Combination of FST and CN Search in Spoken Term Detection

Justin Chiu [1], *Yun Wang* [1], *Jan Trmal* [2], *Daniel Povey* [2], *Guoguo Chen* [2], *Alexander Rudnicky* [1]

[1] Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, USA
[2] Center for Language and Speech Processing & Human Language Technology Center of Excellence, Johns Hopkins University, Baltimore, MD, USA

Jchiu1@andrew.cmu.edu, maigoakisame@gmail.com, jtrmal@gmail.com, dpovey@gmail.com, guoguo@jhu.edu, Alex.Rudnicky@cs.cmu.edu

Abstract

Spoken Term Detection (STD) focuses on finding instances of a particular spoken word or phrase in an audio corpus. Most STD systems have a two-step pipeline, ASR followed by search. Two approaches to search are common, Confusion Network (CN) based search and Finite State Transducer (FST) based search. In this paper, we examine combination of these two different search approaches, using the same ASR output. We find that the CN search performs better on shorter queries, and FST search performs better on longer queries. By combining the different search results from the same ASR decoding, we achieve better performance compared to either search approach on its own. We also find that this improvement is additive to the usual combination of decoder results using different modeling techniques.

Index Terms: Spoken Term Detection, Keyword Search, System combination

1. Introduction

Spoken Term Detection (STD) [1] is one of the fundamental applications of automated speech processing. STD focuses on finding instances of a particular spoken word or phrase in an audio recording corpus. Previous research [2, 3, 4] in STD has shown high performance on rich resource languages, such as English, Chinese and Arabic. This research indicates that better STD performance can be achieved using an ASR framework, as opposed to direct (acoustic or phonetic) search. However, STD under conditions of limited resources [5], high quality ASR is not available [6, 7, 8, 9, 10, 11, 12]. This limitation focuses more attention to the search [13, 14, 15, 16, 17, 18, 19, 20, 21], since search has to be based on recognition hypotheses with high Word Error Rate (WER). The search is consequently done on all hypotheses generated from the decoding; recall is therefore more relevant than WER. Two common representations for hypotheses are lattices and confusion networks [22, 23, 24, 25], to each of which there is a different search method applicable: for searching on lattices, we can use Finite State Transducer (FST) search, while the Confusion Network (CN) search is applied to a condensed transform of the lattice. While both searches show good performance on the STD task, yet there are still issues that need to be addressed: What are the differences between these two approaches? More specifically, can we take advantage of complementarities between the two to achieve better STD performance?

This paper makes two contributions:

- We analyze (Section 6) the two different searches (Section 2) based on the same decoding result. We find (Section 5.1) that CN search performs better on single word queries, and FST search performs better on multiple word queries.
- We compare several search result combination techniques (Section 3), and show that combination leads to better STD results, without additional decoding (Section 5.2). If we add extra decoding results, we can provide additive improvement on the existing STD result. (Section 5.3)

2. Search Description

2.1. FST Search

Our FST search pipeline is described in [13, 14], which is capable of both in-vocabulary (IV) and out-of-vocabulary (OOV) search. We implement the lattice indexing algorithm proposed in [26] making use of the Kaldi toolkit [27].

At the indexing stage, the lattice of each utterance is expanded into a finite-state transducer, such that each successful path in the expanded transducer represents a single word or a sequence of words in the original lattice. The posterior score, start-time and end-time of the corresponding word or word sequence are then encoded as a 3-dimensional weight of the path. Our implementation of the indexing algorithm relies on the fact that the lattices are defeminized at the word level, which is an essential part of our lattice generation procedure [28]. Otherwise the indexing algorithm tends to blow up since the number of potential word sequences grows exponentially with the sequence length.

At the search stage, IV keywords are usually compiled into linear finite-state acceptors (FSA), with zero cost. OOV queries are mapped to IV queries (proxies) [14] according to phonetic similarity, which usually results in non-linear finite-state acceptors with different cost for each proxy. Regardless of being IV or OOV queries, STD is done by composing the query FSA with the index, and one can work out the posterior score, start-time and end-time from the weight of the resulting FST. In this work, we only focus on IV queries since most of the queries in our keyword lists are in-vocabulary.

