KEYWORD SEARCH USING QUERY EXPANSION FOR GRAPH-BASED RESCORING OF HYPOTHESIZED DETECTIONS

Van Tung Pham[1,2], Haihua Xu[2], Xiong Xiao[2], Nancy F. Chen[3]
Eng Siong Chng[1,2], Haizhou Li[1,2,3]

[1]School of Computer Engineering, Nanyang Technological University, Singapore.
[2]Temasek Laboratories, Nanyang Technological University, Singapore.
[3]Institute for Infocomm Research, Singapore

ABSTRACT

In this work, we propose a novel framework for rescoring keyword search (KWS) detections using acoustic samples extracted from the training data. We view the keyword rescoring task as an information retrieval task and adopt the idea of query expansion. We expand a textual keyword with multiple speech keyword samples extracted from the training data. In this way, the hypothesized detections are compared with the multiple keywords using non-parametric approaches such as dynamic time warping (DTW). The obtained similarity scores are used in a graph based method to re-rank the original confidence scores estimated by the automatic speech recognition (ASR) systems. Experimental results on the NIST OpenKWS15 Evaluation show that our rescoring method is effective, especially for the subword system. For subword experiments, the graph-based rescoring with training samples obtains 5.1% and 1.5% absolute improvement over two baseline systems. One is a standard parametric ASR system, while the other is the graph-based rescoring without training samples.

Index Terms— Spoken term detection, keyword spotting, graph based rescoring, acoustic similarity, query expansion.

1. INTRODUCTION

With the prevalence of smart phone devices and high Internet bandwidth, there is an increasing amount of spoken data to be archived, managed, and analyzed. Speech retrieval is thus an important research area.

One of the primary tasks of speech retrieval is spoken term detection (STD) [1] or keyword search (KWS) [2], which aims to find all occurrences of a textual keyword in a speech corpus. Generally, a two-stage approach is utilized for a KWS system [3–5]. In the first stage, audio files of the speech corpus are automatically segmented and then transformed into transcriptions or lattices by an automatic speech recognition (ASR). At the second stage, retrieval techniques, such as weighted finite state transducer (WFST) [6–8] or Ngram inverted index [9–12], are applied on these lattices to produce a list of detections (posting list or candidate list).

Each detection in the posting list has time information and a confidence score, which is normally the posterior probability of the keyword at the time span [13]. However, such scores might not be robustly estimated in adverse acoustic conditions. Thus it is desirable to use information from various sources, e.g. the ASR lexicon [14–16], ASR lattices [14, 17, 18] or acoustic features [19–22], to enhance the detections scores.

Besides KWS, another major task in the speech retrieval area is Query-by-Example (QbE) [23] which aims to search spoken queries in a speech corpus. The common approach for this task is to use template matching directly in the features space, instead of using an ASR as in KWS. A recent study [24] also showed that such template-based matching (and non-parametric) scores are complementary with the ASR posterior probabilities estimated from the acoustic model and the language model (LM).

Inspired by the success of template matching in the QbE, in this work we propose to use keyword samples extracted from the training data to rescore the KWS results. The main idea is that if a hypothesized detection is acoustically more similar to the keyword samples in the training data, it is more likely to be a true detection, and its score should be boosted. Specifically, for a test keyword, each hypothesized detection can be compared with multiple training samples through non-parametric approaches such as dynamic time warping (DTW); then the obtained similarity scores, together with the similarity between detections themselves, are used in a graph based method to rescore the candidate list returned by the standard parametric automatic speech recognition (ASR) systems.

The key difference of this work from the previous graph-based rescoring [19–22] methods is that we introduce keyword samples extracted from the training data into the graph. Our work can be considered as query expansion [25, 26], where the seed query is augmented with additional source of information (training samples in our case) to improve the retrieval performance. Of course, in this work, training samples are not used to search on the test corpus to produce more detections, but rather used to rescore the candidate list.

