IMPROVED SPOKEN TERM DETECTION USING SUPPORT VECTOR MACHINES WITH ACOUSTIC AND CONTEXT FEATURES FROM PSEUDO-RELEVANCE FEEDBACK

Tsung-wei Tu [#1], Hung-yi Lee [*2], Lin-shan Lee [#*3]

[#] *Graduate Institute of Computer Science and Information Engineering, National Taiwan University*
[*] *Graduate Institute of Communication Engineering, National Taiwan University*
No. 1, Sec. 4, Roosevelt Road, Taipei, 10617 Taiwan
[1] hpttw0608@gmail.com, [2] tlkagkb93901106@yahoo.com.tw, [3] lslee@gate.sinica.edu.tw

***Abstract*—This paper reports a new approach to improving spoken term detection that uses support vector machine (SVM) with acoustic and linguistic features. As SVM is a good technique for discriminating different features in vector space, we recently proposed to use pseudo-relevance feedback to automatically generate training data for SVM training and use SVM to re-rank the first-pass results considering the context consistency in the lattices. In this paper, we further extend this concept by considering acoustic features at word, phone and HMM state levels and linguistic features of different order. Extensive experiments under various recognition environments demonstrate significant improvements in all cases. In particular, the acoustic features at the HMM state level offered the most significant improvements, and the improvements achieved by acoustic and linguistic features are shown to be additive.**

I. Introduction

Spoken term detection (STD) refers to the retrieval from a large spoken document archive and returning a list of spoken segments containing the term requested by the user. This technology is crucial to accessing multimedia content, including audio signals. Many different approaches have been proposed for enhancing STD [1], [2], [3]. In general, there are two stages in STD [4]. The audio content is first recognized and transformed into transcriptions or lattices using a set of acoustic and language models. The retrieval engine searches through the recognition results and then based on the query returns to the user a list of potentially relevant spoken segments. The returned segments are usually ranked by the relevance scores derived from the recognition output. As a result, the performance of STD depends heavily on the acoustic and language models used in recognition. However, in practice the relatively poor performance of STD is due to the limited robustness of the available acoustic and language models, in particular with respect to the various topics represented in the audio content on the Internet, as well as the variety of speakers under different acoustic conditions in varying environments.

There have been many previous works [5], [6], [7], [8] taking advantage of the discriminative capability of machine learning methods such as support vector machines (SVM) or multi-layer perceptrons (MLP) to facilitate STD. A SVM or MLP classifier is trained to identify if a spoken segment contains the entered query term or not. To train the machine learning classifiers, the training data must be reasonably matched to the audio corpus to be retrieved. However, such data is usually not available.

To fulfill the training data required for machine learning methods, pseudo-relevance feedback (PRF), which has been used to improve performance on text retrieval [9], [10] as well as STD [11], [12], can be used to automatically generate labelled data. For PRF, the system assumes that the spoken segments with high lattice-derived relevance scores contain the query term and hence denotes them as *pseudo-relevant segments*; segments with low relevance scores are likewise denoted as *pseudo-irrelevant segments*. These pseudo-relevant and irrelevant segment sets are then taken as the training data for machine learning. In this way the system is still able to take advantage of machine learning approaches without using any real labelled data.

We recently proposed to use the *query context*, that is, the contexts of the query terms in the recognition results, in the above PRF framework using SVM [13]. Although query context only includes linguistic-level information, information in the acoustic feature space may be also useful for STD. Ranking performance can also be improved by increasing the relevance scores of the segments that are acoustically similar to pseudo-relevant segments, based on the segment similarities from the MFCC sequences corresponding to query hypotheses [12]. Here in this paper we further propose taking into account features with both linguistic and acoustic information in SVM model training within the PRF framework, and test this approach with recognition results of varying qualities. To use SVM to take into account acoustic information, which is represented as a sequence of features, the MFCC sequence corresponding to the query hypothesis is represented as a single feature vector; here we investigate various ways to construct this feature vector.