2.2. CN Search

Our procedure for generating confusion networks is based on the Minimum Bayes Risk decoding algorithm of [29]. STD is carried out on confusion networks as follows. For single-word queries, each occurrence of the query word in the confusion networks generates a detection. The starting and ending times of the detection are those of the cluster containing the word; the score of the detection is the probability of the word. For multiple-word queries, dynamic programming is used to find all paths in the confusion networks such that the words on the path form the query. The paths may contain epsilon words. Each path generates a detection: the starting and ending times

are those of the first and last clusters in the path, and the score is the product of the probabilities of all the words (including epsilon words) in the path. If multiple detections for the same query overlap, only the one with the highest score is retained.

3. Search Result Combination Techniques

Search results from multiple systems or different search methods are combined on a per-keyword basis. For each keyword, its detections in all the search results are pooled together. These detections are regarded as nodes of a graph; an edge is drawn between two detections if they overlap. Each connected component of this graph generates a combined detection. The starting and ending times of the combined detection are calculated as the average of those of the individual detections; the score of the combined detection is calculated with one of the following three methods [30]:

- *CombMAX*: The score of the combined detection is the maximum of the scores of the individual detections;
- *CombSUM*: The score of the combined detection is the sum of the scores of the individual detections;
- *CombMNZ*: The score of the combined detection is the sum of the scores of the individual detections times the number of individual detections.

In *CombSUM* and *CombMNZ*, if some detection for a keyword ends up with scores larger than 1, then the scores of all detections for this keyword are divided by the highest score.

4. Experiments

4.1. Dataset

We use conversational (telephone) speech recorded in five different languages: Assamese[1], Bengali[2], Haitian[3], Lao[4] and Zulu[5], as available in the IARPA BABEL [5] program. For each language, there are 10 hours of training data and 10 hours of development data. We conduct our experiments using the development query sets and the development data.

4.2. The Evaluation metrics

Spoken Term Detection uses multiple metrics for evaluation. All metrics are based on the Term Weighted Value (TWV) [1]. The formula for TWV is as follows:

$$TWV(\theta) = 1 - (P_{Miss}(term,\theta) + \beta \cdot P_{FA}(term,\theta)) \quad (1)$$

where θ is the detection threshold, β is a factor that controls the balance between misses and false alarms, which is set to 999.9 in BABEL program.

The concept of TWV score is simple: If the system performs perfectly on a query, it has a TWV of 1; if the system misses some of the query words or produces false alarms, it receives penalty on the TWV score. As a result, the TWV score is bounded above by 1 but has no fixed lower bound.

We use two separate metrics to describe the performance of STD systems:

- Maximum Term Weighted Value (MTWV): MTWV is the maximum TWV over the range of all possible values of the detection threshold.
- Supreme Term Weighted Value (STWV): STWV is the maximum TWV without considering false alarms. It is similar to lattice recall for a given query.

The metrics are computed on a per-query basis, and then averaged for reporting. Together these two metrics provide better insight into the overall quality for our search results, as they are not sensitive to specific detection threshold.

4.3. Experimental Setup

We conducted three different sets of experiments. Each set is conducted on three different decoding systems: a Deep Neural Network (DNN) system, a Bottleneck Feature (BNF) system and a Perceptual Linear Prediction (PLP) system. Our search component only processes the IV queries, for the OOV queries, it does not output any result.

The first set of experiments compare the performance of the two different searches, FST search and CN search. The second set of experiments combine the search results from FST search and CN search to determine if we can obtain better STD performance. The final set of experiments combine all of our results to see if the gain from the individual systems is additive. The combination is also performed in different orders to note whether that affects the final result.

5. Results

5.1. Comparison between FST and CN Searches

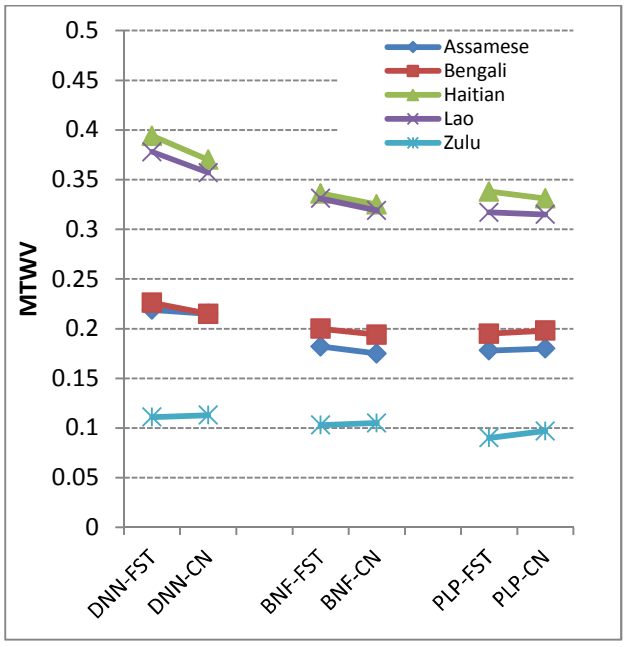

Figure 1: *System comparison between different decoding systems and search methods.*

