = \beta$ if $r \notin R_{OOV}$, where β the cost of an error. We rewrite:

$$P(t,r)P(r \in R_{OOV})P(t \in T_{OOV}) - (1 - P(t,r))\beta P(r \notin R_{OOV}) > 0$$

Solving for $P(t,r)$ yields:

$$P(t,r) > \frac{1 - P(r \in R_{OOV})}{1 - P(r \in R_{OOV}) + \frac{P(t \in T_{OOV})P(r \in R_{OOV})}{\beta}} \tag{3}$$

where $P(r \in R_{OOV})$ is the probability that region r is an OOV region and $P(t \in T_{OOV})$ is the probability that the term t is a missing OOV. Note that if the region r is OOV with probability 1, then we should accept any hit with $P(t,r) > 0$ (all), since no additional errors will be incurred. However if the region is *not* an OOV region, i.e., $P(r \in R_{OOV}) = 0$, then we can only accept hits with probability $P(t,r) > 1$ (none), since an error will always be incurred. As the cost of incurring an error β increases, the threshold tends to 1.

3.2. Incorporating STD Matches in LVCSR Output

The matching regions in the LVCSR output are replaced with the OOV word. An alternative is to re-decode the utterance with an augmented vocabulary as proposed in [8]. Re-decoding means a tradeoff between recognition accuracy of in-vocabulary

[2] $P(t,r)$ is the probability of the acoustic match obtained from the STD system.

System	Recall Regions	Recall OOVs	WER
Baseline	-	-	17.0
STD recovery	29.60	40.71	16.4
Oracle recovery	100	100	14.6

Table 2: Recovery results using oracle OOV detection. Recall Regions denotes the number of OOV regions correctly identified, while Recall OOVs is the number of unique OOVs found.

words and rare words, which can degrade WER while recovering only a small fraction of the new words. Replacing the matched regions is faster as no re-decoding is necessary. Assuming we only replace words in OOV regions, it does not tradeoff accuracy of in-vocabulary words for rare words.

4. Experimental Setup

We used the data set constructed by [1] to evaluate STD of OOVs; we refer to this corpus as `OOVCORP`. The corpus contains 100 hours of transcribed English Broadcast News speech emphasizing OOVs. There are 1290 unique OOVs in the corpus, which were selected with a minimum of 5 acoustic instances per word. Common English words were filtered out to obtain meaningful OOVs: e.g., `NATALIE, PUTIN, QAEDA, HOLLOWAY`. Since the corpus was designed for STD, short OOVs (less than 4 phones) were explicitly excluded. This resulted in roughly 24K (2%) OOV tokens.

We used the IBM Speech Recognition Toolkit [3] [13] with acoustic models trained on 300 hours of HUB4 data [14] and excluded utterances containing OOV words as marked in `OOVCORP`. The language model was trained on 400M words from various text sources with a 83K word vocabulary. The LVCSR system's WER on the standard RT04 BN test set was 19.4%. Excluded utterances were divided into 5 hours of training for the OOV detector, 5% of development set, and 90 hours of test data for the OOV detector. Both train and test sets have a 2% OOV rate. We used this split for all experiments. Note that the OOV training set is different from the LVCSR training set.

In addition to a word-based LVCSR system, we use a hybrid LVCSR system, combining word and sub-word (fragments) units. Combined word/sub-word systems have improved OOV STD performance [2, 15], and achieved better phone error rates, especially in OOV regions [10], and yield state-of-the-art performance for OOV detection [3]. Our hybrid system's lexicon has 83K words and 20K fragments derived using [4]. The 1290 excluded words are OOVs to the word and hybrid systems.

5. Results

We first consider the case where OOV segments are identified using an oracle. We identify OOV regions by finding time segments in the manual transcripts containing words which are not in the LVCSR system vocabulary. This allows for the direct evaluation of the effectiveness of our retrieval strategy as well as the STD framework for integrating the words into the automatic transcripts. We also consider OOV regions that are detected automatically, which represents a more realistic scenario.