978-1-4673-0367-5/11/$26.00 ©2011 IEEE

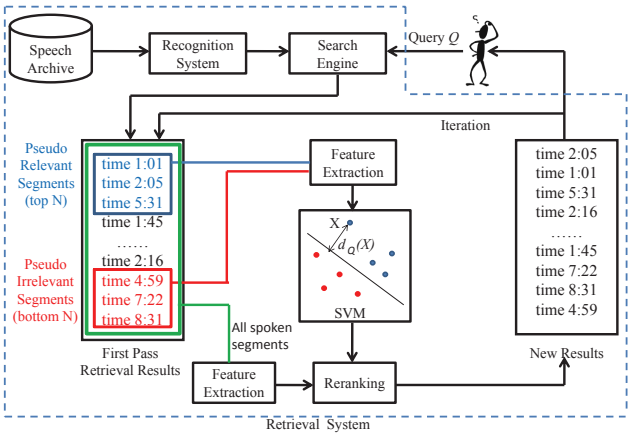


Fig. 1: The framework for pseudo-relevance feedback using SVM.

II. Framework

Fig. 1 shows the framework of pseudo-relevance feedback with SVM. In first-pass retrieval (Section II-A), conventional STD technologies rank the spoken segments X based on the relevance scores derived from the recognition lattices with respect to query Q. On the left in the figure is shown the list of first-pass retrieval results. As described in Section II-B, every spoken segment in the list is represented by a feature vector, and the top N and bottom N spoken segments are selected as the pseudo-relevant or irrelevant spoken segments for SVM model training. In Section II-C, we use SVM as a classifier to classify spoken segments and derive confidence scores according to classification results. Section II-D describes segment re-ranking, which can be performed iteratively.

A. The First Pass

The whole audio archive to be detected is first divided into approximately utterance-length spoken segments X. Then each spoken segment is transcribed to a lattice of word hypotheses and posterior probabilities for each word. The relevance score $S_Q(X)$ of each spoken segment X with respect to the query term Q is defined as

$$S_Q(X) = \sum_{\{a|word(a)=Q\}} P(a|X), \tag{1}$$

where a is any arc in the lattice and $word(a)$ is the word hypothesis of that arc. Only for lattices with word hypotheses that match the query term do we accumulate the relevance score from the hypothesis's posterior probability $P(a|X)$. After obtaining the list of spoken segments that contains the query term and their associated relevance scores $S_Q(X)$, we rank the segments according to their relevance scores. This completes the first pass. Similar relevance score approaches are widely used in other STD techniques.

B. Feature Extraction and Pseudo-relevance Feedback

Given the result of the first pass, each retrieved spoken segment is represented as a feature vector. The features used here describe the acoustic and linguistic information of each spoken segment, which will be used in SVM training to discriminate relevant and irrelevant segments. Feature representations are described further in Section III.

As shown in Fig. 1, the top N and bottom N first-pass retrieved spoken segments are taken as the pseudo-relevant and irrelevant spoken segments, respectively.

C. Support Vector Machines

For each query, in order to classify all spoken segments, pseudo-relevant and irrelevant spoken segments are used to train an SVM model. Thus a hyperplane is trained for each query term Q according to the PRF training data. This hyperplane is then used to classify all of the spoken segments returned in the first pass. Based on these classification results, each spoken segment is given a value $d_Q(X)$ derived from the distance between the hyperplane and its feature point position: $d_Q(X)$ is positive when X is classified as relevant, and negative when irrelevant. The absolute value of $d_Q(X)$ represents the distance from the SVM hyperplane. To derive the confidence score $SVM_Q(X)$ for spoken segment X with respect to query Q, $d_Q(X)$ is linearly normalized as

$$SVM_Q(X) = \frac{d_Q(X) - d_{min}}{d_{max} - d_{min}}, \tag{2}$$

where d_{max} and d_{min} are respectively the maximum and minimum distances to the hyperplane from all spoken segments in a single query.

