Available online at www.sciencedirect.com

ScienceDirect

Computer Speech and Language 28 (2014) 1045–1065

COMPUTER SPEECH AND LANGUAGE

www.elsevier.com/locate/csl

Improved open-vocabulary spoken content retrieval with word and subword lattices using acoustic feature similarity☆

Hung-yi Lee[a,*], Po-wei Chou[b], Lin-shan Lee[a]

[a] *Graduate Institute of Communication Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei 10617, Taiwan*
[b] *Department of Electrical Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei 10617, Taiwan*

Received 2 August 2012; received in revised form 6 October 2013; accepted 29 December 2013
Available online 5 February 2014

Abstract

Spoken content retrieval will be very important for retrieving and browsing multimedia content over the Internet, and spoken term detection (STD) is one of the key technologies for spoken content retrieval. In this paper, we show acoustic feature similarity between spoken segments used with pseudo-relevance feedback and graph-based re-ranking can improve the performance of STD. This is based on the concept that spoken segments similar in acoustic feature vector sequences to those with higher/lower relevance scores should have higher/lower scores, while graph-based re-ranking further uses a graph to consider the similarity structure among all the segments retrieved in the first pass. These approaches are formulated on both word and subword lattices, and a complete framework of using them in open vocabulary retrieval of spoken content is presented. Significant improvements for these approaches with both in-vocabulary and out-of-vocabulary queries were observed in preliminary experiments.

Keywords: Spoken content retrieval; Spoken term detection; Pseudo-relevance feedback; Random walk

1. Introduction

In the Internet era, digital content over the Internet covers almost all the information and activities of human life. The most attractive form of network content is multimedia, which includes audio signals. The subjects, topics, and core concepts of such multimedia content can very often be identified based on the speech information within the audio part of the content. Hence, in the future, spoken content retrieval will be very important in helping users retrieve and browse efficiently across the huge qualities of multimedia content (Lee and Chen, 2005). Spoken term detection (STD) is a subtask of the above spoken content retrieval, in which the query is a term (a word or a phrase of a few words) in text form and a spoken segment is taken as relevant if it includes the query term. The work of this paper is primarily for STD, although it is certainly possible to generalize the discussions here to other tasks in spoken content retrieval.

Substantial research has been conducted in spoken content retrieval, and many successful techniques have been developed. Lattice-based approaches taking into account multiple recognition hypotheses (Saraclar, 2004; Chelba et al., 2007) have been used to mitigate the relatively low accuracy of 1-best transcriptions. Lattices were usually converted

☆ This paper has been recommended for acceptance by Murat Saraclar.

* Corresponding author. Tel.: +886 963195801.
E-mail address: tlkagkb93901106@yahoo.com.tw (H.-y. Lee).

0885-2308/$ – see front matter © 2014 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.csl.2013.12.003

into sausage-like structures to facilitate indexing and to reduce memory requirements. Examples of such sausage-like lattice-based structures include position-specific posterior lattices (PSPL) (Chelba and Acero, 2005; Pan and Lee, 2007) and confusion networks (CN) (Mangu et al., 2000; Hori et al., 2007; Pan and Lee, 2007). As an alternative, the weighted finite state transducer (WFST) algorithm can also be used to index and retrieve lattices (Allauzen et al., 2004). The out-of-vocabulary (OOV) query is another important issue because queries often contain OOV words (Logan et al., 2000). The most fundamental approach for handling the OOV problem is to represent both the queries and the spoken segments by subword units and then match them on the subword unit level (Akbacak et al., 2008; Pan et al., 2007; Logan et al., 2005; Wallace et al., 2007; Turunen, 2008; Turunen and Kurimo, 2007; Wang et al., 2008; Itoh et al., 2007; Garcia and Gish, 2006; Ng, 2000). Word-based and subword-based indexing can be further integrated to yield better performance (Pan et al., 2007). Many successful applications of spoken content retrieval have been demonstrated including those for broadcast news (Pan et al., 2012), course lectures (Kong et al., 2009; Glass et al., 2007), historical spoken archives (Hansen et al., 2004; Oard et al., 2004), podcasts (Goto et al., 2007), and YouTube videos (Alberti et al., 2009).

