LOW-RESOURCE KEYWORD SEARCH STRATEGIES FOR TAMIL

Nancy F. Chen[1], Chongjia Ni[1], I-Fan Chen[2], Sunil Sivadas[1],Van Tung Pham[3], Haihua Xu[3], Xiong Xiao[3], Tze Siong Lau[3], Su Jun Leow[3], Boon Pang Lim[1], Cheung-Chi Leung[1], Lei Wang[1], Chin-Hui Lee[2], Alvina Goh[3], Eng Siong Chng[3], Bin Ma[1], Haizhou Li[1]

[1]Institute for Infocomm Research, A*STAR, Singapore, [2]Georgia Institute of Technology, USA, [3]Nanyang Technological University, Singapore

nfychen@i2r.a-star.edu.sg

ABSTRACT

We propose strategies for a state-of-the-art keyword search (KWS) system developed by the SINGA team in the context of the 2014 NIST Open Keyword Search Evaluation (OpenKWS14) using conversational Tamil provided by the IARPA Babel program. To tackle low-resource challenges and the rich morphological nature of Tamil, we present highlights of our current KWS system, including: (1) Submodular optimization data selection to maximize acoustic diversity through Gaussian component indexed N-grams; (2) Keyword-aware language modeling; (3) Subword modeling of morphemes and homophones.

Index Terms— Spoken term detection (STD), keyword spotting, under-resourced languages, active learning, unsupervised learning, semi-supervised learning, inflective languages, agglutinative languages, morphology, deep neural network (DNN)

1. INTRODUCTION

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. Applications of KWS include spoken document indexing and retrieval [1] and spoken language understanding [2].

KWS systems can be categorized into two groups: (i) classic keyword-filler based KWS [3], and (ii) large vocabulary continuous speech recognition (LVCSR) based KWS [4]. In keyword-filler based KWS systems, a spoken utterance is represented as a sequence of keywords and non-keywords (often referred to as fillers [3]). Customized detectors are built for the keywords. Keyword-filler based systems often achieve high detection rate using only a small dataset for acoustic model training, but they do not scale well when the number of keywords increases.

By contrast, LVCSR-based KWS systems are flexible in handling a large number of keywords, yet require sufficiently large amounts of transcribed training data to achieve good performance. Therefore, LVCSR-based KWS has worked well on resource-rich languages like English, as has been shown in the NIST 2006 Spoken Term Detection Evaluation [5]. However, such transcribe-and-search approaches pose particular challenges to low-resource languages such as Zulu and Tamil.

To tackle these challenges, the IARPA Babel program aims to foster research "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." The NIST 2014 Open Keyword Search Evaluation (OpenKWS14) was held in April using the surprise language of Tamil.

The challenges of the NIST OpenKWS14 Evaluation include linguistic peculiarities of Tamil (e.g., more than 30% out-of-vocabulary (OOV) rate due to its rich morphological structure), poor audio quality (e.g., noise, soft volume, cross-talk), and limited amount of transcribed data. To address such challenges, we discuss our recent endeavors (see Figure 1) and related work below.

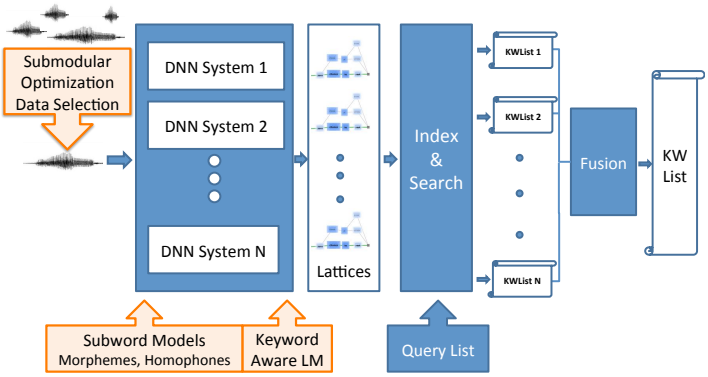

Fig. 1. Proposed keyword search system for low-resource languages. Orange blocks are highlights discussed in this work: submodular optimization for selecting data to transcribe, subword modeling of morphemes and homophones, keyword-aware language modeling.

