EXEMPLAR-INSPIRED STRATEGIES FOR LOW-RESOURCE SPOKEN KEYWORD SEARCH IN SWAHILI

Nancy F. Chen[1], *Van Tung Pham*[2], *Haihua Xu*[2], *Xiong Xiao*[2], *Van Hai Do*[2], *Chongjia Ni*[1], *I-Fan Chen*[3], *Sunil Sivadas*[1], *Chin-Hui Lee*[3], *Eng Siong Chng*[2], *Bin Ma*[1], *Haizhou Li*[1]

[1]Institute for Infocomm Research, A*STAR, Singapore
[2]Nanyang Technological University, Singapore
[3]Georgia Institute of Technology, USA
nfychen@i2r.a-star.edu.sg

ABSTRACT

We present exemplar-inspired low-resource spoken keyword search strategies for acoustic modeling, keyword verification, and system combination. This state-of-the-art system was developed by the SINGA team in the context of the 2015 NIST Open Keyword Search Evaluation (OpenKWS15) using conversational Swahili provided by the IARPA Babel program. In this work, we elaborate on the following: (1) exploiting exemplar training samples to construct a non-parametric acoustic model using kernel density estimation at test time; (2) rescoring hypothesized keyword detections through quantifying their acoustic similarity with exemplar training samples; (3) extending our previously proposed system combination approach to incorporate prosody features of exemplar keyword samples.

Index Terms— Spoken term detection (STD), keyword spotting, under-resourced languages, deep neural network (DNN), large vocabulary continuous speech recognition (LVCSR), automatic speech recognition (ASR)

1. INTRODUCTION

Spoken keyword search (KWS) is a detection task where the goal is to find all occurrences of an orthographic term (e.g., word or phrase) from audio recordings. Approaches to spoken keyword search are often LVCSR-based, which follow the transcribe-and-search paradigm. For resource rich languages such as Arabic, English, and Mandarin, high performance is readily achieved with abundant training data [1]. However, for low resource languages (e.g., Vietnamese, Tamil, Swahili), it is more challenging, since LVCSR requires large amounts of training data to obtain good performance. Such challenges have led to initiatives such as the NIST Open Keyword Search Evaluation since 2013 and the IARPA Babel program: "... to rapidly develop speech recognition capability for keyword search in a previously unstudied language, working with speech recorded in a variety of conditions with limited amounts of transcription."

Approaches for low-resource keyword search can be categorized into two types. The first approach indirectly addresses the problem by improving speech recognition or keyword search performance in general. Classic approaches include acoustic feature selection and extraction [2, 3], keyword verification (rescoring) [4, 5], score normalization [6, 7], and system combination (fusion) [6, 7]. This latter approach is usually more popular, because once a baseline LVCSR-based KWS system is set up, just by altering the input (acoustic features) and/or output (posterior scores), one can efficiently obtain gains effectively.

The second approach for low-resource keyword search tackles the data sparsity problem directly at various levels such as data selection or active learning [8], data augmentation [9], linguistically augmented modeling [10], or data-efficient training [11, 12, 13].

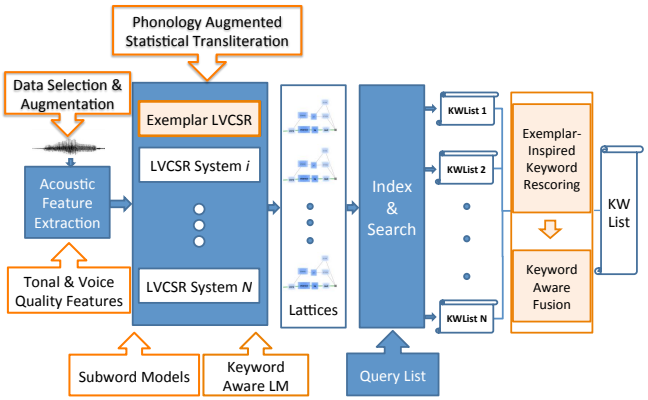

Fig. 1. SINGA spoken keyword search system for low-resource languages. Blocks filled with light orange are highlights discussed in this work: Exemplar LVCSR, Exemplar-Inspired Keyword Rescoring, Keyword Aware Fusion. Details for the other modules can be found in other publications [3, 8, 10].