D. Re-ranking and Iterative Re-ranking

The new segment relevance score $\hat{S}_Q(X)$ is obtained by integrating the original relevance score $S_Q(X)$ in (1) with the confidence score $SVM_Q(X)$ in (2) as

$$\hat{S}_Q(X) = S_Q(X) SVM_Q(X)^{\alpha}, \tag{3}$$

where α is a parameter emphasizing the importance of confidence score $SVM_Q(X)$. A new ranking list is thus generated based on these new relevance scores.

This process can be conducted iteratively by taking those re-ranked by $\hat{S}_Q(X)$ in (3) as the first-pass retrieval results, and repeating the PRF, SVM, and re-ranking procedures over these new results.

III. Feature Representations

In order to train an SVM model for each query term, each spoken segment is represented by a feature vector. In the following, we propose seven different feature representations: three from the acoustic domain and four from the linguistic domain.

A. Acoustic Feature Representations

A "hypothesized region" is the most probable occurrence of query Q in the segment, that is, the word arc whose hypothesis corresponds exactly to the query term, with the highest posterior probability in the lattice. The left top of Fig. 2 is an example of a hypothesized region. Different occurrences of the same term are usually represented by similar MFCC

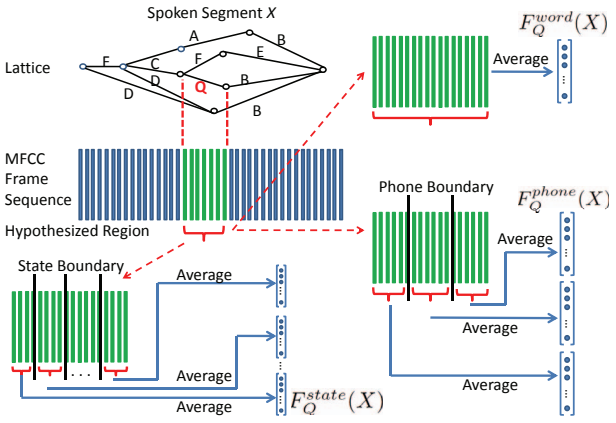


Fig. 2: Different acoustic feature representations for the hypothesized region in the lattice of segment X with respect to query term Q.

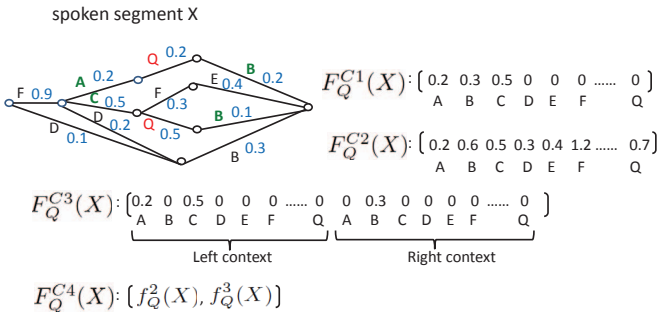


Fig. 3: Different linguistic feature representations for the lattice of segment X lattice with respect to query term Q.

sequences; likewise, markedly different MFCC sequences usually correspond to different terms [12]. Thus it is possible to discriminate relevant and irrelevant spoken segments based on the corresponding MFCC sequences in hypothesized regions. Although the hypothesized region is represented by a sequence of MFCC feature vectors, when using SVM for training and testing, the region should be represented as a single feature vector. Fig. 2 illustrates the three methods used to accomplish this. These feature representations are:

- $F_Q^{word}(X)$: the mean MFCC vector for the hypothesized region, that is, the average of all MFCC frames in the hypothesized region. Hence its dimension is the same as an MFCC vector, and the value of each dimension is the average of all of the corresponding components of the MFCC vectors for all frames in the hypothesized region. This is shown in the upper right corner of Fig. 2.
- $F_Q^{phone}(X)$: using forced alignment, the hypothesized region is segmented into a sequence of phone segments based on the phone sequence of the query term; each phone segment is represented by its mean MFCC feature vector. $F_Q^{phone}(X)$ is hence the concatenation of the phone feature vectors. For example, for a ten-phone query term, the resulting dimensionality is ten times the length of a single MFCC feature vector. This is shown in the lower right corner of Fig. 2.
- $F_Q^{state}(X)$: each phone segment is further segmented into a sequence of state segments by forced alignment, each of which is represented by its average, after which the feature vectors of the state segments in a hypothesized region are concatenated. Thus for three-state phones the dimensionality of $F_Q^{state}(X)$ is three times that of $F_Q^{phone}(X)$. This is shown in the lower left corner of Fig. 2.