2. RELATION TO PRIOR WORK

2.1. Active Learning for Selecting Audio to Transcribe

Transcribing speech data is time-consuming and labor-intensive, especially for low-resource languages where linguistic expertise is limited or lacking. Thus it is critical to select the most informative and representative subset of audio for human transcription. In prior work, most approaches select utterances in a greedy fashion according to their utility scores (e.g., confidence scores from automatic speech recognition (ASR) [6, 7]). Similar to the confidence-based approaches, [8, 9] use entropy reduction to select unlabeled utterances. Low-resource methods for speech data selection include [10], which only considers the ideal case where transcriptions are available in the first place.

Aforementioned methods usually require an ASR system in place already for data selection, nor do they guarantee optimality with regard to an objective function. Alternatively,[11, 12] formulate this data selection problem as a constrained submodular optimization setup. In this work, we follow this line of thinking and extend it to KWS tasks. In particular, we propose to use Gaussian component index based n-grams as acoustic features to select utterances to

transcribe.

2.2. Keyword-Aware Language Modeling

If keyword queries are known a priori, one can leverage such knowledge to improve KWS performance. Previous work has shown how to exploit keyword information for acoustic modeling [13] and decoding [14]. Our previous efforts exploit keyword information in language modeling in Vietnamese [15]. In this work, we investigate our approach in Tamil and further expand it to a framework integrating advantages from both keyword-filler based KWS and LVCSR-based KWS.

2.3. Subword Modeling: Morphemes and Homophones

Mainstream LVCSR systems suffer from out-of-vocabulary (OOV) issues. For morphologically-rich languages like Tamil, OOV rate is especially high. While phones are commonly used to help resolve OOVs [16], morphs (automatically parsed morphemes[1]) have also been used in ASR [17]. In this work, we mitigate the data sparsity issue of the morphologically-rich vocabulary in Tamil by integrating morphs in the lexicon and language models. Our approach is similar to [18], but we apply it on Tamil instead of Turkish. For Tamil ASR, morph-based LMs have been reported [19]; in this work, we estimate smoother word-morph LMs to address the serious data sparsity issues of our dataset.

In our unpublished work, we found homophones useful in Vietnamese KWS. In this work, we continue this line of investigation in Tamil. To the best of our knowledge, to date there is no reported work on using homophones in KWS.

3. LOW-RESOURCE KEYWORD SEARCH STRATEGIES

3.1. Submodular Optimization to Select Audio to Transcribe

3.1.1. Problem Formulation

Given a set of N utterances $V = \{1, 2, ..., N\}$, we can construct a non-decreasing submodular set function $f : 2^V \rightarrow \mathbb{R}$, mapping each subset $S \subseteq V$ to a real number. We can formulate the problem of selecting the best subset S given some budget K (e.g., maximum number of transcribed utterances) as a monotone submodular function maximization under a knapsack constraint:

$$\max_{S \subseteq V} \{f(S) : |S| \leq K\} \tag{1}$$

Submodularity can be interpreted as the property of diminishing returns, which is for any subset $R \subseteq S \subseteq V$ and any utterance $s \in V \setminus S$,

$$f(S \cup \{s\}) - f(S) \leq f(R \cup \{s\}) - f(R). \tag{2}$$

Let U be a set of features, and $P = \{p_u\}_{u \in U}$ be the probability distribution over the set U. Let $m_u(S) = \sum_{s \in S} m_u(s)$ be a non-negative score for feature u in set S, where $m_u(s)$ measures the degree to which utterance $s \in S$ possesses feature u. We can compute the KL-divergence between the two distributions P and $\overline{m}_u(S) = \frac{m_u(S)}{\sum_{u \in U} m_u(S)}$:

$$\begin{aligned} & D_{KL}(P||\overline{m}_u(S)) \\ = & \sum_{u \in U} p_u \log p_u - \sum_{u \in U} p_u log(m_u(S)) + \log(\sum_{u \in U} m_u(S)) \\ = & const. + \log(\sum_{u \in U} m_u(S)) - \sum_{u \in U} p_u \log(m_u(S)) \end{aligned}$$