The paper is organized as follows. In section 2, we describe the proposed approach. Section 3 presents the experiment setup and evaluation metric for the KWS task. Section 4 shows the experimental results, analysis and discussions. Finally, section 5 concludes our work and discuss potential directions in future.

2. QUERY EXPANSION FOR RESCORING HYPOTHESIZED DETECTIONS

The proposed rescoring approach for the KWS is shown in Figure 1. When presented with a keyword q, the KWS system first searches over the lattices (generated by an ASR) to generate a posting list with confidence scores and timing information. The rescoring system then searches the keyword q in the training data to extract speech samples. With the speech samples, the acoustic similarity between speech segments, either detections or keyword samples, are measured using dynamic time warping. The acoustic similarity is then used to rescore the candidate list through two approaches elaborated in Section 2.2 and Section 2.3.

978-1-4799-9988-0/16/$31.00 ©2016 IEEE

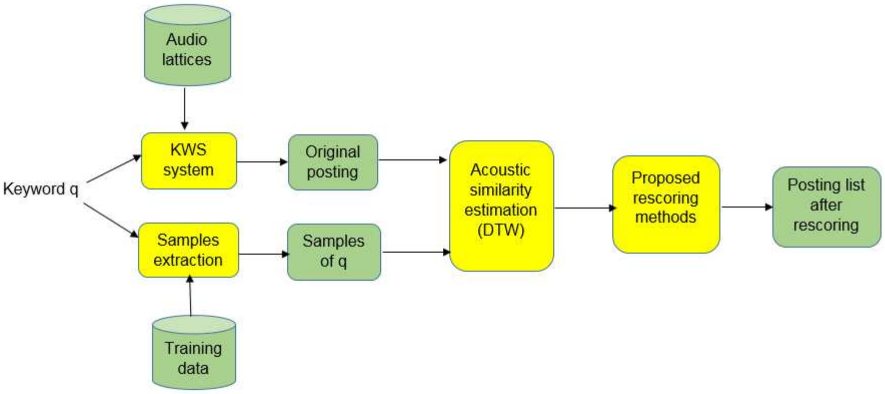

Fig. 1: *Proposed rescoring framework exploiting keyword training samples*

2.1. Keyword samples extraction

Suppose we want to extract samples for a keyword q from the training data. Consider the case where the keyword q contains multiple words; i.e., $q = W_1W_2...W_n$. Given the time boundaries of each word in the training data, the samples for keyword q can be extracted through the following two scenarios:

Case 1: If the whole sequence $W_1W_2...W_n$ appear in the training data, then we extract the whole speech segment at the found locations as samples for q.

Case 2: If only segments of the keyword $q = W_1W_2...W_n$, can be found, we reconstruct q by concatenating samples of W_i, $i = 1..n$. To ensure the quality of the concatenation, the individual samples of the concatenation have to belong to the same gender.

2.2. Rescoring KWS candidates using multiple samples

Let d be a detection of a keyword q, $C(d)$ is the original confidence score of d, i.e. its lattice posterior probability [13]. We define the new score, denoted as $AVG\_SIM(d)$, as the average similarity between d and all samples of the keyword q. $AVG\_SIM(d)$ is calculated as following:

$$AVG\_SIM(d) = \frac{1}{n}\sum_{i=1}^{n} S(d, x_i) \tag{1}$$

where n is the number of samples of q, x_i is the i^{th} sample of q, $S(d, x_i)$ is the acoustic similarity between two speech segments d and x_i, which is estimated using the DTW method in [27]. The acoustic similarity is estimated using multilingual bottle-neck features as described in [28]. The final confidence score $C'(d)$ is then computed as interpolation between the original score $C(d)$ and the new score $AVG\_SIM(d)$:

$$C'(d) = C(d)^{\delta} AVG\_SIM(d)^{1-\delta} \tag{2}$$

where δ is the interpolation factor (which is from 0.2 to 0.4 in our experiments in section 4) tuned using development data.