In this work, we focus on data-efficient training. Existing approaches include parsimonious modeling with fewer parameters (e.g., sharing parameters as in subspace Gaussian mixture model [11]) or using the limited training data resourcefully (e.g., using keyword information in acoustic and language models [12, 14]). The strategies presented in this work resemble the latter approach, but we focus on exploiting *exemplar training samples* at the modeling level and the feature level.

For low-resource languages, it makes intuitive sense to also consider non-parametric approaches such as exemplar processing, since parametric models rely heavily on copious amounts of training data. Recent query-by-example speech retrieval evaluations have also shown that template-based matching and non-parametric scores are complementary to standard ASR posterior scores [15]. Therefore, in this work, we focus on exemplar-inspired features or techniques that are complementary to standard LVCSR-based keyword search. In particular, we exploit exemplar samples from the training data to efficiently construct a non-parametric acoustic model using kernel density estimation at test time (zero training); we rescore hypothesized keyword detections through quantifying their acoustic similarity with exemplar training samples (zero training); we also extend our previously proposed system combination

approach to incorporate a richer set of features, including prosody of exemplar keyword samples to reduce false alarms. A system diagram of our keyword search system is shown in Figure 1.

2. RELATION TO PRIOR WORK

In this section, we discuss how the proposed exemplar-inspired approaches relate to previous work in acoustic modeling, keyword verification, and system combination.

2.1. Exemplar LVCSR: Kernel Density Acoustic Modeling

In the context of the annual NIST Open Keyword Search Evaluations held since 2013, to date all submitted systems are based on parametric LVCSR. While there has been efforts in integrating the advantages of filler-based keyword spotting detection into the LVCSR framework (e.g., keyword aware language modeling [14]), to date no system has exploited non-parametric techniques such as exemplar-based processing [16, 17, 18]. In this work, we extend exemplar-based acoustic models [19, 20] in English speech recognition to the keyword search task for low-resource spoken languages.

2.2. Graph-Based Keyword Rescoring Exploiting Exemplars

Most common approaches for keyword search rescoring (also referred to as keyword verification or score normalization) use machine learning classification with two main research directions. One is making the training objective directly related to keyword search performance [21], while the other focuses on feature selection [22].

In this work, we view the keyword rescoring task as an information retrieval task and adopt the spirit of query expansion. Instead of considering the keyword query in orthographic form, we exploit the keyword exemplar samples in various acoustic forms in the training data. By quantifying and ranking the acoustic similarities between exemplar training samples and hypothesized detections, we expect the detections more similar to exemplars to be more reliable, and thus should be re-ranked with higher confidence.

2.3. Keyword Aware System Combination

Keyword search system combination (also referred to as fusion) approaches such as CombMax, CombSum, CombMNZ [23] are system and keyword independent; each system and each keyword are treated equally. Methods such as WCombMNZ have shown to be more effective [6, 24] because individual systems are weighed according to their keyword search performance prior to combination.

In this work, we further extend our previous work [24] to include a richer set of features that are keyword-specific at the phonological and prosodic levels. Our previous work [24] only included limited phonetic information about the keywords, but prosodic characteristics could also be useful, as spoken terms with high speaking rate are more likely to be false alarms [25]. Not only do we consider phonological and prosodic information from keyword detections in the test set, we also take advantage of the prosodic characteristics from exemplar keyword samples in the training data.

3. EXEMPLAR-INSPIRED KWS STRATEGIES

3.1. Kernel Density Acoustic Modeling

In kernel density acoustic modeling, instead of training parameters for estimating the emission probability in a standard HMM system, exemplar training samples are directly used:

$$\hat{P}(\mathbf{O_t}|s_j) = \frac{1}{Z_j N_j} \sum_{i=1}^{N_j} \exp(-\frac{||\mathbf{O}_t - \mathbf{e}_{ij}||^2}{\sigma}), \quad (1)$$

where $\mathbf{O_t}$ is the feature vector at frame t in the test data; $\mathbf{e_{ij}}$ is the i^{th} exemplar training sample belonging to class j (a tied triphone state); N_j is the total number of exemplars in class j; Z_j is a normalization constant to ensure Eq. (1) is a valid distribution; and σ is used to control the scale of the Gaussians, determining the smoothness of the distribution.