In general, there are two stages in conventional spoken content retrieval (Chelba et al., 2008). In the first stage, the audio content is recognized and transformed into transcriptions or lattices by a recognition engine using a set of acoustic models and language models. In the second stage, after the user enters a query, the retrieval engine searches through the recognition output and returns a list of relevant spoken segments to the user. The returned segments are usually ranked by the relevance scores derived from the recognition output. In the above two-stage framework, the spoken content retrieval techniques were actually applied on top of ASR output, either 1-bests or lattices. The performance of spoken content retrieval is thus inevitably limited by ASR performance, and in many cases the ASR performance is still unpredictable today, especially for spontaneous speech produced under adverse environments, and speech produced in the languages with limited resources for acoustic and language model training (Akbacak, 2009). Also, in many application tasks, it is practically very difficult, if not impossible, to obtain acoustic and language models that are robust enough to produce good ASR performance for the huge quantities of target spoken archives which are generated under different acoustic conditions by larger number of speakers for different scenarios with different subject domains. It is not surprising that such mismatched models may result in relatively poor recognition output. In such cases even very robust retrieval approaches are not able to compensate for the recognition errors.

In text-based information retrieval, even if the texts to be retrieved include all precise words, it is still difficult to retrieve precisely all documents relevant to the query. One major reason for this is many queries are too short to completely represent the user's intent. However, related documents may have many words in common; thus if a given document contains many words that also appear in documents judged to be relevant in the first retrieval pass, this document may have a higher probability to be relevant. In other words, it is possible to get better retrieval results by considering the "similarity" (common words) between documents. Pseudo-relevance feedback (PRF) (Kurland et al., 2005; Tao and Zhai, 2006; Cao et al., 2008; Lv and Zhai, 2009, 2010), also known as blind relevance feedback, is one way to accomplish this. PRF assumes that a small number of top-ranked documents in the first-pass retrieved results are relevant (or "pseudo-relevant"), and sometimes also assumes that low-ranked documents are irrelevant (or "pseudo-irrelevant"); the documents retrieved in the first pass are then re-ranked based on their similarity (and dissimilarity) to the pseudo-relevant (and/or -irrelevant) documents.

If the spoken content to be retrieved from can be transcribed into text, the PRF methods developed for text information retrieval can be directly applied on the transcriptions (Zhou, 2003; Lee et al., 2012a). However, since the transcriptions may include many recognition errors, when transcribing speech signals into text, much information is lost and not recoverable. A better approach of utilizing the idea of PRF is not directly applying it on the transcriptions but on the speech signal level with a hope to better use the information carried by the signals (Parada et al., 2009; Chen et al., 2010). The basic idea is if a spoken segment has an acoustic feature vector sequence very similar in some way to those of other spoken segments judged to include the target query term in the first retrieval pass, it may have a higher probability to include the target query term. In this approach, given a user query, the retrieval engine first searches through the lattices to produce a first-pass returned list ranked according to a relevance score directly derived from the lattices. The returned segments with the highest and lowest relevance scores are then respectively defined as the pseudo-relevant and -irrelevant sets. The similarities between each first-pass retrieved spoken segment and the pseudo-relevant and -irrelevant sets can be computed based on the acoustic feature vector sequences of the query hypotheses, and the first-pass returned list is re-ranked accordingly. Furthermore, it is also possible to take the pseudo-relevant and -irrelevant sets respectively as positive and negative examples to train a binary classifier by machine learning

techniques, and then use the binary classifier to determine the relevance of the spoken segments retrieved in the first pass (Lee and Lee, 2013).

The PRF approach can be taken one step further with graph-based re-ranking (Chen et al., 2011). In this approach, for each query entered we construct a graph for the first-pass retrieved spoken segments, in which each node represents a spoken segment and the edges represent the acoustic feature similarity between the segments' query hypotheses. With this graph, the above concept now translates to the concept that segments strongly connected to many segments with higher/lower scores on the graph should have higher/lower scores. The relevance scores for the segments therefore propagate over the graph, and the segments are re-ranked accordingly. In this way, all the spoken segments in the first-pass returned list are considered globally, rather than assuming pseudo-relevant and -irrelevant sets in the PRF approach. For STD utilizing machine learning models to determine the relevance of the spoken segments, similar graph-based concept have been applied for selecting pseudo training examples not restricted to top/bottom-ranked spoken segments in the first-pass retrieved results (Lee and Lee, 2013). These approaches are similar to the very successful PageRank (Langville and Meyer, 2005; Brin and Page, 1998) used to rank web pages; PageRank considers the hyperlink between every two pages and computes a converged importance score for each page. Similar approaches have also been found useful in video search (Hsu et al., 2007; Tian et al., 2008) and extractive summarization (Otterbacher et al., 2009; Lee et al., 2011), in which the similarities between each pair of videos or sentences[1] are respectively used to formulate the ranking problem over graphs.