2.3. Rescoring KWS candidates through graph random walk with samples

Although rescoring by multiple samples is simple, it has some drawbacks. First, the average operator in equation 1 is too simple to describe the acoustic similarity between the detection d and all samples of the keyword q. Second, the rescoring process treats each detection independently, hence ignore similarity information between detections themselves.

In this work we adopt the graph-based re-ranking technique, which was previously used for the rescoring task [19–22]. Previous works [19–22] only used detections from the posting list to construct the graph. Our work is different from the previous works in that the true samples of the keyword extracted from training data are introduced into the graph. The detailed rescoring algorithm is described below.

2.3.1. Graph construction

For a keyword q, a directed graph is constructed from all samples and all detected candidates of the keyword. Each node of the graph represents a speech segment, either corresponding to a detection or a training sample; and the weight between a pair of nodes is the acoustic similarity between the two nodes, produced by the DTW algorithm [27]. The graph is then pruned by k-nearest neighbors criteria, i.e. two node x_i and x_j are connected to each other if x_i is among k highest similarity of x_j or x_j is among k highest similarity of x_i.

2.3.2. Random walk rescoring with samples

Once the graph is constructed, a set of graph-based scores are estimated by algorithms such as random walk [19, 20], or manifold regularization [29]. From our initial results, those approaches produce similar performance, thus we only present the graph random walk in this work. Specifically, let $C(x_i)$ be the initial score of a node x_i, a graph-based score $G(x_i)$ is estimated as

$$\begin{aligned} G(x_i) =& (1-\alpha-\gamma)C(x_i) + \alpha \sum\nolimits_{x_j \in D(x_i)} G(x_j)S'(x_i, x_j) \\ &+ \gamma \sum\nolimits_{x_j \in E(x_i)} G(x_j)S'(x_i, x_j) \end{aligned} \tag{3}$$

where $D(x_i)$ is the set of detections connected to x_i, $E(x_i)$ is the set of training samples that connected to x_i, $0 \leq\alpha$, $\gamma \leq 1$ are factors. $S'(x_i, x_j)$ is the normalized similarity between x_i and x_j computed as follow:

$$S'(x_i, x_j) = \frac{S(x_i, x_j)}{\sum_{x_k \in D(x_j) \cup E(x_j)} S(x_j, x_k)} \tag{4}$$

Table 1: Number of detected keywords and keywords with samples for word and subword systems on $evalpart1$ data set

Systems	Detected keywords	Keywords with samples
Word	1711	1509
Subword	1620	1514

where $D(x_j) \cup E(x_j)$ denotes the set of all nodes connected to x_j. Note that the initial scores $C(x_i)$ of detections are ASR scores, while initial scores of training samples are set to 1.

Intuitively, the equaltion 3 enforce two constraints on the graph-based scores $G(x_i)$: (1) the graph-based scores should not be too different from the initial ASR scores; (2) two nodes, that are acoustically similar to each other, will have similar graph-based scores. The scores $G(x_i)$ can be easily estimated using an iterative method [19, 20] or a closed-form solution [22]. The final confidence score for each detection d is then estimated as

$$C'(d) = C(d)^{\delta} G(d)^{1-\delta} \quad (5)$$

where δ is the interpolation factor

3. EXPERIMENT SETUP

3.1. NIST OpenKWS15 Data

The KWS experiments are conducted on Swahili, which is the surprise languague for the NIST OpenKWS15 Evaluation. The released acoustic data includes 40 hours training data, 10 hours development data (denoted as $dev10h$) and 15 hours of part 1 of evaluation data (denoted as $evalpart1$). The training data is used to develop the ASR systems. The $dev10h$ data is used for parameter tuning of KWS, and we evaluate the KWS performance on $evalpart1$.

Since no lexicon is provided by the OpenKWS15 organizer, we have generated our own grapheme-based lexicon. The text data, which is provided by the organizer, contains 84M words. It is used to establish the lexicon of 350K words in size and also to train the language model.