In this work, we only intend to replace the standard acoustic model with an exemplar-based kernel density estimator in a standard DNN LVCSR system. The scores of an exemplar-based acoustic model could differ drastically from a standard DNN acoustic model in terms of dynamic range. Therefore, it is essential to calibrate the likelihood scores obtained in Eq. (1) to reach optimal performance during decoding.

3.2. Graph-Based Keyword Rescoring Exploiting Exemplars

A more detailed description of the method is presented in [26]. We only highlight the main points below to save space.

3.2.1. Graph Construction

For a given keyword k, if a detection d is more acoustically similar to the exemplar keyword samples in the training data, d is more likely to be a true detection, and therefore should be given a higher confidence. We can construct a graph where the nodes are detections or exemplar training samples, while an edge between two nodes represent the acoustic similarity between the two nodes.

3.2.2. Random Walk Rescoring

The graph-based scores can be estimated using random walk. Let $C(x_i)$ be the raw ASR confidence score for the node x_i, and the rescored graph-based score is $R(x_i)$:

$$R(x_i) = (1-\alpha-\beta)C(x_i) + \alpha \sum_{x_j \in D(x_i)} R(x_i)S'(x_i, x_j) + \beta \sum_{x_j \in E(x_i)} R(x_i)S'(x_i, x_j) \quad (2)$$

where the initial scores $C(x_i)$ are set to 1 if x_i is an exemplar training sample instead of a detection, $D(x_i)$ is the detections connected to x_i, $E(x_i)$ is the set of exemplar training samples connected to x_i, α, β in $[0, 1]$, and $S'(x_i, x_j)$ is the normalized acoustic similarity between x_i and x_j:

$$S'(x_i, x_j) = \frac{S(x_i, x_j)}{\sum_{x_k \in D(x_j) \cup E(x_j)} S(x_j, x_k)} \quad (3)$$

The final confidence score for each detection d is:

$$C'(d) = C(d)^{\delta} R(d)^{1-\delta} \quad (4)$$

where δ in $[0, 1]$.

3.3. Keyword Aware System Combination

For machine learning-based system combination, the raw ASR confidence scores are by default used as features. In this work, we also include rescored confidences that have been reported in the literature (see Table 1). In addition, we include keyword-specific features at the phonology and prosody levels and extract them from the keyword queries, hypothesized detections, and exemplar training samples.

Every detection d that belongs to the same keyword query would share the same keyword-specific features. For the j^{th} detection $d_j(i)$ in system i, a feature vector from Table 1 is generated for training a system combination classifier with two class labels: (1) false alarm, and (2) true detection.

Table 1. Features for proposed keyword aware system combination.

Category	Feature
ASR Confidence Score	Raw ASR posterior probability
	Keyword specific thresholding (KST) score [7]
	KST decision
	Sum-to-one (STO) score [6]
	β-STO score [4]
	pFA score [27]
	pFA-KST score [27]
	Rank of detection [24]
Phonology	Location of keyword detection in utterance
	Number of words in keyword query
	Number of vowels in keyword query
	Number of consonants in keyword query
Prosody	Duration of keyword detection
	Speaking rate of keyword detection
	Duration stats of keyword exemplars
	Speaking rate stats of keyword exemplars

4. EXPERIMENTS

For clarity purposes, we only show a subset of submitted systems for OpenKWS15 (Section 4.2 and Section 4.3, and selected post-evaluation experiments and analysis (Section 4.4 and Section 4.5) to demonstrate the proposed strategies discussed in this work.

4.1. Setup

4.1.1. NIST OpenKWS15 Swahili Corpus

This effort uses the IARPA Babel Program Swahili language collection release IARPA-babel202b-v1.0d for the NIST OpenKWS15 Evaluation[1]. The training set includes 40 hours of conversational telephone speech. The Very Limited Language Pack (VLLP) provides word transcriptions for a 3-hour subset of this training audio, a 3-hour tuning set, and a 10-hr developmental set. The evaluation set is 75 hours with no transcriptions nor timing information; transcriptions of a 15-hour subset (*evalpart1*) was released after OpenKWS15. All results reported are on *evalpart1*.