Although the approaches utilizing acoustic feature similarity in STD were shown to be helpful, in the prior works (Chen et al., 2010, 2011; Tu et al., 2011; Lee and Lee, 2013) they were formulated based on a relatively limited task in which the query includes only a single in-vocabulary (IV) word, and the whole retrieval process was based on word lattices. Here in this paper we focus on a generalized framework for these approaches for a more complete task: the query can be shorter or longer, including one to several words, IV or OOV, and the retrieval is considered on both word- and subword-based lattices. The formulations of the approaches proposed previously (Chen et al., 2010, 2011) are modified to consider the case that the query can include several words or be represented as a sequence of subword units. Furthermore, in this paper it is verified that these approaches can improve the retrieval performance of OOV queries, which has not been investigated before.

Part of the generalized framework was mentioned previously (Lee et al., 2012b), but here the influences of the sizes of pseudo-relevant/-irrelevant sets and different graph construction methods are explored, and deeper analysis based on detection error trade-off (DET) curves and the discussion of time complexity are also included in this paper. Below, the generalized framework for both approaches using PRF and graph-based re-ranking is presented in Section 2. Experiments are presented in Sections 3 and 4, and we conclude in Section 5.

2. Proposed approach

The framework for the proposed approach for the task considered here is shown in Fig. 1. The spoken segments are first transcribed into word or subword lattices by a speech recognizer. When the user enters a query, which can be shorter or longer including IV or OOV words, the retrieval engine searches over the lattices and produces the first-pass returned list as described in Section 2.1. The acoustic feature similarity between every two retrieved segments is then computed as presented in Section 2.2. Based on this similarity, the list is re-ranked using either pseudo-relevance feedback (PRF) in Section 2.3 or graph-based re-ranking in Section 2.4.

2.1. First pass

Given query Q, the system returns the spoken segments x_i with relevance scores $R(x_i, Q)$ higher than a threshold, and ranks these segments according to the values of $R(x_i, Q)$ as the first-pass retrieval result $\mathcal{X}$. The relevance score $R(x_i, Q)$ used to produce the first-pass returned list can be derived from either word or subword lattices. Relevance scores from word lattices are usually more accurate than those from subword lattices, but we must rely on the latter when the query Q consists of OOV words. Below we first show how to determine $R(x_i, Q)$ using word lattices, and then show that the subword-based scores can be similarly obtained from subword-based lattices.

[1] Or utterances for spoken documents.

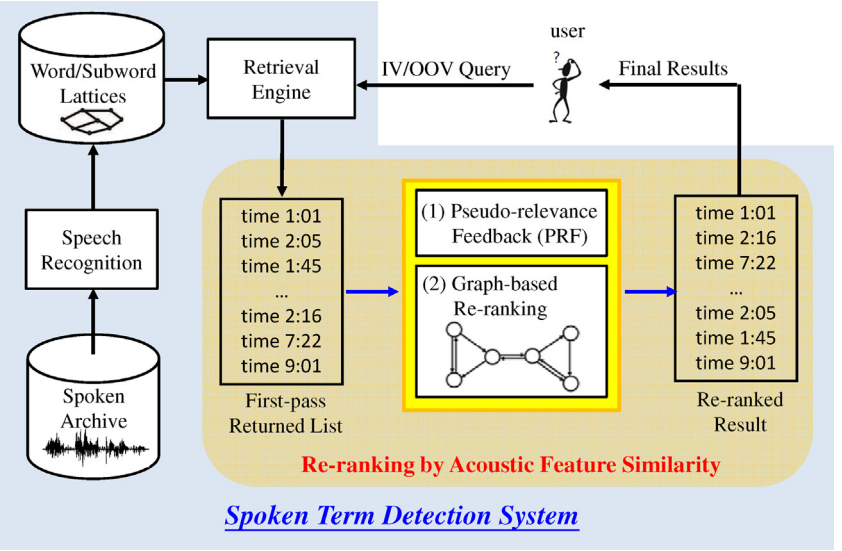

Fig. 1. The generalized framework for spoken content retrieval considering acoustic feature similarity.