NIST released 2 keyword lists: one is formulated by the BBN/IBM teams and another is created by NIST for evaluation. We use the BBN/IBM keyword list for parameter tuning, e.g. the parameter δ in equation 5, then report the performance on evaluation keyword list. For the evaluation keyword list, there are 1860 keywords that appear in the $evalpart1$ data set. Since any rescoring method only affects the performance of detected keywords, we evaluate the performance on detected keywords.

Table 1 summarizes the number of detected keywords in the evaluation keyword list for both word and subword systems. We also report the number of keywords that appear in the training speech data. It can be seen that we can find corresponding speech samples in the training data for most of the detected keywords.

3.2. Evaluation metric

To evaluate the KWS performance, NIST defines the term-weighted value (TWV) [1] which integrates the miss rate and false alarm rate (FA) of each keyword into a single metric and then averages over all keywords:

$$TWV(\theta) = 1 - \frac{1}{M}\sum_{k=1}^{M}((P_{miss}(q_k, \theta) + \beta P_{fa}(q_k, \theta)) \quad (6)$$

where θ is a threshold, M is the number of keywords, q_k is a keyword, P_{miss} and P_{fa} are probabilities of miss and FA respectively. The weight β is related with the prior probability of a keyword, and the cost ratio between the false alarm and the miss errors.

Table 2: The MTWV of two proposed methods, i.e. RMS and GBRWS, as compared to raw ASR scores and graph-based rescoring without samples (GBR) on $evalpart1$ data set

Systems	Raw ASR scores	GBR	RMS	GBRWS
Word	0.5616	0.5797	0.5727	**0.5846**
Subword	0.4716	0.5067	0.5028	**0.5224**

Actual term-weighted value (ATWV) is the TWV of a chosen θ, whereas the maximum term-weighted value (MTWV) is the best TWV found over all the possible values θ. The ATWV score is sensitive to the threshold selection thus might lead to uncertainty in comparison between difference experiments. When comparing across ATWVs, it is difficult to know if the difference is caused by different systems or by the threshold selection. Therefore, MTWV is used as evaluation metric. In addition to ATWV and MTWV, NIST also uses a detection error tradeoff (DET) curve to evaluate the performance of a KWS system.

3.3. Keyword Search systems

Two KWS systems have been built: one is a word-based system, and another is a subword-based system. For subword system, we adopt the Morfessor toolkit [30] to segment both word-based dictionary and word transcriptions into subwords, i.e. morpheme units. The open-source Kaldi toolkit [31] is used to build our ASR systems. We used filter-bank features to train a deep neural network (DNN) acoustic model. Both systems use 3gram LM. For indexing and search, we utilized the WFST algorithm [7] which is a part of the Kaldi recipe [31].

4. EXPERIMENT RESULTS AND ANALYSIS

Table 2 shows the MTWV of the proposed methods in section 2.2 and 2.3, i.e. rescoring by multiple samples (denoted as RMS) and graph-based rescoring with training samples (denoted as GBRWS), as compared to the two baselines, i.e. raw ASR scores and graph-based rescoring without samples [19, 20] (denoted as GBR), on $evalpart1$. Their DET curves are also presented in Figures 2 and 3. Note that those KWS performances are obtained after applying the well-known Keyword Specific Thresholding normalization [32, 33]. From Table 2, it can be seen that the proposed rescoring methods are effective, especially for the subword system. With the subword system, RMS provides 3.1% absolute MTWV improvement over the raw ASR scores. The proposed GBRWS provides more gain, which is 5.1% absolute improvement over the raw ASR score, on the subword system. It can be explained that the word-based system might be more robust than the subword system due to longer acoustic evidence of word units, thus the word system benefits less from the rescoring methods. It can also be seen that the GBRWS outperforms the conventional GBR for both word and subword systems. For the subword system, 1.6% absolute improvement is observed on the $evalpart1$ data set.

From DET curves in Figure 2 and 3, we have same observations: (1) both proposed methods outperform significantly the raw ASR score across the board, especially for the subword-based system; (2) the GBRWS outperforms other methods for both word and subword systems. It is also worth noting that the proposed query expansion methods are more effective on low false alarm region than on low miss rate region.