In the NIST OpenKWS15 Evaluation, no pronunciation lexicon was provided, but text data from the web was. The web resources included crawled websites, Wikipeida and Wiktionary, and OpenSubtitles from movies and TV shows.

[1]http://www.nist.gov/itl/iad/mig/openkws15.cfm

In addition to web resources, audio, transcriptions and pronunciation lexicons for six development languages (Cantonese, Pashto, Tagalog, Tamil, Turkish, Vietnamese) were also provided for multilingual training. Transcribed data for each development language ranged from 69.3 to141.3 hours, reaching a total of 538.4 hours.

4.1.2. Evaluation Metric

Term-weighted value (TWV) is 1 minus the weighted sum of the term-weighted probability of miss detection $P_{\text{miss}}(\theta)$ and the term-weighted probability of false alarm $P_{\text{FA}}(\theta)$:

$$\text{TWV}(\theta) = 1 - [P_{\text{miss}}(\theta) + \beta P_{\text{FA}}(\theta)], \quad (5)$$

where θ is the decision threshold. Actual term-weighted value (ATWV) is the TWV using the chosen decision threshold, whereas the maximum term-weighted value (MTWV) is the best TWV found over all θ.

4.2. DNN LVCSR Baseline System

All systems were developed using Kaldi [28]. We adopted voice activity detection (VAD) in [3]. We trained shared-hidden layer multilingual DNN (SHL-MDNN) and multilingual stacked bottleneck features as in [29]. Six development languages (Cantonese, Pashto, Tagalog, Tamil, Turkish, Vietnamese) were used to train the bottleneck feature extractor; fine-tuning with the 3-hr Swahili target data was done on the 2nd bottleneck neural network while freezing the 1st bottleneck neural network. The SHL-MDNN consists of 5 shared hidden layers (each with 2048 nodes) trained by the same six development languages mentioned above, and the output softmax layer is fine-tuned with the 3-hr Swahili target data.

Since no pronunciation lexicon was provided, 1-letter graphemes were used as phonetic units to specify pronunciation. Words extracted from the crawled web data and OpenSubtitles were also added to the lexicon, resulting in the final size of 350k. A trigram language model was trained on the word tokens, including those from the web resources. Deterministic weighted transducers were used to index and search soft-hits, which contain the utterance identifications, start/end times, and posterior scores. Sum-to-one normalization [30], WCombMNZ [30], and keyword specific thresholding (KST) [7] were applied consecutively to combine systems. For individual systems, only KST was done.

Two DNN systems were trained. The first DNN is a standard LVCSR system using words as the lexical and decoding units in the pronunciation dictionary and language model, respectively. The second DNN system is a variant of the first one, where the lexical and decoding units are subwords, converting all word units to automatically parsed morphemes as in [8]. Table 2 and Table 3 list the ATWV and MTWV results for the DNN word and subword systems respectively.

4.3. Exemplar LVCSR: Kernel Density Based Acoustic Model

4.3.1. Implementation Details

Setup is the same in Section 4.2 except that we replace the DNN acoustic model with the kernel density estimation exemplar model. The raw MFCC features x_t from Swahili are first used to train an HMM/GMM acoustic model of clustered triphone states. This HMM/GMM model is used to generate frame-level tied-state labels (s_j in Eq. (1)) for the entire training data through forced alignment. Previous work has shown that using features such as MFCC is insufficient to obtain competitive performance when using kernel density

Table 2. Word-based keyword search performance.

System ID	Approach	ATWV	MTWV
W1	DNN baseline	0.4320	0.4336
W2	Exemplar	0.4403	0.4420
W1r	W1 + graph rescoring	0.4500	0.4596
W2r	W2 + graph rescoring	0.4554	0.4670
W3=W1+W2	WCombMNZ	0.4564	0.4571
W4=W1+W2	Proposed Fusion	0.4778	0.4850
W3r=W1r+W2r	WCombMNZ	0.4667	0.4667
W4r=W1r+W2r	Proposed Fusion	0.4885	0.4915

Table 3. Subword-based (morpheme) keyword search performance.