We are given the query Q represented as one to several words, $Q_w = \{w_j, j = 1, 2, \ldots, N\}$, w_j being the jth word and N the number of words in Q. To compute the word-based relevance score $R(x_i, Q_w)$ for a segment x_i from the word lattice, we calculate the expected count for each n-gram $\{w_k, \ldots, w_{k+n-1}\}$, $k = 1, \ldots, N-n+1$, in the query from the segment's lattice as in (1), and then aggregate the results for all such n-grams to produce the word-based n-gram score $R_{n\text{-gram}}(x_i, Q_w)$ for each order of n in (2).

$$E[w_k, \ldots, w_{k+n-1}|x_i] = \frac{\sum_{u \in W(x_i)} P(x_i|u)P(u)C(u, \{w_k, \ldots, w_{k+n-1}\})}{\sum_{u \in W(x_i)} P(x_i|u)P(u)}, \tag{1}$$

where $W(x_i)$ is the set of all allowed paths in the lattice of x_i, u one of the allowed paths, $P(x_i|u)$ the likelihood for the observation sequence of x_i given the path u based on the acoustic model set, $P(u)$ the prior probability of u from the language model, and $C(u, \{w_k, \ldots, w_{k+n-1}\})$ the occurrence count of the n-gram $\{w_k, \ldots, w_{k+n-1}\}$ in u, and

$$R_{n\text{-gram}}(x_i, Q_w) = \sum_{k=1}^{N-n+1} E[w_k, \ldots, w_{k+n-1}|x_i]. \tag{2}$$

The different proximity types, one for each n-gram order n allowed by the query length, are finally integrated in a weighted sum to yield word-based relevance score $R(x_i, Q_w)$ for word lattices as

$$R(x_i, Q_w) = \sum_{n=1}^{N} a_n R_{n\text{-gram}}(x_i, Q_w), \tag{3}$$

where a_n is a weight parameter. Since $R(x_i, Q_w)$ here is the aggregation of all the possible n-grams in the query, segments that only partially match the query can still be retrieved; this may increase the recall rate of the retrieval results but not necessary decrease the precision if a_n are properly set (Meng et al., 2009).

To retrieve the subword-based lattices, the query Q is represented as a sequence of subword units instead, $Q_s = \{s_j, j = 1, 2, \ldots, M\}$, where s_j is the jth subword unit and M the number of subword units in Q. The subword-based relevance score $R(x_i, Q_s)$ can be obtained in exactly the same way as that in (1)–(3), except that $E[s_k, \ldots, s_{k+n-1}|x_i]$ is computed on a subword lattice.

$$E[s_k, \ldots, s_{k+n-1}|x_i] = \frac{\sum_{u \in W(x_i)} P(x_i|u)P(u)C(u, \{s_k, \ldots, s_{k+n-1}\})}{\sum_{u \in W(x_i)} P(x_i|u)P(u)}, \tag{4}$$

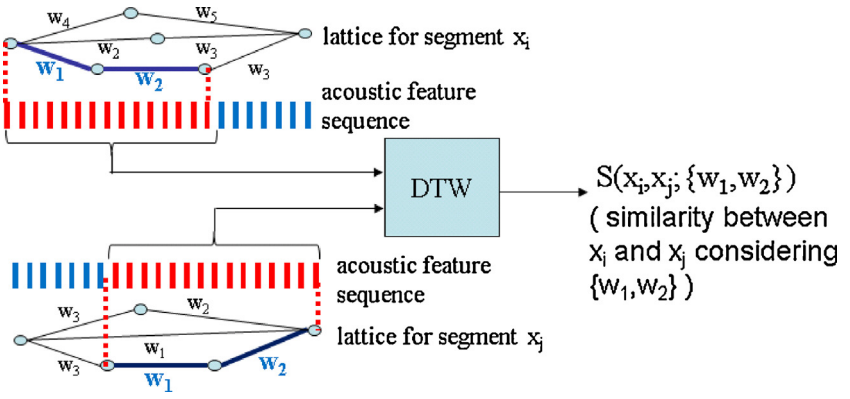

Fig. 2. The computation of $S(x_i, x_j; \{w_1, w_2\})$, the acoustic feature similarity between x_i and x_j considering the 2-gram $\{w_1, w_2\}$.