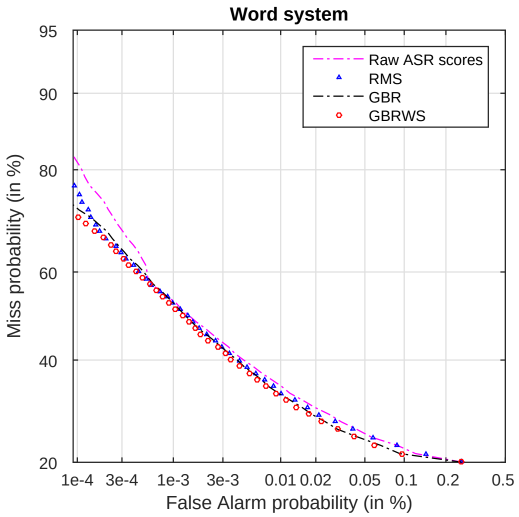

Fig. 2: The DET curves of two proposed methods, RMS and GBRWS, as compared to raw ASR score and GBR for the word system on $evalpart1$ data

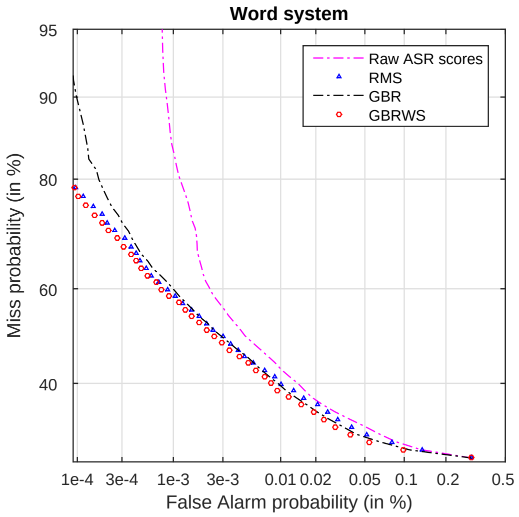

Fig. 3: The DET curves of two proposed methods, RMS and GBRWS, as compared to raw ASR score and GBR for the subword system on $evalpart1$ data

We conduct further analysis on the miss rate and FA rate at the optimal threshold, i.e. at the point corresponding to MTWV scores, of $evalpart1$ data to know which type of keywords are benefited by our proposed methods. Three types of keywords are taken into consideration, i.e. short keywords (single-word keywords with no more than 6 phones), medium keywords (single-word keywords with more than 6 phones or two-words keywords) and long keywords (remaining keywords).

Figure 4 shows the miss and FA rate of the raw ASR score and the proposed GBRWS method on word and subword systems for three types of keywords. From the figure, it can be seen that the GBRWS is effective on the short keyword for both word and subword systems, especially for reducing the FA rate a lot. For medium and long keywords, the graph-based rescoring method helps to reduce miss rate considerably with the cost of slightly increased the FA rate. This observation suggests that the proposed method is more helpful for short keywords.

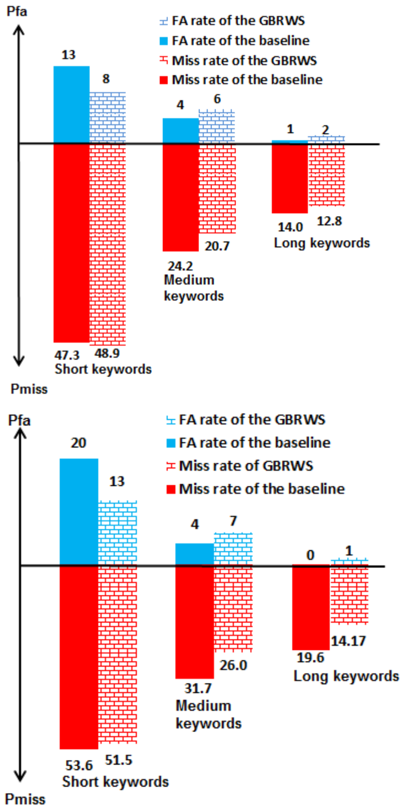

Fig. 4: The miss and FA rate of the baseline raw ASR score and the proposed GBRWS method on word (top figure) and subword (bottom figure) systems for three types of keywords (short, medium, and long keywords) on $evalpart1$ data