System ID	Approach	ATWV	MTWV
M1	DNN baseline	0.4230	0.4263
M2	Exemplar	0.4340	0.4356
M1r	W1 + graph rescoring	0.4355	0.4517
M2r	W2 + graph rescoring	0.4558	0.4693
M3=M1+M2	WCombMNZ	0.4505	0.4505
M4=M1+M2	Proposed Fusion	0.4766	0.4805
M3r=M1r+M2r	WCombMNZ	0.4597	0.4624
M4r=M1r+M2r	Proposed Fusion	0.4840	0.4889

estimation [20]. Superior acoustic features are critical in successfully exploiting kernel density estimation. In this work, we use cross-lingual bottleneck features generated from a DNN bottleneck feature extractor described in Section 4.2. We also apply fMLLR adaptation to the bottleneck features to further reduce speaker variability and compensate for channel mismatch. Euclidian distance is used in the exponential function in Eq (1). Score calibration was needed to adjust the dynamic range of acoustic scores so that we can readily plug them into the standard language model in Section 4.2. A neural network with 1 hidden layer was thus trained for score calibration (setup similar to [20]). Pruning was also done for efficient decoding to meet the evaluation timeline.

4.3.2. Results

Table 2 and Table 3 list the results. We see that the proposed exemplar system performs similarly to its DNN counterpart for the word and subword model experiments. When the proposed exemplar system is combined with its DNN counterpart, relative fusion gains reach 5.6% and 6.5% for the word and subword experiments respectively when we use the WCombMNZ baseline approach to combine systems (W3 in Table 2 and M3 in Table 3) when considering ATWV.

4.4. Graph-Based Keyword Rescoring Exploiting Exemplars

Due to space constraints, we only highlight a subset of our analysis; detailed comparisons for different experimental settings and variants of the proposed approach can be found in [26].

Graph-based rescoring using exemplar keywords was performed on the DNN and Exemplar models for both the word-based systems (W1r and W2r in Table 2) and subword-based systems (M1r and M2r in Table 3). To disentangle threshold settings and algorithmic improvements, we discuss MTWV improvements as is customary in prior work [6], though ATWV results are still provided for interested readers. The DNN model improved 6.0 % relative for both word-based and subword-based systems. The exemplar model improved 5.7% and 7.7% relative for the word-based and subword-based systems, respectively.

4.5. Keyword Aware System Combination

4.5.1. Implementation Details

A neural network with 1 hidden layer and 200 hidden nodes was trained on the developmental set and a self-derived development keyword list. The development keyword list was generated by randomly sampling N-grams of consecutive words in the training and developmental set that satisfy the distributions in Table 4 and Table 5.

Table 4. N-gram frequency distribution of self-generated development keyword list.

N	1	2	3	4	5
Frequency	0.5	0.4	0.075	0.02	0.005

Table 5. Occurrence frequency of self-generated development keyword list.

Occurrence	1	2	3	$4-10$	>10
Frequency	0.4	0.25	0.15	0.12	0.08

4.5.2. Results

From Table 2 and Table 3, we observe that the proposed keyword aware system combination led to relative improvements ranging from 4.6 to 5.2 % for ATWV and 5.3 - 5.7% for MTWV, when compared to the respective WCombMNZ baselines. We also see similar trends of consistent gains (ranging from 2.1 to 4.3% relative) when using the proposed keyword aware system combination method for fusing 14 subsystems for the VLLP condition for OpenKWS15 (Swahili), 8 subsystems for the FLP condition for OpenKWS15 (Swahili), 7 subsystems for the LLP (limited language pack; 10 hours of transcriptions) condition for OpenKWS14 (Tamil), and 5 subsystems for the VLLP condition for Vietnamese (OpenKWS13)[2].