5.1. Recovery with Oracle OOV Detection

We use the STD system presented in Section 3 to phonetically match each retrieved word to the corresponding OOV regions in the decoded audio. We built a phonetic index from the lattices obtained from the LVCSR system, and consider the list of 29K words as OOV queries to the STD system. At search time the textual queries are converted to their phonetic representation using the pronunciations obtained from the letter to sound (L2S) system discussed in Section 4. Note that this query list contains 53% of the missing OOVs as described in Section 2.

Table 2 summarizes our results for searches with the top 6 weighted pronunciations for each query. We are able to recover 29.6% of the OOV regions and 40.7% of the unique OOVs. The remaining OOVs could not be assigned a recovered term due to mismatches between the hypothesized pronunciation and the phonetic string in the index. Further improvement could be achieved by modeling phonetic confusability as shown in [16].

The authors in [8] present an approach to recover OOVs using the Web. The retrieved words are included in a locally augmented lexicon and each region was re-decoded. The authors were able to recovered 7.7% of the missing OOVs in a 6 hour test-set of French Broadcast News, assuming the OOV regions had been manually identified. The STD approach we propose is significantly faster since no re-decoding is necessary. The performance for oracle recovery (Table 2) indicates to the potential WER improvement when all OOVs are correctly recovered and integrated in the transcripts, which is roughly equal to the corpus OOV rate. Clearly, there is room for improvement.

STD Threshold	Recall Regions	Recall OOVs	WER
Baseline	-	-	17.0
TST	13.94	31.75	18.9
TST+HT	10.02	11.24	16.9
TRST	10.67	14.70	16.9
TRST+HT	13.01	15.17	**16.8**
TRST+TST	15.42	**20.68**	17.0

Table 3: Recovery results using an automatic OOV detector.

5.2. Recovery with Automatic OOV Detection

We repeated the above experiments using the automatic OOV detection system presented in [3], which showed state-of-the-art performance on this data set. The OOV detector's performance on our 90 hour test set is presented in Figure 3 using a standard detection error tradeoff (DET) curve. We incorporate OOV detection as a post-processing step to the STD system, by penalizing mismatches between query type (OOV) and false alarms returned from in-vocabulary (IV) regions (as described in [16]). The mismatch penalty is tuned on the development set.

Table 3 shows the results in terms of recall and WER when using automatic OOV detection for different thresholds. Using the standard term specific threshold (TST) [12] resulted in degraded WER performance, unless it is combined with a hard-threshold (TST+HT). The TST however, achieves the highest recall (31.75%). The degraded WER and high recall can be explained by the fact that the TST is designed to maximize NIST ATWV metric, which assumes that every query appears in the index at least once. This guarantees that at least one hit is retrieved for every query processed by the STD system. For OOV recovery, the query terms include noise (invalid words) from the web, hence accepting at least one hit per term causes large number of false alarms and higher recall for true OOVs.

The best recall/WER tradeoff is obtained using the proposed term-region specific threshold combined with a hard-threshold (TRST + HT), which retrieves 15.17% of the missing OOVs and achieves an improvement of 0.2% in WER, which is statistically significant ($p < 0.001$).[4] This result is ex-

[3] We use the IBM system with speaker adaptive training based on maximum likelihood with no discriminative training.

[4] For statistical significance, we used the *mapsswe* test.

Original Decode		After Recovery
the netting yahoo government negotiated	$\Rightarrow$	the NETANYAHU government negotiated
former president slogan I'm a loss of itch	$\Rightarrow$	former president slogan MILOSEVIC

Figure 2: Example utterances before and after recovery (automatic OOV detection). Incorrect terms are underlined in the decoded string and corrections are emphasized. The system corrects the OOV in the first string and one of the two OOVs in the second (Slobodan remains incorrect as "slogan"), improving understandability significantly. See Section 2 for the Web queries that retrieved these OOVs.

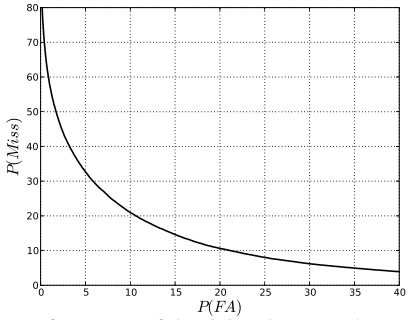

Figure 3: Performance of the OOV detector [3] on the test set.

pected since the threshold only allows hits in predicted OOV regions, yielding fewer false alarms. In this work we consider $P(t \in T_{OOV}) = 1$ for Equation 3. Including prior knowledge about the a candidate term should improve performance further.