5. CONCLUSION AND FUTURE WORK

We proposed keyword query expansion as a novel framework that exploits keyword samples in the training data. The acoustic similarity between detections and training samples are used to rescore the hypothesized detections through multiple samples or a graph-based rescoring technique. We observe empirical gains on the NIST OpenKWS15 Swahili data, especially for subword systems and short keywords.

This work is only applicable for seen-word keywords, but in principle it can be applied to unseen-word keywords as long as their subword representations are available in the training data. Thus, for future work, we plan to improve our approach by generating training samples through concatenating subword samples. In addition, we are investigating how to further categorize the training samples to discard those with low audio quality.

6. ACKNOWLEDGMENT

This work is supported by the DSO funded project MAISON DSOCL14045, Singapore.

7. REFERENCES

[1] NIST, “The spoken term detection (std) 2006 evaluation plan,” in *http://www.nist.gov/speech/tests/std*, 2006.

[2] NIST, “The open keyword search (std) 2013 evaluation,” in *http://www.nist.gov/itl/iad/mig/upload/OpenKWS13-evalplan-v4.pdf*, 2013.

[3] N. F. Chen, S. Sivadas, B. P. Lim, H. G. Ngo, H. Xu, V. T. Pham, B. Ma, and H. Li, “Strategies for vietnamese keyword search,” in *Proceedings of ICASSP*, 2014.

[4] N.F. Chen, C. Ni, S. Sivadas, V. T. Pham, H. Xu, X. Xiao, T. S. Lau, S. J. Leow, B. P. Lim, C. C. Leung, L. Wang, C. H. Lee, A. Goh, E. S. Chng, B. Ma, and H. Li, “Low-resource keyword search strategies for tamil,” in *Proceedings of ICASSP*, 2015.

[5] N. F. Chen, V. T. Pham, H. Xu, X. Xiao, V. H. Do, C. Ni, I. Chen, S. Sivadas, C. H. Lee, E. S. Chng, B. Ma, and H. Li, “Exemplar-inspired strategies for low-resource spoken keyword search in swahili,” in *Proceedings of ICASSP*, 2016.

[6] C. Allauzen, M. Mohri, and M. Saraclar, “General Indexation of Weighted Automata Application to Spoken Utterance Retrieval,” in *Proceedings of HLT*, 2004.

[7] D. Can and M. Saraclar, “Lattice Indexing for Spoken Term Detection,” *IEEE Trans. Speech Audio Process*, vol. 19, no. 8, 2011.

[8] S. Parlak and M. Saraclar, “Spoken term detection for turkish broadcast news,” in *Proceedings of ICASSP*, 2008.

[9] M. Saraclar and R. Sproat, “Lattice-Based Search for Spoken Utterance Retrieval,” in *Proceedings of HLT-NAACL*, 2004.

[10] C. Chelba and J. Silva, “Soft indexing of speech content for search in spoken documents,” *Computer Speech and Language*, vol. 21, no. 3, pp. 458–478, 2007.

[11] J. Mamou, D. Carmel, and R. Hoory, “Spoken document retrieval from call-center conversation,” in *Proceedings of SIGIR*, 2006.

[12] K. Thambiratnam and S. Sridharan, “Rapid yet accurate speech indexing using Dynamic Match Lattice Spotting,” *IEEE Trans. Speech Audio Process*, vol. 15, no. 1, 2007.

[13] G. Evermann and P.C. Woodland, “Large vocabulary decoding and confidence estimation using word posterior probabilities,” in *Proceedings of ICASSP*, 2000.

[14] V. Soto, L. Mangu, A. Rosenberg, and J. Hirschberg, “A comparison of multiple methods for rescoring keyword search lists for low resource languages,” in *Proceedings of Interspeech*, 2014.