$$R_{n\text{-gram}}(x_i, Q_s) = \sum_{k=1}^{M-n+1} E[s_k, \ldots, s_{k+n-1}|x_i], \tag{5}$$

$$R(x_i, Q_s) = \sum_{n=1}^{M} a_n{}' R_{n\text{-gram}}(x_i, Q_s). \tag{6}$$

Here (4), (5) and (6) are exactly the same as (1), (2) and (3) except that the word w_j is replaced by the subword unit s_j, $R_{n\text{-gram}}(x_i, Q_s)$ and $R(x_i, Q_s)$ are subword-based n-gram score and subword-based relevance score respectively, and $a_n{}'$ is the corresponding parameter.

2.2. Acoustic feature similarity

Here the similarity $S(x_i, x_j)$ between the acoustic feature vector sequences for two retrieved segments x_i and x_j, referred to as acoustic feature similarity below, is computed, which will be used in both PRF and graph-based re-ranking in the next two subsections. $S(x_i, x_j)$ can be obtained again based on either word or subword units; here we show the word-based version first for demonstration.

Given query Q consisting of a sequence of words $Q_w = \{w_j, j = 1, 2, \ldots, N\}$, for each n-gram $\{w_k, \ldots, w_{k+n-1}\}$ in Q where $k = 1, \ldots, N-n+1$, the dynamic time warping (DTW) distance (Aradilla et al., 2006) is first calculated between the acoustic feature sequences corresponding to the subpaths with word hypothesis sequences $\{w_k, \ldots, w_{k+n-1}\}$ in the lattices of x_i and x_j[2]. An example is shown in Fig. 2. This yields $d(x_i, x_j; \{w_k, \ldots, w_{k+n-1}\})$, the word-based DTW distance between x_i and x_j considering the n-gram $\{w_k, \ldots, w_{k+n-1}\}$ in the query. The word-based similarity between x_i and x_j considering $\{w_k, \ldots, w_{k+n-1}\}$ is then

$$S(x_i, x_j; \{w_k, \ldots, w_{k+n-1}\}) = 1 - \frac{d(x_i, x_j; \{w_k, \ldots, w_{k+n-1}\}) - d_{\min}}{d_{\max} - d_{\min}}, \tag{7}$$

where d_{max} and d_{min} are the largest and smallest values of $d(x_i, x_j; \{w_k, \ldots, w_{k+n-1}\})$ for all pairs of segments in the first-pass returned list. Eq. (7) simply linearly normalizes the DTW distance and transforms it into a similarity score between 0 and 1. If the n-gram $\{w_k, \ldots, w_{k+n-1}\}$ does not exist in the lattice of either x_i or x_j,

[2] If there are multiple subpaths whose word hypotheses are $\{w_k, \ldots, w_{k+n-1}\}$ in a lattice, only the one with the highest posterior probability is considered. Instead of picking the subpaths with the highest posterior probability, there are other reasonable alternatives. For example, first cluster the subpaths with the same hypothesis sequences and similar time spans into groups. Use the time span of the subpath with the highest posterior probability in each group to represent the time span of the group, and take the summation of the posterior probabilities of all the elements in a group as its score. Then use the time span of the group of subpaths $\{w_k, \ldots, w_{k+n-1}\}$ with the highest score to compute the DTW distances. However, this alternative did not lead to too much difference from picking the highest subpaths in terms of the experimental results, but required extra computing efforts, so for simplicity we do not report its results here.

$S(x_i, x_j; \{w_k, \ldots, w_{k+n-1}\})$ is set to 0. We then aggregate the similarities considering all such n-grams to produce word-based score $S_{n\text{-gram}}(x_i, x_j; Q_w)$ for each order of n as

$$S_{n\text{-gram}}(x_i, x_j; Q_w) = \sum_{k=1}^{N-n+1} S(x_i, x_j; \{w_k, \ldots, w_{k+n-1}\}). \tag{8}$$

The different proximity types are finally integrated as a weighted sum to yield the word-based similarity between x_i and x_j:

$$S(x_i, x_j; Q_w) = \sum_{n=1}^{N} b_n S_{n\text{-gram}}(x_i, x_j; Q_w), \tag{9}$$

where b_n is another weight parameter.