We define our objective function f as follows:

$$f(S) = \log(\sum_{u \in U} m_u(S)) - D_{KL}(P||\overline{m}_u(S)) \tag{3}$$

$$= \sum_{u \in U} p_u \log(m_u(S)). \tag{4}$$

The first term in Eq. (3) represents the acoustic diversity characterized by m, while the second term represents how close the distribution $\overline{m}_u(S)$ is to the distribution P, estimated from a held-out dataset. We want to maximize the diversity characterized by m (first term in Eq. (3)) but at the same time ensure m characterizes the held-out data compactly (second term in Eq. (3)).

3.1.2. GMM Tokenization for Maximizing Acoustic Diversity

A GMM with M components trained on unlabeled training data is used to label (tokenize) the training data and a held-out dataset in terms of Gaussian components. For each frame i, label j is assigned: $j = \arg\max_j P(i|c_j)$, where c_j is the Gaussian mixture component, $j = 1, ..., M$.

The concepts of term frequency and inverse document frequency are used to characterize the tokenized audio. A *term* is defined as an n-gram of the labeled indices. These terms make up the feature set U in Section 3.1.1. The modular function defined in Section 3.1.1 is defined as the product of the term frequency $tf_u(s)$ and inverse document frequency idf_u

$$m_u(S) = \sum_{s \in S} m_u(s) = \sum_{s \in S} tf_u(s) \times idf_u, \tag{5}$$

where each utterance s is considered a document in the training set.

The probability distribution p_u in Eq. (4) is the term frequency estimated from the held-out data: $p_u = \frac{n_u}{\sum_u n_u}$, where n_u is number of times feature u occurred.

3.2. Keyword-Aware Language Modeling (LM)

Let $q = (w_1, w_2, \ldots, w_L)$ be an L-word query. In keyword-filler based KWS, the prior probability $P(q)$ is by default set to $P(q) = 1/N$, often resulting in high false alarms. (Typically $N \leq 100$.)

By contrast, in LVCSR-based KWS, the prior probability of q can be estimated as:

$$P_{\text{LVCSR}}(q) = \sum_{h \in H} P(q|h) \approx \sum_{h \in H} \{\prod_{i=1}^{L} P_{\text{n-gram}}(w_i|h_i(h, q))\}, \tag{6}$$

where $h_i(h, q)$ is the history of w_i in the query q dictated by the order n, $P_{\text{n-gram}}(.)$ is the probability estimated by n-gram language model. In low-resource scenarios, where there is insufficient text data to properly train n-gram LMs, prior probabilities are often

[1] *Morphemes* are the smallest semantic units in linguistics. For example, the word *unsuccessful* consists of 3 morphemes: 'un', 'success', 'ful'.

underestimated. This underestimation causes high miss probability, especially for multi-word queries. To alleviate the underestimation problem, one can integrate the approach in keyword-filler based KWS and LVCSR-based KWS:

$$P_{\mathrm{KW-aware}}(q) = \max\{P_{\mathrm{LVCSR}}(q), k(q)\} \quad (7)$$

where $k(q)$ is the minimum prior set for query q to alleviate the prior under-estimation problem in low-resource LVCSR-KWS, where insufficient text data is available for language modeling. In this paper, we assume all keywords share the same k (i.e, $k(q) = k$.) For more detailed discussion of such proposed grammar, please refer to [20].