Figure 1 shows the MTWV for different systems on five different languages (Assamese, Bengali, Haitian, Lao and Zulu) and 3 different decoder front-ends: Deep Neural Net

[1] Language collection release IARPA-babel102b-v0.5a.
[2] Language collection release IARPA-babel103b-v0.4b.
[3] Language collection release IARPA-babel201b-v0.2b.
[4] Language collection release IARPA-babel203b-v3.1a
[5] Language collection release IARPA-babel206b-v0.1e

(DNN), Bottle-neck Features (BNF), and Perceptual Linear Predictive (PLP). We performed a statistical analysis by fitting a general linear model to the data and found statistically significant differences between languages, front-end features and search methods, all at $p<0.001$. FST search generally outperforms CN search on every language except for Zulu. This is due to the distribution of query length (the number of word tokens per query) in the Zulu query set. We provide an analysis for it in Section 6.1.

Language	Metric	FST	CN	***CombSUM***
Assamese	MTWV	0.193	0.190	**0.203**
	STWV	0.369	0.372	**0.38**
Bengali	MTWV	0.207	0.202	**0.217**
	STWV	0.361	0.366	**0.373**
Haitian	MTWV	0.356	0.342	**0.368**
	STWV	0.496	0.501	**0.514**
Lao	MTWV	0.342	0.330	**0.358**
	STWV	0.474	0.476	**0.492**
Zulu	MTWV	0.101	0.105	**0.107**
	STWV	0.235	0.236	**0.236**

Table 1: *MTWV/STWV for search combination*

5.2. Combination of FST and CN searches

We evaluated three different combination techniques: *CombMAX*, *CombSUM* and *CombMNZ*. *CombSUM* appears to be the best way to combine FST and CN search. The results shown in Table 1 are averaged over front-end. It is worth noting that the performance on each decoding front-end shows the same trend as with the average performance. There are two observations that are worth making. First, the search combination has less effect on Zulu. This is due to the distribution of query length (see Table 3). Second, CN search has better performance on STWV over FST search. This is caused by the conversion from lattice to CN. The detail for both observations is discussed in the Analysis section (Section 6.1, 6.3).

5.3. Combination between decoding systems

The final set of experiments is carried out to determine whether the improvement from search combination is additive to the existing decoding based system combinations.

Language	Metric	Single Best	Search Comb.	**Search+ Decode**
Assamese	MTWV	0.219	0.229	**0.248**
	STWV	0.430	0.441	**0.465**
Bengali	MTWV	0.226	0.234	**0.258**
	STWV	0.407	0.417	**0.445**
Haitian	MTWV	0.394	0.402	**0.423**
	STWV	0.564	0.576	**0.597**
Lao	MTWV	0.378	0.396	**0.418**
	STWV	0.541	0.556	**0.584**
Zulu	MTWV	0.113	0.116	**0.128**
	STWV	0.264	0.265	**0.279**

Table 2: *MTWV/STWV from search combination to decoding system combination*

After combining result from multiple searches, these results are further combined with the result from different decoding systems to achieve even greater improvement, as shown in Table 2. The result is the average MTWV over all languages. We pick the DNN system as our single best system. By search combination, we achieve better performance on all five languages. If we combine the search combination result from other decoding systems, we gain further improvement. This indicates that the improvement from system combination comes from the diversity between systems. Although the BNF system and the PLP system have slightly worse performance compared to the DNN system, combining them nevertheless yields improvement. We also tested doing system combinations in different orders but found out that the order of combination does not have much impact on performance.

6. Analysis

6.1. Analysis of search and query length distribution

During our experiments, we discovered that the improvement from search combinations varies for different languages. On closer inspection, we found that the difference is due to the distributions of query length for each language. Each of the 5 language has around 2000 queries, yet query length distribute differently, as shown in Table 3.