[15] J. Tejedor, D. T. Toledano, D. Wang, S. King, and J. Colas, “Feature analysis for discriminative confidence estimation in spoken term detection,” *Speech Communication*, vol. 28, no. 5, 2014.

[16] O. Vinyals and S. Wegmann, “Chasing the metric: Smoothing learning algorithms for keyword detection,” in *Proceedings of ICASSP*, 2014.

[17] V. T. Pham, H. Xu, N. F. Chen, S. Sivadas, B. P. Lim, E. S. Chng, and H. Li, “Discriminative score normalization for keyword search decision,” in *Proceedings of ICASSP*, 2014.

[18] M. Seigel, P. Woodland, and M. Gales, “A confidence-based approach for improving keyword hypothesis scores,” in *Proceedings of ICASSP*, 2013.

[19] H. Y. Lee, Y. Zhang, E. Chuangsuwanich, and J. Glass, “Graph-based re-ranking using acoustic feature similarity between search results for spoken term detection on low-resource,” in *Proceedings of ICASSP*, 2013.

[20] H. Y. Lee and L. S. Lee, “Improved semantic retrieval of spoken content by document/query expansion with random walk over acoustic similarity graphs,” in *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2013, vol. 22, pp. 80–94.

[21] Y. N. Chen, C. P. Chen, H. Y. Lee, C. Chan, and L. S. Lee, “Improved spoken term detection with graph-based re-ranking in feature space,” in *Proceedings of ICASSP*, 2011.

[22] Atta Norouzian, Richard C. Rose, Sina Hamidi Ghalehjegh, and Aren Jansen, “Zero resource graph-based confidence estimation for open vocabulary spoken term detection,” in *Proceedings of ICASSP*, 2013.

[23] MediaEval, “The 2015 query by example search on speech task (quesst),” in *http://multimediaeval.org/mediaeval2015/quesst2015/index.html*, 2015.

[24] Haihua Xu, Peng Yang, Xiong Xiao, Lei Xie, Cheung-Chi Leung, Hongjie Chen, Jia Yu, Hang Lv, Lei Wang, Su Jun Leow, Bin Ma, Eng Siong Chng, and Haizhou Li, “Language independent query-by-example spoken term detection using n-best phone sequences and partial matching,” in *Proceedings of ICASSP*, 2015.

[25] C. Manning, P. Raghavan, and H. Schtze, *An introduction to Information Retrieval*, Cambridge University Press., 2008.

[26] Luis M. de Campos, Juan M. Fernandez, and Juan F. Huete, “Query Expansion in Information Retrieval Systems using a Bayesian Network-Based Thesaurus,” in *Fourteenth conference on Uncertainty in artificial intelligence*, 1998.

[27] Luis J. Rodriguez-Fuentes, Amparo Varona, Mikel Penagarikano, German Bordel, and Mireia Diez, “High-performance query-by-example spoken term detection on the sws 2013 evaluation,” in *Proceedings of ICASSP*, 2014.

[28] Van Tung Pham, Haihua Xu, Tze Yuang Chong, Xiong Xiao, Eng Siong Chng, , and Haizhou Li, “On the study of very low-resource language keyword search,” in *Proceedings of APSIPA*, 2015.

[29] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, , and Bernhard Scholkopf, “Learning with local and global consistency,” in *Advances in Neural Information Processing Systems 16*. IEEE, 2004.

[30] Mathias Creutz and Krista Lagus, “Unsupervised discovery of morphemes,” in *In Proceedings of the Workshop on Morphological and Phonological Learning of ACL-02*, 2002.

[31] D. Povey et.al, “The kaldi speech recognition toolkit,” in *Proceedings of ASRU*, 2011.

[32] D. Miller et.al., “Rapid and accurate spoken term detection,” in *Proceedings of Interspeech*, 2007.

[33] D. Karakos et.al., “Score normalization and system combination for improved keyword spotting,” in *Proceedings of ASRU*, 2013.