5. DISCUSSION

In this work, we apply exemplar-inspired techniques and features to improve low-resource keyword search performance. We demonstrated that kernel density based LVCSR exploiting exemplar training samples for acoustic modeling is comparable to and complements DNN LVCSR, incorporating prosodic features of exemplar keywords improves system combination performance, and rescoring detections using exemplars through random walk is effective for subwords and short keywords. Since these techniques do not rely on any linguistic peculiarities, we expect them to also generalize well to other languages besides Swahili.

[2]The subsystems used for Tamil and Vietnamese are subsets or variants of those reported in [3, 8]

6. REFERENCES

[1] Jonathan G Fiscus, Jerome Ajot, John S Garofolo, and George Doddingtion, "Results of the 2006 spoken term detection evaluation," in *ACM SIGIR Workshop on Searching Spontaneous Conversational*, 2007, pp. 51–55.

[2] Florian Metze, Zaid A. W. Sheikh, Alex Waibel, Jonas Gehring, Kevin Kilgour, Quoc Bao Nguyen, and Van Huy Nguyen, "Models of tone for tonal and non-tonal languages," in *Proc. IEEE ASRU*, 2013.

[3] Nancy F Chen, Sunil Sivadas, Boon Pang Lim, Hoang Gia Ngo, Haihua Xu, Van Tung Pham, Bin Ma, and Haizhou Li, "Strategies for Vietnamese keyword search," in *Proc. IEEE ICASSP*, 2014, pp. 4121–4125.

[4] Victor Soto, Lidia Mangu, Andrew Rosenberg, and Julia Hirschberg, "A comparison of multiple methods for rescoring keyword search lists for low resource languages," *Interspeech*, pp. 2464–2468, 2014.

[5] Hung-yi Lee, Yu Zhang, Ekapol Chuangsuwanich, and James Glass, "Graph-based re-ranking using acoustic feature similarity between search results for spoken term detection on low-resource languages," *Interspeech*, pp. 2479–2483, 2014.

[6] Jonathan Mamou, Jia Cui, Xiaodong Cui, Mark JF Gales, Brian Kingsbury, Kate Knill, Lidia Mangu, David Nolden, Michael Picheny, Bhuvana Ramabhadran, et al., "System combination and score normalization for spoken term detection," in *ICASSP*, 2013, pp. 8272–8276.

[7] Damianos Karakos, Richard Schwartz, Stavros Tsakalidis, Le Zhang, Shivesh Ranjan, Tim Ng, Roger Hsiao, Guruprasad Saikumar, Ivan Bulyko, Long Nguyen, et al., "Score normalization and system combination for improved keyword spotting," in *Proc. IEEE ASRU*, 2013, pp. 210–215.

[8] Nancy F Chen, Chongjia Ni, I-Fan Chen, Sunil Sivadas, Van Tung Pham, Haihua Xu, Xiong Xiao, Tze Siong Lau, Su Jun Leow, Boon Pang Lim, et al., "Low-Resource Keyword Search Strategies for Tamil," in *Proc. ICASSP*, 2015.

[9] Xiaodong Cui, Vikas Goel, and Brian Kingsbury, "Data augmentation for deep neural network acoustic modeling," in *ICASSP*, 2014, pp. 5582–5586.

[10] Hoang Gia Ngo, Nancy F Chen, Binh Minh Nguyen, Bin Ma, and Haizhou Li, "Phonology-augmented statistical transliteration for low-resource languages," in *Interspeech*, 2015.

[11] Daniel Povey, Lukas Burget, Mohit Agarwal, Pinar Akyazi, Kai Feng, Arnab Ghoshal, Ondrej Glembek, Nagendra K Goel, Martin Karafiát, Ariya Rastrow, R. C. Rose, P Schearz, and S. Thomas, "Subspace Gaussian mixture models for speech recognition," in *Proc. IEEE ICASSP*, 2010, pp. 4330–4333.

[12] I-Fan Chen, Nancy F Chen, and Chin-Hui Lee, "A keyword-boosted smbr criterion to enhance keyword search performance in deep neural network based acoustic modeling," in *Interspeech*, 2014.

[13] I-Fan Chen, Chongjia Ni, Boon Pang Lim, Nancy F Chen, and Chin-Hui Lee, "A Keyword-Aware Grammar Framework for LVCSR-Based Spoken Keyword Search," in *ICASSP*, 2015.