To retrieve subword-based lattices, the query Q is represented as a sequence of subword units instead, $Q_s = \{s_j, j = 1, 2, \ldots, M\}$. The computation of subword-based similarity $S(x_i, x_j; Q_s)$ is exactly the same as that in (7)–(9), except that the word sequence Q_w is replaced by Q_s, and each word w_i is replaced by subword unit s_j.

$$S(x_i, x_j; \{s_k, \ldots, s_{k+n-1}\}) = 1 - \frac{d(x_i, x_j; \{s_k, \ldots, s_{k+n-1}\}) - d'_{\min}}{d'_{\max} - d'_{\min}}, \tag{10}$$

$$S_{n\text{-gram}}(x_i, x_j; Q_s) = \sum_{k=1}^{M-n+1} S(x_i, x_j; \{s_k, \ldots, s_{k+n-1}\}), \tag{11}$$

$$S(x_i, x_j; Q_s) = \sum_{n=1}^{M} b_n{}' S_{n\text{-gram}}(x_i, x_j; Q_s). \tag{12}$$

Here (10), (11) and (12) are exactly the same as (7), (8) and (9), except that the word sequence Q_w is replaced by Q_s, and each word w_i is replaced by subword unit s_j, and $d_{max}{}'$, $d_{min}{}'$ and $b_n{}'$ are the corresponding parameters.

Although we can obtain the relevance score $R(x_i, Q_w)$ and $R(x_i, Q_s)$, and similarity $S(x_i, x_j; Q_w)$ and $S(x_i, x_j; Q_s)$ based on different units and use them together – for example, it is possible to derive $R(x_i, Q_w)$ in (3) from word lattices but compute $S(x_i, x_j; Q_s)$ in (12) on subword lattices and use them together – for simplicity in the experiments below, we always use $R(x_i, Q_w)/R(x_i, Q_s)$ and $S(x_i, x_j; Q_w)/S(x_i, x_j; Q_s)$ obtained from the same type (word or subword) of lattices together. Below for simplicity in notation, we simply use $R(x_i, Q)$ to denote relevance score and $S(x_i, x_j)$ to denote similarity, regardless of whether they are obtained from word or subword lattices.

Although we here report only experiments using MFCC features for the DTW distances, other acoustic features could be used, such as phone posteriorgrams (Hazen et al., 2009) or Gaussian posteriorgrams (Zhang and Glass, 2010, 2009), and evaluating the acoustic similarity between two acoustic feature sequences based on models is even preferred (Chan and Lee, 2011; Wang et al., 2012). These approaches may provide less speaker-dependent DTW distance measures and could thus be useful if the target spoken segments are produced by many different speakers.

2.3. Pseudo-relevance feedback (PRF)

In PRF, the top-ranked y segments with the highest relevance scores $R(x_i, Q)$ in (3) or (6) are selected as pseudo-relevant set $\mathcal{Y}$; the bottom-ranked z segments with the lowest $R(x_i, Q)$ are selected as pseudo-irrelevant set $\mathcal{Z}$. The similarity between each segment x_i in the first-pass retrieved results $\mathcal{X}$ and the pseudo-relevant and -irrelevant sets is then defined as

$$SIM(x_i) = \frac{1}{y}\sum_{x \in \mathcal{Y}} S(x_i, x) - \frac{1}{z}\sum_{x \in \mathcal{Z}} S(x_i, x), \tag{13}$$

where y and z are the sizes of the pseudo-relevant and -irrelevant sets.

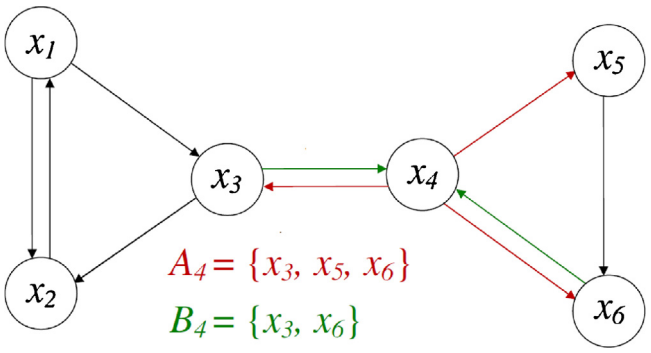

Fig. 3. A simplified example of a graph, the nodes of which correspond to spoken segments. The edge weights are acoustic feature similarities between the nodes. A_i and B_i are the node sets connected respectively by the outgoing and incoming edges of x_i.