B. Linguistic Feature Representations

In the previous work [13], we have proposed the use of query context deriving from the assumption that the same term usually occurs in similar contexts, and that markedly different contexts are indicative of different terms. We thus use the context of the query terms in the recognition results to refine the relevance score of the spoken segments. Four different query context features for the spoken segment with respect to the query term are shown in Fig. 3 and summarized below:

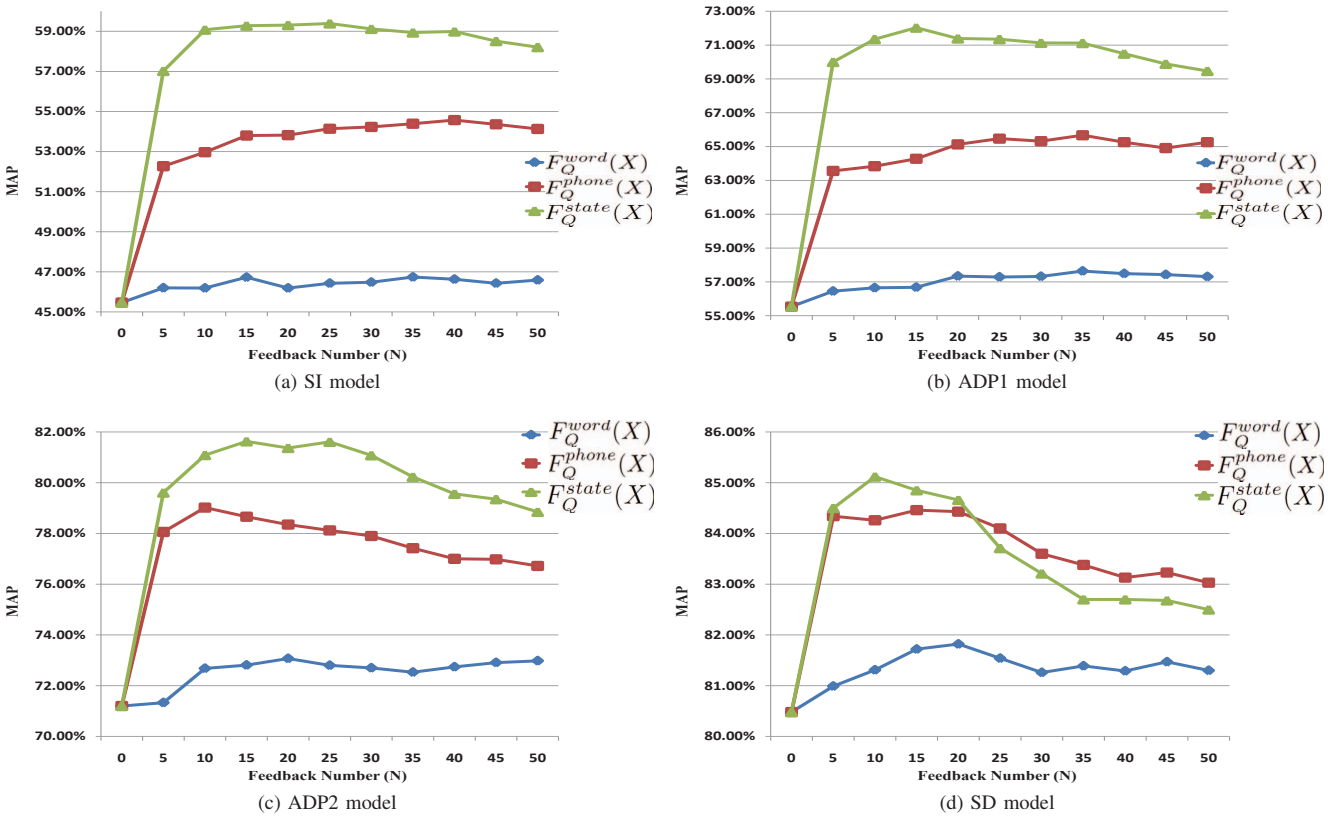
- $F_Q^{C1}(X)$: takes into account only the immediate context of the query. As each dimension of the feature vector corresponds to a lexical word, the vector dimensionality is the number of words in the lexicon. The value of each vector component is the posterior probability summed over all corresponding word arcs immediately connected to the query in the lattice.
- $F_Q^{C2}(X)$: similar to $F_Q^{C1}(X)$, except that all words appearing in the lattice are included, not only those adjacent to the query. Thus it contains the context information for the query throughout the entire segment.
- $F_Q^{C3}(X)$: separates the left and right immediate contexts of the query and hence has twice the dimensionality of $F_Q^{C1}(X)$.
- $F_Q^{C4}(X)$: the concatenation of $F_Q^{C2}(X)$ and $F_Q^{C3}(X)$, this vector is three times the size of of $F_Q^{C1}(X)$.