3.3. Word-Morph Interpolated Language Model

Representing out-of-vocabulary (OOV) entries using morphs (automatically parsed morphemes) is insufficient to resolve data sparsity issues with morphologically-rich languages like Tamil. If the morph-based lexical entries have low occurrences, the miss probability of such keywords are still high despite it no longer being OOV. To mitigate this effect, we exploit word-morph interpolated language models (LM) to provide smoother estimates.

Three LMs are first constructed: (1) Word-based LM λ_{W}: a 3-gram word LM is trained on all the word entries. (2) Morph-based LM λ_{M}: a 3-gram morph LM is trained by parsing word entries into morphs using Morfessor [21]. (3) Hybrid Word-Morph LM λ_{H}: words with more than one occurrence in the training data were retained, whereas words with only one occurrence were parsed into morphs by Morfessor [21]. An interpolated language model was then estimated: $\lambda_{W-M} = \alpha\lambda_{\mathrm{W}} + \beta\lambda_{\mathrm{M}} + (1-\alpha-\beta)\lambda_{\mathrm{H}}$.

4. EXPERIMENTS

For clarity purposes, we only show a subset of submitted systems for OpenKWS14 and corresponding follow-up analysis to demonstrate the proposed strategies discussed here.

4.1. Setup

This effort uses the IARPA Babel Program Tamil language collection release IARPA-babel204b-v1.1b for the NIST OpenKWS14 Evaluation. The training set includes 80 hours of conversational telephone speech. Two conditions are defined: (1) Full Language Pack (FLP): 60 hours of transcriptions and a corresponding lexicon. (2) Limited Language Pack (LLP): a 10 hr subset of FLP transcriptions. The developmental set is 10 hr with transcriptions. The evaluation set is 75 hr with no transcriptions nor timing information; transcriptions of a 15 hr subset (*evalpart1*) was released after OpenKWS14. All results reported are on *evalpart1*.

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

$$\mathrm{TWV}(\theta) = 1 - [P_{\mathrm{miss}}(\theta) + \beta P_{\mathrm{FA}}(\theta)], \quad (8)$$

where θ is the decision threshold. Actual term-weighted value (ATWV) is the TWV using the chosen decision threshold.

4.2. Baseline System

All systems were developed using Kaldi [22]. While fundamental frequency variation features were shown to improve ASR for both tonal and non-tonal languages [23] and improve KWS for OpenKWS13 [24], it actually hurt ASR/KWS performance for this data. F0, on the other hand, consistently helped in pilot experiments, therefore all systems adopted F0.

4.2.1. Implementation Details

We adapted voice activity detection (VAD) in [24] to reduce noise. The WAV files are especially noisy, resulting in virtually 100% word error rate. Classic noise canceling methods such as Wiener filtering was ineffective. Instead, speech enhancement using a log minimum mean-square error spectral amplitude estimator [25] was applied to WAV files before VAD, as it improved the speech quality significantly by removing perceptually audible distortions. When comparing VAD and ground-truth segments, ATWV showed insignificant difference ($< 0.13\%$ relative).

MFCC (13-dim) and F0 (2-dim) were extracted; 9 adjacent feature frames were then concatenated and applied with a LDA+MLLT+fMLLR transform. The 40-dim fMLLR features were used for bottleneck feature (BNF) extraction (6 hidden layers each with 2048 nodes) to extract 42-dim BNF. The 40-dim fMLLR feature and 42-dim BNF were then concatenated to form 82-dim features. Then fMLLR transform was applied again (60-dim). We used 6 hidden layers (2048 nodes each) and 4838 senone target states for the DNN acoustic model. The training procedure is as follows: (1) 1 iteration of pre-training; (2) cross entropy criterion training; (3) scalable minimum Bayes risk criterion based sequence training [26].

Phonetisaurus [27] was used to obtain OOV pronunciations. A trigram language model was trained on word tokens. The beam width was set to 18 for lattice decoding. 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 [16], WComMNZ [16], and keyword specific thresholding (KST) [28] were applied consecutively to combine systems. For individual systems, only KST was done.