The value of $SIM(x_i)$ is then linearly normalized into a number between 0 and 1 as $SIM'(x_i)$,

$$SIM'(x_i) = \frac{SIM(x_i) - S_{\min}}{S_{\max} - S_{\min}}, \tag{14}$$

where S_{max} and S_{min} are the largest and smallest values of $SIM(x_i)$ among all spoken segments x_i retrieved.

The relevance score $R(x_i, Q)$ for each segment x_i is then updated into a new relevance score

$$R_p(x_i, Q) = R(x_i, Q)^{1-\delta_1} SIM'(x_i)^{\delta_1}, \tag{15}$$

where δ_1 is a weight parameter between 0 and 1. The segments in $\mathcal{X}$ are then re-ranked according to $R_p(x_i, Q)$, and then displayed to the user.

Usually in the literature the sizes of the pseudo-relevant/-irrelevant objects $\mathcal{Y}$ and $\mathcal{Z}$ are fixed for all of the queries. However, in text information retrieval, it has been found that the optimal number of pseudo-relevant objects varies from query to query (Montgomery et al., 2004). This observation suggests $\mathcal{Y}$ and $\mathcal{Z}$ should be determined dynamically for different queries based on their properties, which is not an easy task.[3] The graph-based re-ranking presented in the next subsection may achieve this goal to some extent. Instead of selecting a set of pseudo-relevant/-irrelevant spoken segments, in the graph-based re-ranking, the contribution of a spoken segment to the final scores of other segments is based on the acoustic similarity structure between the segments in the first-pass retrieved result (represented as a graph). For instance, the spoken segments ranked in the first place of the first-pass retrieved results of two different queries have different influences to the other spoken segments in the retrieved results because the acoustic similarity structures are different for the two first-pass results.

2.4. Graph-based re-ranking

An alternative to PRF for using acoustic feature similarity is graph-based re-ranking, which involves first constructing a graph for the first-pass retrieved segments for each query (Section 2.4.1) and then applying a random walk for relevance score propagation over the graph (Section 2.4.2).

2.4.1. Graph construction

Here for each query a directed graph is constructed from the first-pass returned list $\mathcal{X}$, in which each node represents a segment. A simplified example for such a graph is shown in Fig. 3. Because directions are needed for score propagation over the graph, the edges between nodes need to have directions. There can be at least several approaches for connecting the edges with directions for the graph. In the first two cases below, we first connect each pair of nodes for segments

[3] In the preliminary experiments, we have investigated different alternatives for dynamically determining the size of pseudo-relevant set for each queries. For example, take the segments whose relevance scores higher than a threshold as pseudo-relevant. In this way, different queries have different pseudo-relevant sets with different sizes. However, due to the diverse properties of the queries, these alternatives did not outperform that of simply taking a fixed number of top-ranked segments as pseudo-relevant, so they are not reported here.

x_i and x_j with a pair of edges in both directions ($x_i \rightarrow x_j$ and $x_i \leftarrow x_j$). The weight of edge from x_i to x_j ($x_i \rightarrow x_j$) is $S(x_i, x_j)$ in (9) or (12).[4] We then prune those edges with lower weights in two different ways as listed below.

- *Fixed Number of Outgoing Edges* (OUT): Each segment (or node) x_i only keeps the K outgoing edges with the highest weights. In this way, each node in the graph has a fixed number K of outgoing edges but a variable number of incoming edges. In other words, during the score propagation, each node influences equal number of nodes.
- *Fixed Number of Incoming Edges* (IN): Each segment (or node) x_i only keeps the K incoming edges with the highest weights. Thus each node in the graph has a fixed number K of incoming edges but a variable number of outgoing edges. In other words, the score of each node is influenced by equal number of nodes during the score propagation.