IV. Combination of Different Features

In Sections III-A and III-B we proposed various feature representations. However, the re-ranking process uses only a single feature representation. Although we would like to integrate features from different domains to improve performance, the large differences in dimensionalities between the feature representations may result in poor results when cascading different features directly, as long features could dominate the results. Hence we instead integrate their confidence scores from SVM classification when combining different features.

$$\hat{S}_Q(X) = S_Q(X)(SVM_Q^A(X) + SVM_Q^L(X))^\alpha, \quad (4)$$

where $SVM_Q^A(X)$ and $SVM_Q^L(X)$ represent confidence scores of one type of acoustic and linguistic feature representation, respectively.



(a) SI model (b) ADP1 model

(c) ADP2 model (d) SD model

Fig. 4: Performance yielded with acoustic features under different feedback numbers and different recognition environments.

V. Experimental Setup

In our experiments, we used a corpus of recorded lectures composed of forty-five hours of a course offered at National Taiwan University produced by a single instructor. Thirty-three hours of the corpus were used as the testing archive to be detected, and the other twelve hours of the corpus were used to train the acoustic model. The speech is quite spontaneous and relatively noisy. It is uttered in the host language of Mandarin Chinese, but embedded with many technical terms produced in the guest language of English. A lexicon with about 10.7K words was used, and a trigram language model was trained on a 600M-word news corpus. We used mean average precision (MAP) as the measure for our performance evaluation. 162 Chinese queries were manually selected, each a single word.

In order to evaluate the performance of our proposed approach under different recognition accuracies, we used four different sets of acoustic models:

- Speaker Independent Model (SI): trained on 24.6 hours of read speech produced by 100 male and 100 female speakers.
- Speaker Adaptation Model 1 (ADP1): adapted from the SI model with 500 utterances taken from the training set of the lecture corpus. Applied only global MLLR.
- Speaker Adaptation Model 2 (ADP2): adapted from the SI model with 500 utterances taken from the training set of the lecture corpus. Applied cascaded global MLLR with 256 classes and maximum a posterior estimation.
- Speaker Dependent Model (SD): trained on the 12-hour training set of the lecture corpus.

The character accuracies of the 1-best transcriptions are shown in Table I.