4.2.2. Results

Table 1 shows the baseline ATWV results when using word 3-gram LM (λ_{W}) for LLP and FLP. In the next section, we examine how leveraging keyword information can boost performance.

Table 1. Keyword-Aware LM outperforms baseline LM. All LMs are word-based.

Transcription Condition	Baseline ATWV	Keyword-Aware LM ATWV	Relative Gain (%)
LLP: 10 hr	0.2313	0.3182	37.6%
FLP: 60 hr	0.4222	0.4852	14.9%

4.3. Keyword Aware Language Model (LM) Experiment

4.3.1. Implementation Details

Setup is the same as Section 4.2.1 except the LM is estimated using context-simulated keyword LM [15]:

$$P_{\mathrm{KW-LM}}(w|h) = \gamma P_{\mathrm{KWLM}}(w|h) + (1-\gamma)P_{\mathrm{LM}}(w|h), \quad (9)$$

where $\gamma = 0.3$, h is the history of the current word w, P_{KWLM} is an LM estimated by padding keywords with bigram entries from the training data, and P_{LM} is the trigram LM in Section 4.2.1.

4.3.2. Results

Table 1 shows that when keyword-aware LM is used, relative gains reach 37.6% (LLP) and 14.9% (FLP). Further analysis shows that the gains are due to reduction in miss probability, which is penalized more heavily than false alarms in OpenKWS settings. We also observe larger gains when keywords are multi-words when using the keyword-aware LM framework. Due to space constraints, comparisons related to how effective the keyword-aware framework differs according to language peculiarities, implementation methods, and keyword length are reported in [20].

4.4. Subword Experiments

Subword modeling is essential especially for low-resource languages where keywords are not known a priori. Here we investigate how morphemes and homophones help resolve data sparsity issues.

4.4.1. Morpheme Subword Modeling

The system implementation is the same as in Section 4.2.1, except the lexicon and word-morph interpolated LM setup is as described in Section 3.3, where $\alpha = 0.4, \beta = 0.3$. We applied linguistic constraints to reduce linguistically-illegal morphs, but gains were insignificant compared to that of increasing lattice sizes. In addition, non-speech tags were removed from the LM (consistent marginal gains on 5 developmental keyword lists). Table 2 shows that using word-morph interpolated LM improves ATWV by 4.5% relative for LLP and 3.3% relative for FLP.

Table 2. Word-Morph LM outperforms word LM.

Transcription Condition	Word LM ATWV	Word-Morph LM ATWV	Rel. Gain (%)
LLP: 10 hr	0.2313	0.2418	4.5
FLP: 60 hr	0.4222	0.4363	3.3

Table 3. Homophone System S_{H} and Sub-Homophone System $S_{\mathrm{H_{sub}}}$ complement each other.

System	LLP ATWV	FLP ATWV
S_{H}	0.0832	0.2634
$S_{\mathrm{H_{sub}}}$	0.0838	0.2748
S_{H} + $S_{\mathrm{H_{sub}}}$	0.1243	0.2872

4.4.2. Homophone Subword Modeling

Homophones are words that are written differently but sound the same, like *see* and *sea*. The homophone system S_{H} was implemented as in Section 4.2.1, except words are replaced with their pronunciations. The sub-homophone system, $S_{\mathrm{H_{sub}}}$ was implemented by further segmenting homophones into morphs using Morfessor.

From Table 3, we see that the homophone system S_H and sub-homophone system $S_{H_{sub}}$ perform similarly to each other for both LLP and FLP conditions. When fused with each other, we get 49.4% relative gain for LLP and 4.5% relative gain for FLP, suggesting that sub-homophones are much more complementary to homophones in low resource scenarios. The homophone results shown here are sub-optimal when compared to its word-morph counterpart. We suspect this discrepancy to be language-dependent. In the OpenKWS13 Vietnamese task, we observed a 26.3% relative gain when using homophones instead of words. Vietnamese words are constructed by a finite set of syllables, which are phonetically equivalent to homophones, making homophones an elegant choice in handling OOVs. For future work, we plan to investigate whether sub-homophones can drive further gains in Vietnamese.