Length	**Assamese**	**Bengali**	**Haitian**	**Lao**	**Zulu**
1	947	926	573	325	1857
2	850	877	953	902	109
3+	162	167	398	698	19

Table 3. *Distribution of query length in five languages*

The queries for Haitian and Lao have relatively low percentages of queries with length 1. On the other hand, the queries for Zulu have extremely high percentage of queries with length one. This distribution is highly correlated with the result showed in Table 1, where it is showed the search combination is more helpful for Haitian and Lao and less beneficial for Zulu. The statistical analysis indicates a significant interaction ($p<0.01$) between query length and search technique.

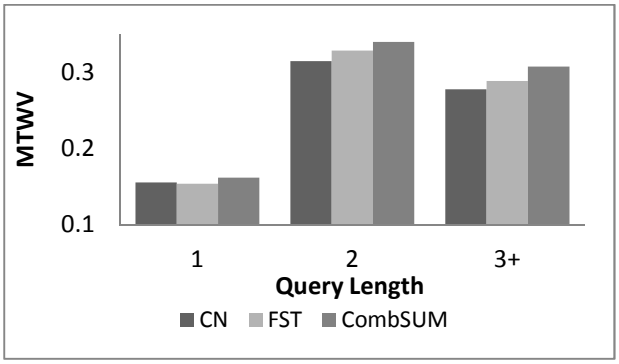

Figure 2: *MTWV interactions for search methods and query length*

Figure 2 shows the interactions between search methods and the query length, averaged over all languages and decoding systems. This analysis yields two findings.

CN search performs somewhat better on queries of length one word, while FST search outperforms CN search on longer queries. As well, CN search has fewer false alarms compared with FST search on the one word queries. This is a consequence of lattice to CN conversion, since hypotheses in the lattice are merged or pruned during the conversion process. The false alarm hypothesis can be pruned, or its probability can be suppressed by other well-recognized hypotheses in the same confusion set. The conversion process does not have too much impact on correct detections, since these are mostly preserved in the CN. As a result, the preserved correct detections and the removed false alarms contribute to better MTWV score. FST search outperforms CN search on multi-word queries. This is because lattices can better preserve history information for decoding hypothesis compare to CN. This observation provides an explanation for the result shown in Figure 3, where FST search outperform CN search on every language except for Zulu. From Table 3, we can see the query set for Zulu is mostly composed of single word queries. We believe the overall difference in MTWV is caused by the imbalanced query set, not by properties of the language.

Search combination provides better improvement on multi-word query, compared with single word query. This matches our finding that the improvement from system combinations comes from the diversity of systems. FST search and CN search use different approaches to search on multi-word queries. This diversity contributes to the consistent improvement over different languages and systems. For the single-word query, since there is less difference on two search approaches, the improvement for system combination is limited due to the lack of diversity. This answers why the search combination has less effect on Zulu. The Zulu query set is mostly single word queries, and there is insufficient diversity between the two different search approaches.

6.2. Analysis between search and decoding systems

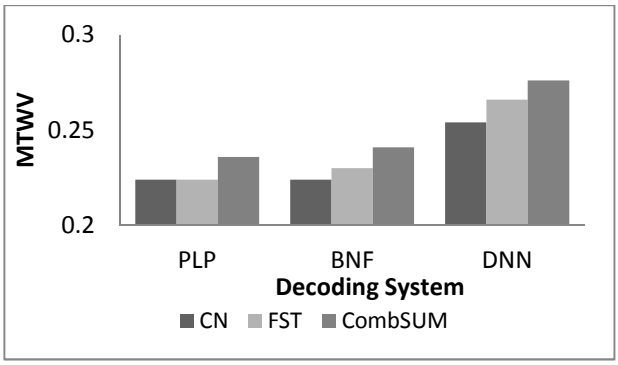

Figure 3: *MTWV interactions for search methods and decoding systems*

Figure 3 shows the interactions between different decoding systems and search methods. The result is the average MTWV over all languages, using different search methods. We have two observations according to this analysis. First, search combination provides consistent improvements across different decoding systems. This indicates the search combination is not sensitive to the properties of decoding systems. Second, the difference on MTWV for CN and FST search is correlated to the performance of the decoding system. The DNN system has the best overall performance on the MTWV, and the difference between FST and CN search is the largest. On the other hand, the PLP system has the worst performance on MTWV among the three systems. The difference between FST and CN search is also the least in our experiment. This suggests that FST and CN search have similar performance on a weaker decoding result, and the difference is larger when a higher quality decoding result is available. But combining the different results can still gain extra improvement, as shown in Section 5.3.