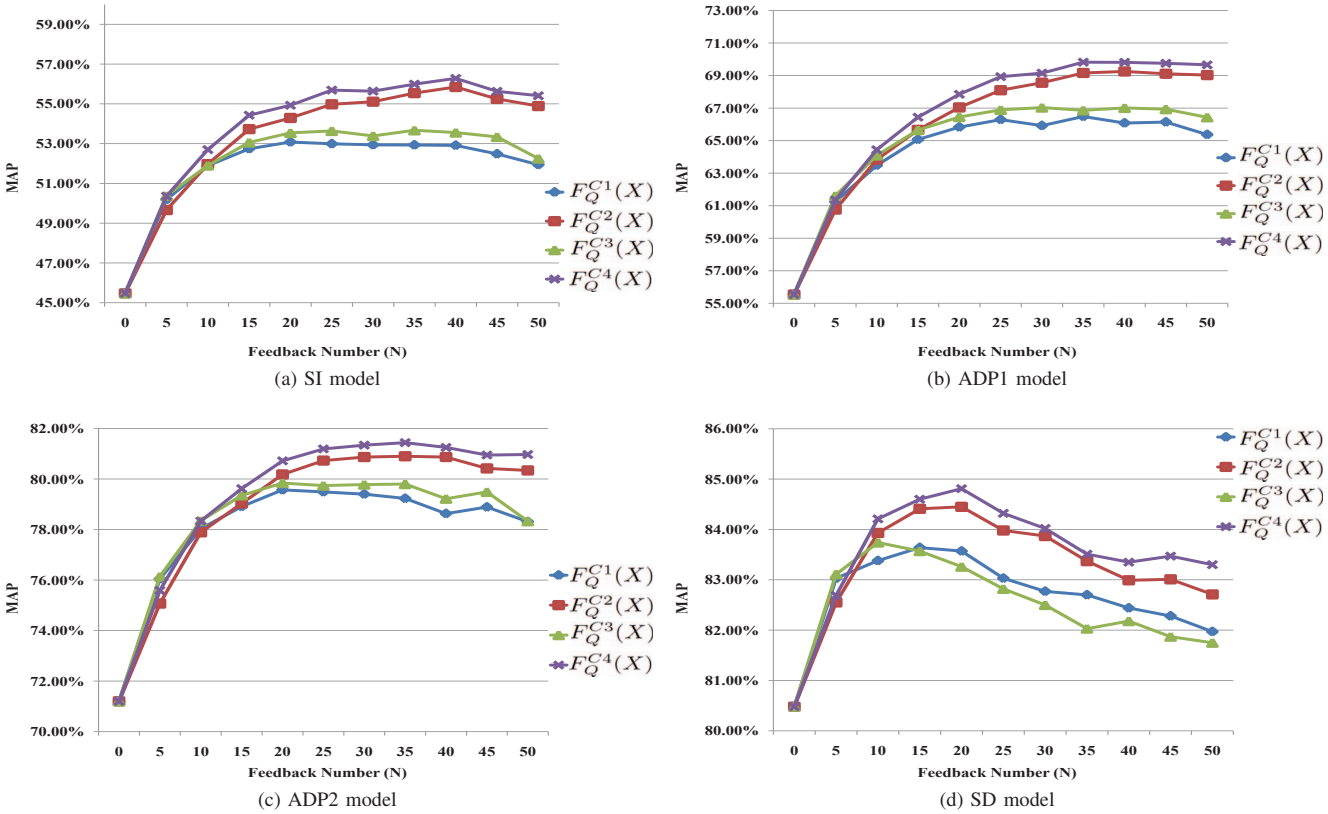
VI. Experimental Results

A. Acoustic Features

Fig. 4 shows the MAP for the acoustic feature representations from Section III-A with the four different sets of acoustic models. The vertical axis depicts the MAP, and the horizontal axis is the feedback number N, set from 5 to 50 with intervals of 5. The SVM training set includes the top N and bottom N first-pass spoken segments as the pseudo-relevant and irrelevant sets. Note if in the first pass there are detected fewer than $2N$ spoken segments in a single query, N is simply set to half of the detection number. $N = 0$ represents the original first-pass result. The parameter α in (3) is adjusted

TABLE I: Character accuracy (%) for different acoustic models

	SI	ADP1	ADP2	SD
Word accuracy	50.26	62.55	72.93	84.08



(a) SI model (b) ADP1 model

(c) ADP2 model (d) SD model

Fig. 5: Performance yielded with linguistic features under different feedback numbers and different recognition environments.

using 4-fold cross-validation. For $F_Q^{phone}(X)$ and $F_Q^{state}(X)$, the acoustic models used for forced alignment are the same as those used to generate the lattices.