[14] I-Fan Chen, Chongjia Ni, Boon Pang Lim, Nancy F Chen, and Chin-Hui Lee, "A Keyword-Aware Language Modeling Approach to LVCSR-based Keyword Search," *Journal of Signal Processing Systems*, 2015.

[15] Haihua Xu et. al., "Language independent query-by-example spoken term detection using n-best phone sequences and partial matching," in *ICASSP*, 2015.

[16] Tara N Sainath, Bhuvana Ramabhadran, David Nahamoo, Dimitri Kanevsky, Dirk Van Compernolle, Kris Demuynck, Jort Florent Gemmeke, Jerome R Bellegarda, and Shiva Sundaram, "Exemplar-based processing for speech recognition: An overview," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 98–113, 2012.

[17] John Labiak and Karen Livescu, "Nearest neighbors with learned distances for phonetic frame classification.," in *INTERSPEECH*, 2011, pp. 2337–2340.

[18] Natasha Singh-Miller and Michael Collins, "Learning label embeddings for nearest-neighbor multi-class classification with an application to speech recognition," in *Advances in Neural Information Processing Systems*, 2009, pp. 1678–1686.

[19] Thomas Deselaers, Georg Heigold, and Hermann Ney, "Speech recognition with state-based nearest neighbour classifiers.," in *INTERSPEECH*. Citeseer, 2007, pp. 2093–2096.

[20] Van Hai Do, Xiong Xiao, Eng Siong Chng, and Haizhou Li, "Kernel Density-based Acoustic Model with Cross-lingual Bottleneck Features for Resource Limited LVCSR," in *Interspeech*, 2014.

[21] Kartik Audhkhasi, Abhinav Sethy, Bhuvana Ramabhadran, and Shrikanth S Narayanan, "Semi-supervised term-weighted value rescoring for keyword search," in *ICASSP)*, 2014, pp. 7869–7873.

[22] Van Tung Pham, Haihua Xu, Nancy F Chen, Sunil Sivadas, Boon Pang Lim, Eng Siong Chg, and Haizhou Li, "Discriminative score normalization for keyword search detection," in *Proc. ICASSP*, 2014.

[23] Joon Ho Lee, "Analyses of multiple evidence combination," in *ACM SIGIR Forum*. ACM, 1997, vol. 31, pp. 267–276.

[24] Van Tung Pham, Nancy F Chen, Sunil Sivadas, Haihua Xu, I-Fan Chen, Chongjia Ni, Eng Siong Chng, Haizhou Li, et al., "System and keyword dependent fusion for spoken term detection," in *Spoken Language Technology Workshop*, 2014.

[25] Lin-shan Lee, James Glass, Hung-yi Lee, and Chun-an Chan, "Spoken content retrieval - beyond cascading speech recognition with text retrieval," *IEEE TASLP*, vol. 23, no. 9, pp. 1389–1420, 2015.

[26] Van Tung Pham, Haihua Xu, Xiong Xiao, Nancy F. Chen, Eng Siong Chng, and Haizhou Li, "Keyword search using query expansion for graph-based rescoring of hypothesized detections," in *ICASSP*, 2016.

[27] Damianos Karakos, Ivan Bulyko, Richard Schwartz, Stavros Tsakalidis, Long Nguyen, and John Makhoul, "Normalization of phonetic keyword search scores," in *ICASSP*, 2014, pp. 7834–7838.

[28] Daniel Povey et al., "The Kaldi speech recognition toolkit," in *Proc. of IEEE ASRU*, 2011.

[29] Haihua Xu, Van Hai Do, Xiong Xiao, and Eng Siong Chng, "A Comparative Study of BNF and DNN Multilingual Training on Cross-Lingual Low-Resource Speech Recognition," in *Interspeech*, 2015.

[30] Jonathan Mamou, Bhuvana Ramabhadran, and Olivier Siohan, "Vocabulary independent spoken term detection," in *Proc. ACM SIGIR conference on Research and development in information retrieval*, 2007, pp. 615–622.