6.3. The higher STWV in CN search

From Table 1, we can see that CN search consistently has higher STWV compared to the FST search. This is because creation of confusion networks gives rise to extra links between words. These links are only available during CN search, and they contribute to the somewhat higher STWV. This phenomenon increases the STWV for the CN system, yet does not have huge impact on the MTWV score. FST search still produces a better MTWV score over multi-word queries.

7. Related Work

Recently, it has been shown that good STD performance can be obtained by combining ASR systems [20, 30, 31]. However, these works focus on combining different ASR systems. Our work shows that even by combining the different search results from the same ASR system, we can achieve better performance. Moreover, if multiple decoding results are available, doing decoding system based combination after search combination can achieve even better results.

8. Conclusion

In this paper, we examine two different STD search approaches, CN search and FST search, using the same ASR output. We find that CN search performs better on single word queries, and that FST search performs better on longer queries. We find that combination of the results of the two different searches achieves better performance than either search approach on its own. Our finding holds across three different decoding systems on five different languages. If we have multiple ASR system available, doing ASR based combination can achieve an even better result; the order of combination does not impact performance.

9. Acknowledgement

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense U.S. Army Research Laboratory (DoD / ARL) contract number W911NF-12-C-0015.[1]

10. References

[1] Fiscus, J. G., Ajot, J., Garofolo, J. S. and Doddingtion, G., "Results of the 2006 Spoken Term Detection Evaluation", Proc. SIGIR, Vol 7, 51-57, 2007

[2] Miller, D. R., Kleber, M., Kao, C. L., Kimball, O., Colthurst, T., Lowe, S. A., Schwartz. R. M and Gish, H., "Rapid and Accurate Spoken Term Detection", Proc. Interspeech, 314-317, 2007

[3] Mamou, J., Ramabhadran, B. and Siohan, O., "Vocabulary Independent Spoken Term Detection", Proc. SIGIR, 615-622, 2007

[4] Vergyri, D., Shafran, I., Stolcke, A., Gadde, V. R. R., Akbacak, M., Roark, B. and Wang, W., "The SRI/OGI 2006 Spoken Term Detection System", Proc. Interspeech, 2393-2396, 2007

[5] Harper. M, "IARPA Solicitation IARPA-BAA-11-02", 2011

[6] Miao, Y. and Metze, F., "Improving Low-Resource CD-DNN-HMM using Dropout and Multilingual DNN Training", Proc. Interspeech, 2237-2241, 2013

[7] Miao, Y., Metze, F. and Rawat, S., "Deep Maxout Networks for Low-Resource Speech Recognition", Proc. Automatic Speech Recognition and Understanding (ASRU), 398-403, 2013

[8] Zhang, X., Trmal, J., Povey, D. and Khudanpur, S., "Improving Deep Neural Network Acoustic Models using Generalized Maxout Networks", To appear, International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014

[9] Rath, S. P., Povey, D., Vesely, K. and Cernocky, J., "Improved Feature Processing for Deep Neural Networks", Proc. Interspeech, 2013

[10] Miao, Y., Zhang, H., and Metze, F., "Distributed Learning of Multilingual DNN Feature Extractors using GPUs", in Proc. Interspeech, 2014. To appear

[11] Miao, Y., Zhang, H., and Metze, F., "Towards Speaker Adaptive Training of Deep Neural Network Acoustic Models", in Proc. Interspeech, 2014. To appear

[12] Miao, Y. and Metze, F., "Improving Language-Universal Feature Extraction with Deep Maxout and Convolutional Neural Networks", in Proc. Interspeech, 2014. To appear

[13] Chen, G., Khudanpur, S., Povey, D., Trmal, J., Yarowsky, D. and Yilmaz, O., "Quantifying the value of pronunciation lexicons for keyword search in low resource languages", Proc. Of International Conference on Acoustics, Speech and Signal Processing (ICASSP), 8560-8564, 2013

[14] Chen, G., Yilmaz, O., Trmal, J., Povey, D. and Khudanpur, S., "Using proxies for OOV keywords in the keyword search task", Proc. Automatic Speech Recognition and Understanding (ASRU), 416-421, 2013