The MAP of the first-pass result is clearly consistent with the character accuracy shown in Table I. Although PRF yields MAP improvements in all conditions, $F_Q^{word}(X)$ yields little improvement, because $F_Q^{word}(X)$ is too coarse to represent the hypothesized region. However, more sophisticated feature representations such as $F_Q^{phone}(X)$ or $F_Q^{state}(X)$ do yield significant improvements. In general, $F_Q^{state}(X)$ performs better than $F_Q^{phone}(X)$ because it better represents the hypothesized region.

As the acoustic model quality increases, the best feedback number N decreases. This is shown in Table II. This is because more robust acoustic models ensure that the pseudo-relevant and irrelevant segments more closely correspond to the true relevant and irrelevant segments and hence that fewer such segments are needed, translating to a lower N. In contrast, if poor acoustic models are used, a greater N can help to ensure that the true relevant and irrelevant segments are contained in the pseudo-relevant and irrelevant segments.

B. Linguistic Features

Linguistic features also benefit STD. The results are shown in Fig. 5 with different sets of acoustic models. $F_Q^{C2}(X)$ outperforms $F_Q^{C1}(X)$, which means that not only the immediate neighbors of the query term help in STD; there is also useful information within the whole context of a spoken segment. $F_Q^{C3}(X)$ is more discriminative than $F_Q^{C1}(X)$, because it includes more detailed information about neighborhood on both sides. $F_Q^{C4}(X)$, the concatenation of $F_Q^{C2}(X)$ and $F_Q^{C3}(X)$, outperforms the other three context feature representations in almost all cases.

The feedback number N with respect to the best MAP is shown in Table III. The trend from Table II can be seen here as well.

TABLE II: N corresponding to the best MAP in acoustic features

	SI	ADP1	ADP2	SD
N	25	20	20	15
MAP	59.39	71.39	81.63	85.12

TABLE III: N corresponding to the best MAP in linguistic features.

	SI	ADP1	ADP2	SD
N	40	40	35	20
MAP	56.28	69.81	81.44	84.81

TABLE IV: The comparison of MAP (%) between iteratively re-ranking and combination features ($N = 20$).

		SI	ADP1	ADP2	SD
Baseline		45.47	55.54	71.20	80.48
Upper bound after re-ranking		76.08	86.32	90.54	90.31
$F_Q^{state}(X)$	1 iteration	59.31	71.39	81.63	84.66
	10 iterations	62.31	75.88	82.66	85.84
$F_Q^{C4}(X)$	1 iteration	54.93	67.85	80.72	84.81
	10 iterations	60.19	74.46	83.61	85.40
$F_Q^{state}(X)$ + $F_Q^{C4}(X)$	1 iteration	59.92	73.38	83.74	85.73
	10 iterations	**65.31**	**78.42**	**86.26**	**86.89**
Maximum relative improvement (%)		43.66	41.20	21.15	7.96

C. Feature Combinations and Iterative Training

In the above experiments, we used only one type of feature representation in each experiment. Here we combine different types of features, and find that acoustic and linguistic features are in fact complementary. The combination is accomplished as described in (4) and the results are shown in Table IV. The second row of Table IV is the first-pass results and is treated as the baseline. The third row is the upper bound after re-ranking. Because we are considering re-ranking, the segments that were not retrieved in the first pass will never be retrieved. Hence, the upper bound is the best result obtainable when using re-ranking (assuming we are able to re-rank the first pass perfectly). To simplify analysis, $F_Q^{state}(X)$ and $F_Q^{C4}(X)$ are chosen to represent the acoustic and linguistic features respectively, and the feedback number N is set to 20. The fourth to sixth rows show the results of different feature representations, each with two rows: one with only one iteration and another with ten iterations.

$F_Q^{state}(X)$ outperforms $F_Q^{C4}(X)$ in all cases but the SD model with one iteration. Moreover, by using the combination of these two features, the MAP increases slightly in all recognition environments. This implies that acoustic and linguistic features are different domains of knowledge. In iterative training, no matter which feature representation is applied, the MAP improves dramatically compared to one iteration. This includes the combination of $F_Q^{state}(X)$ and $F_Q^{C4}(X)$, for which iterative training yields considerable improvements.