Finally, we combined both thresholds (TRST + TST). If the score is higher than a hard-threshold HT, we used the TST. This ensures that hits with high scores are accepted even if the region is not labeled OOV. If the score is less than a hard threshold but the matching region is OOV, we accept it using the TRST. This achieves 20% recall, which improves the intelligibility of the transcript, while not degrading the WER.

From a random sample of 100 OOV words correctly recovered, 92% were named entities. From this set 68% were people, and 24% were locations, organization, or other.[5] As expected, the retrieved words are high information bearing words that improve understanding of the transcription significantly. Figure 2 depicts two example utterances from our test set before and after recovery using automatic OOV detection.

6. Conclusion

In this paper we have proposed a novel approach to recover Out-Of-Vocabulary words using the Web as a corpus. Our method incorporates the retrieved words using a Spoken Term Detection system. This approach is faster than previous proposed methods for lexicon augmentation [8] since it does not require re-decoding the audio. The best performance retrieves up to 40.71% of the missing OOVs, when assuming the OOV regions are correctly identified. This results in 0.6% absolute WER reduction of the baseline system. We also evaluate our approach when OOVs are automatically identified, the more realistic situation. While performance degrades due to OOV detector errors, we are still able to recover 15.17% of the missing OOVs improving WER by 0.2%, or retrieve 20.68% without increasing WER. Furthermore, the recovered words are 92% named entities, which improves the understanding of the transcription.

7. Acknowledgements

We thank Bhuvana Ramabhadran for many insightful discussions and the IBM Speech Group for the recognition lattices.

[5] We manually labeled the retrieved words in the true transcription.

8. References

[1] D. Can, E. Cooper, A. Sethy, C. White, B. Ramabhadran, and M. Saraclar, "Effect of pronounciations on OOV queries in spoken term detection," in *ICASSP*, 2009.

[2] J. Mamou, B. Ramabhadran, and O. Siohan, "Vocabulary independent spoken term detection," in *SIGIR*, 2007.

[3] C. Parada, M. Dredze, D. Filimonov, and F. Jelinek, "Contextual information improves oov detection in speech," in *NAACL-HLT*, 2010.

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

[5] L. B. et al., "Combination of strongly and weakly constrained recognizers for reliable detection of OOVS," in *ICASSP*, 2008.

[6] T. Ng, M. Ostendorf, M.-Y. Hwang, M. Siu, I. Bulyko, and X. Lei, "Web-data augmented language model for Mandarin speech recognition," in *ICASSP*, 2005.

[7] M. Creutz, S. Virpioja, and A. Kovaleva, "Web augmentation of language models for continuous speech recognition of sms text messages," in *EACL*, 2009.

[8] G. L. Stanislas Oger, Vladimir Popescu, "Using the world wide web for learning new words in continuous speech recognition tasks: Two case studies," in *SPECOM*, 2009.

[9] C. Allauzen, M. Mohri, and Murat, "General indexation of weighted automata - application to spoken utterance retrieval," in *NAACL-HLT*, 2004.

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

[11] Stanley F. Chen, "Conditional and joint models for grapheme-to-phoneme conversion," in *Eurospeech*, 2003.

[12] D. Miller, M. Kleber, C. lin Kao, and O. Kimball, "Rapid and accurate spoken term detection," in *INTERSPEECH*, 2007.

[13] 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.

[14] J. Fiscus, J. Garofolo, M. Przybocki, W. Fisher, and D. Pallett, *1997 English Broadcast News Speech (HUB4)*, Linguistic Data Consortium, Philadelphia, 1998.

[15] C. Parada, A. Sethy, and B. Ramabhadran, "Query-by-example spoken term detection for OOV terms," in *ASRU*, 2009.

[16] ——, "Balancing false alarms and hits in spoken term detection," in *ICASSP*, 2010.