[15] Soto, V., Cooper, E., Mangu, L., Rosenberg, A. and Hirschberg, J., "Rescoring Confusion Networks for Keyword Search", To appear, International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014

[16] Novotney, S., Bulyko, I., Schwartz, R. M., Khudanpur, S. and Kimball, O., "Semi-Supervised Methods for Improving Keyword Search of Unseen Terms", Proc. Interspeech, 3-6, 2012

[17] Zhang, B., Schwartz, R. M., Tsakalidis, S., Nguyen, L. and Matsoukas, S., "White Listing and Score Normalization for Keyword Spotting of Noisy Speech", in Proc. Interspeech, 1832-1835, 2012

[18] Peng, F., Roy, S., Shahshahani, B. and Beaufays, F., "Search Results based N-best Hypothesis Rescoring with Maximum Entropy Classification", Proc. Automatic Speech Recognition and Understanding (ASRU), 422-427, 2013

[19] Wintrode, J., Khudanpur, S., "Can You Repeat That? Using Word Repetition to Improve Spoken Term Detection", To appear, Association for Computational Linguistics (ACL), 2014

[20] Mangu, L., Soltau, H., Kuo, H. K., Kingsbury, B. and Saon, G., "Exploiting Diversity for Spoken Term Detection", Proc. Of International Conference on Acoustics, Speech and Signal Processing (ICASSP), 8282-8286, 2013

[21] Chiu, J. and Rudnicky, A., "Using Conversational Word Burst in Spoken Term Detection", Proc. Interspeech, 2247-2251, 2013

[22] Mangu, L., Brill, E. and Stolcke, A., "Finding Consensus Among Words: Lattice-Based Word Error Minimization", Proc. Eurospeech, 495-498, 1999

[23] Mangu, L., Brill, E. and Stolcke, A., "Finding Consensus in Speech Recognition: Word Error Minimization and Other Applications of Confusion Networks", Computer Speech & Language, 14(4), 373-400, 2000

[24] Bufyko, I., Kimball, O., Siu, M. H., Herrero, J. and Blum, D., "Detection of Unseen Words in Conversational Mandarin", Proc. Of International Conference on Acoustics, Speech and Signal Processing (ICASSP), 5181-5184, 2012

[25] Bulyko, I., Herrero, J., Mihelich, C. and Kimball, O., "Subword Speech Recognition for Detection of Unseen Words", Proc. Interspeech, 2012

[26] Can, D. and Saraclar, M., "Lattice Indexing for Spoken Term Detection", IEEE Transactions on Audio, Speech and Language Processing, 19(8), 2338-2347, 2011

[27] Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., Vesely, K., "The Kaldi Speech Recognition Toolkit", Proc. Automatic Speech Recognition and Understanding (ASRU), 1-4, 2011

[28] Povey, D., Hannemann, M., Boulianne, G., Burget, L., Ghoshal, A., Janda, M., Karafiat, M., Kombrink, S., Motlicek, P., Qian, Y., Riedhammer, K., Vesely, K., Vu, N. T., "Generating Exact Lattices in the WFST Framework", Proc. Of International Conference on Acoustics, Speech and Signal Processing (ICASSP), 4213-4216, 2012

[29] Xu, H., Povey, D., Mangu, L., and Zhu, J., "Minimum Bayes Risk decoding and system combination based on a recursion for edit distance." Computer Speech and Language, 25(4), 802-828, 2011

[30] Mamou, J., Cui, J., Cui, X., Gales, M. J., Kingsbury, B., Knill, K., Mangu, L., Nolden, D., Picheny, M., Ramabhadran, B., Schluter, R., Sethy, A. and Woodland, P. C., "System Combination and Score Normalization for Spoken Term Detection", Proc. of International Conference on Acoustics, Speech and Signal Processing (ICASSP), 8272-8276, 2013

[31] Karakos, D., Schwartz, R., Tsakalidis, S., Zhang, L., Ranjan, S., Ng, T., Hsiao, R., Saikumar, G., Bulyko, I., Nguyen, L., Makhoul, J., Grezl, F., Hannemann, M., Karafiat, M., Szoke, I., Vesely, K., Lamel, L. and Le, V. B., "Score Normalization and System Combination for Improved Keyword Spotting", Proc. Automatic Speech Recognition and Understanding (ASRU), 210-215, 2013