Fig. 6 shows the results of different numbers of iterations using the combination of $F_Q^{state}(X)$ and $F_Q^{C4}(X)$. The MAP improved strongly during the first three iterations, and then seemed to saturate quickly after several iterations.

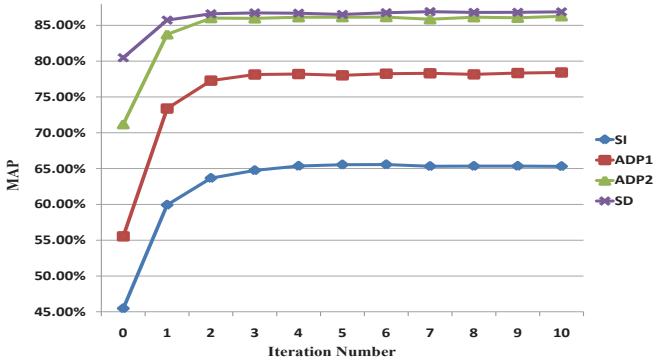


Fig. 6: Iterative training using the combination of $F_Q^{state}(X)$ and $F_Q^{C4}(X)$ with different numbers of iterations.

VII. Conclusion

We proposed multiple feature representations from the acoustic and linguistic domains for SVM training which are then used to re-rank the first-pass results of STD. Both the combination of different features and iteratively re-ranking yield significant improvements. We also considered STD under different recognition environments and found very encouraging results in all cases.

References

[1] D. Wang, S. King, J. Frankel, and P. Bell, "Stochastic pronunciation modelling and soft match for out-of-vocabulary spoken term detection," in *ICASSP*, 2010.

[2] T. Mertens, D. Schneider, and J. Kohler, "Merging search spaces for subword spoken term detection," in *Interspeech*, 2009.

[3] Y.-N. Chen, C.-P. Chen, H.-Y. Lee, C.-A. Chan, and L.-S. Lee, "Merging search spaces for subword spoken term detection," in *ICASSP*, 2011.

[4] C. Chelba, T. Hazen, and M. Saraclar, "Retrieval and browsing of spoken content," in *Signal Processing Magazine, IEEE*, 2008.

[5] C.-H. Meng, H.-Y. Lee, and L.-S. Lee, "Improved lattice-based spoken document retrieval by directly learning from the evaluation measures," in *ICASSP*, 2009.

[6] J. Tejedor, D. T. Toledano, M. Bautista, S. King, D. Wang, and J. Colas, "Augmented set of features for confidence estimation in spoken term detection," in *Interspeech*, 2010.

[7] D. Wang, S. King, J. Frankel, and P. Bell, "Term-dependent confidence for out-of-vocabulary term detection," in *Interspeech*, 2009.

[8] D. Vergyri, I. Shafran, A. Stolcke, R. R. Gadde, M. Akbacak, B. Roark, and W. Wang, "The SRI/OGI 2006 Spoken Term Detection System," in *Interspeech*, 2007.

[9] X. Shen, B. Tan, and C. Zhai, "Context sensitive information retrieval using implicit feedback," in *SIGIR*, 2005.

[10] G. Cao, J.-Y. Nie, J. Gao, and S. Robertson, "Selecting good expansion terms for pseudo-relevance feedback," in *SIGIR*, 2008.

[11] H.-Y. Lee, C.-P. Chen, C.-F. Yeh, and L.-S. Lee, "Improved spoken term detection by discriminative training of acoustic models based on user relevance feedback," in *INTERSPEECH*, 2010.

[12] C.-P. Chen, H.-Y. Lee, C.-F. Yeh, and L.-S. Lee, "Improved spoken term detection by feature space pseudo-relevance feedback," in *INTERSPEECH*, 2010.

[13] H.-Y. Lee, T.-W. Tu, C.-P. Chen, C.-Y. Huang, and L.-S. Lee, "Improved spoken term detection using support vector machines based on lattice context consistency," in *ICASSP*, 2011.