4.5. Submodular Optimization Data Selection Experiment

In this experiment, we analyze how to select data to transcribe to maximize KWS performance and minimize transcription cost.

4.5.1. Implementation Details

We follow the proposed algorithm described in Section 3.1.2. The total number of mixture components $M = 2048$, and bigrams of labeled frames are used to designate a term to compute term-frequency and inverse-document-frequency. The 10 hr developmental data is used as the held-out dataset for estimating the distribution of p_u in Eq. (4). The KWS system is the same as in Section 4.4.1.

4.5.2. Results

Table 4 shows that the proposed 10 hr subset outperforms Baseline-1 (random 10 hr subset) and Baseline-2 (NIST-LLP 10 hr subset) by 21.0% and 15.5%, showing that the LLP 10 hr subset can be more optimally chosen to achieve better KWS performance without increasing transcription cost. The relative ATWV gain of increasing transcriptions from 10 hr to 60 hr is less in the submodular case (52.7%) compared to those from Baseline-1 (82.9 %) and Baseline-2 (76.4 %), indicating that the return on transcription cost is more effective when using the submodular optimization approach.

Table 4 also shows that by maximizing acoustic diversity in the proposed approach, we implicitly enrich the vocabulary in the lexicon, and thus alleviate OOV issues: compared to the proposed 10 hr subset, the OOV keywords decreases by 17.0 % relative for Baseline-1 (random 10 hr subset) and 42.3 % relative for Baseline-2 (NIST-LLP 10 hr subset). This byproduct benefit helps resolve OOV issues at a more fundamental stage when developing spoken language technology. For more detailed analysis, please see [29].

Table 4. Submodular data selection for word transcriptions.

Transcription Condition	ATWV	OOV counts
Baseline-1: Random 10 hr subset	0.2386	1171
Baseline-2: NIST-LLP (10 hr subset)	0.2474	1686
Proposed submodular 10 hr subset	0.2857	972
Upper bound: NIST-FLP (full 60 hr)	0.4363	407

5. DISCUSSION

In this work, we investigated three strategies for low-resource keyword search. We expect our submodular optimization data selection approach to generalize well in languages other than Tamil since similar approaches works in Mandarin LVCSR [30]. Similarly, the keyword-aware language model approach also works for Vietnamese [20]. By contrast, subword modeling (morphemes, homophones) appears to be more language-dependent.

While our LVCSR-KWS work in Tamil and Vietnamese [24] focus on text queries, we have inspired strategies used in spoken term detection of audio queries. For example, [31] proposed partial-matching symbolic search, which complements popular pattern matching approaches using dynamic time warping in Query-by-Example Search on Speech (QUESST), formerly called Spoken Web Search (SWS), in MediaEval 2014.

6. REFERENCES

[1] John Makhoul, Francis Kubala, Timothy Leek, Daben Liu, Long Nguyen, Richard Schwartz, and Amit Srivastava, "Speech and language technologies for audio indexing and retrieval," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1338–1353, 2000.

[2] Biing-Hwang Juang and Sadaoki Furui, "Automatic recognition and understanding of spoken language-a first step toward natural human-machine communication," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1142–1165, 2000.

[3] Jay G Wilpon, L Rabiner, Chin-Hui Lee, and ER Goldman, "Automatic recognition of keywords in unconstrained speech using hidden markov models," *IEEE TASLP*, vol. 38, no. 11, pp. 1870–1878, 1990.

[4] J Gauvain and Lori Lamel, "Large-vocabulary continuous speech recognition: advances and applications," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1181–1200, 2000.

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

[6] Dilek Hakkani-Tur, Giuseppe Riccardi, and Allen Gorin, "Active learning for automatic speech recognition," in *Proc. IEEE ICASSP*, 2002, vol. 4, pp. IV–3904.

[7] Lori Lamel, Jean-Luc Gauvain, and Gilles Adda, "Lightly supervised and unsupervised acoustic model training," *Computer Speech & Language*, vol. 16, no. 1, pp. 115–129, 2002.

[8] Dong Yu, Balakrishnan Varadarajan, Li Deng, and Alex Acero, "Active learning and semi-supervised learning for speech recognition: A unified framework using the global entropy reduction maximization criterion," *Computer Speech & Language*, vol. 24, no. 3, pp. 433–444, 2010.

[9] Nobuyasu Itoh, Tara N Sainath, Dan Ning Jiang, Jie Zhou, and Bhuvana Ramabhadran, "N-best entropy based data selection for acoustic modeling," in *Proc. IEEE ICASSP*, 2012, pp. 4133–4136.

[10] Yi Wu, Rong Zhang, and Alexander Rudnicky, "Data selection for speech recognition," in *Proc. IEEE ASRU*, 2007, pp. 562–565.

[11] Hui Lin and Jeff Bilmes, "How to select a good training-data subset for transcription: Submodular active selection for sequences," in *INTERSPEECH*, 2009.

[12] Kai Wei, Yuzong Liu, Katrin Kirchhoff, and Jeff Bilmes, "Using document summarization techniques for speech data subset selection.," in *HLT-NAACL*, 2013, pp. 721–726.

[13] 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.

[14] Bing Zhang, Richard M Schwartz, Stavros Tsakalidis, Long Nguyen, and Spyros Matsoukas, "White listing and score normalization for keyword spotting of noisy speech.," in *INTERSPEECH*, 2012.

[15] I-Fan Chen, Chongjia Ni, Boon Pang Lim, Nancy F Chen, and Chin-Hui Lee, "A novel keyword+lvcsr-filler based grammar network representation for spoken keyword search," in *ISCSLP*, 2014.

[16] 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.

[17] Hasim Sak, Murat Saraçlar, and Tunga Gungor, "Morpholexical and discriminative language models for turkish automatic speech recognition," *IEEE TASLP*, vol. 20, no. 8, pp. 2341–2351, 2012.

[18] Yanzhang He, Brian Hutchinson, Peter Baumann, Mari Ostendorf, Eric Fosler-Lussier, and Janet Pierrehumbert, "Subword-based modeling for handling oov words inkeyword spotting," in *Proc. IEEE ICASSP*, 2014, pp. 7864–7868.

[19] Melvin Jose Johnson Premkumar, Ngoc Thang Vu, and Tanja Schultz, "Experiments towards a better lvcsr system for tamil," in *INTERSPEECH*, 2013.

[20] 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 *Proc. IEEE ICASSP*, 2015.

[21] "Morfessor 2.0.0: http://www.cis.hut.fi/projects/morpho/morfessor2.shtml," last accessed, August 2014.

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

[23] 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*, Olomouc; Czech Republic, 2013.

[24] 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.

[25] Yariv Ephraim and David Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE TASLP*, vol. 32, no. 6, pp. 1109–1121, 1984.

[26] 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.

[27] J. R. Novak, "Phoneticsaurus - A WFST-driven Phoneticizer. Available: https://code.google.com/p/phonetisaurus," 2012.

[28] 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.

[29] Chongjia Ni, Cheung-Chi Leung, Lei Wang, Nancy F Chen, and Bin Ma, "Unsupervised Data Selection and Word-Morph Mixed Language Model for Tamil Low-Resource Keyword Searh," in *Proc. IEEE ICASSP*, 2015.

[30] Chongjia Ni, Lei Wang, Haibo Liu, Cheung-Chi Leung, Li Lu, and Bin M, "Submodular data selection with acoustic and phonetic features for automatic speech recogntion," in *Proc. IEEE ICASSP*, 2015.

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