In the next two cases below, the two edges connecting nodes x_i and x_j in both directions ($x_i \rightarrow x_j$ and $x_i \leftarrow x_j$) are kept or deleted jointly, so all edges existing on the graph have both directions.

- *K-nearest Neighbor* (KNN): Nodes x_i and x_j are connected to each other in both directions ($x_i \rightarrow x_j$ and $x_i \leftarrow x_j$) if x_i is among the K-nearest neighbors of x_j (or given x_j, $S(x_i, x_j)$ is among the K highest of all x_i), or if x_j is among the K-nearest neighbors of x_i (or given x_i, $S(x_i, x_j)$ is among the K highest of all x_j).
- *Mutual K-nearest Neighbor* (M-KNN): Similar as the above, except both directions of K-nearest neighbor relationships are required. Nodes x_i and x_j are connected to each other in both directions if x_i is among the K-nearest neighbors of x_j, and x_j is among the K-nearest neighbors of x_i.

In the graphs generated by the first two approaches, it is possible that two nodes are connected by edges in only one direction. The last two approaches, however, lead to symmetric graphs, or two nodes are always connected with edges of both directions.

2.4.2. Random walk

A new set of graph-based relevance scores $R_g'(x_i, Q)$ for all x_i in the first-pass returned list $\mathcal{X}$ can be obtained via score propagation on the graph, which can be expressed as

$$R_g'(x_i, Q) = (1-\alpha)R(x_i, Q) + \alpha \sum_{x_j \in B_i} R_g'(x_j, Q)\hat{S}(x_j, x_i), \tag{16}$$

where $R(x_i, Q)$ is the relevance score in (3) or (6), α is an interpolation weight between 0 and 1, B_i is the set of all segments connected to x_i by incoming edges as in Fig. 3, and x_j is a node in B_i. $\hat{S}(x_j, x_i)$ is the normalized edge weight $S(x_j, x_i)$ over all edges outgoing from node x_j on the graph[5]:

$$\hat{S}(x_j, x_i) = \frac{S(x_j, x_i)}{\sum_{x_k \in A_j} S(x_j, x_k)}, \tag{17}$$

where A_j is the set of segments connected to x_j by outgoing edges as in Fig. 3. In (16) the graph-based score $R_g'(x_i, Q)$ of a segment x_i depends on two factors interpolated by α: the relevance score in (3) or (6) (the first term on the right hand side of (16)) and the score propagation over the graph from all nodes x_j in B_i to x_i via incoming edges based on the normalized edge weights $\hat{S}(x_j, x_i)$ (the second term on the right hand side).

Based on (16), a segment x_i would have large $R_g'(x_i, Q)$ under the following two conditions:

1. Original relevance score $R(x_i, Q)$ is large, or the confidence of the occurrence of the query in x_i is high based on its lattice.
2. x_i is connected to other nodes x_j with large $R_g'(x_j, Q)$, or x_i is acoustically similar to other spoken segments x_j with larger probabilities of containing the query.

[4] Because $S(x_i, x_j) = S(x_j, x_i)$, the edges connecting x_i and x_j in both directions ($x_i \rightarrow x_j$ and $x_i \leftarrow x_j$) have equal weights.

[5] Note that $\hat{S}(x_j, x_i) \neq \hat{S}(x_i, x_j)$.

The normalization in (17) formulates (16) as a random walk problem on the graph; random walk theory guarantees that a set of unique solutions of $R_g'(x_i, Q)$ can be found. For all retrieved spoken segments x_i, $R_g'(x_i, Q)$ in (16) can be found efficiently by power method (Langville and Meyer, 2005).[6]

Each node x_i is first given an initial value $R_g^0(x_i, Q)$.[7] Then at each iteration t, $R_g^{t-1}(x_i, Q)$ obtained in the last iteration are updated to $R_g^t(x_i, Q)$ as below:

$$R_g^t(x_i, Q) = (1-\alpha)R(x_i, Q) + \alpha \sum_{x_j \in A_i} R_g^{t-1}(x_j, Q)\hat{S}(x_j, x_i). \tag{18}$$

Eq. (18) is parallel to (16), except that $R_g^t(x_i, Q)$ is at the left hand side of the equation, and $R_g^{t-1}(x_j)$ at the right hand side. Whenever the results converge, that is, $R_g^{t-1}(x_i, Q)$ and $R_g^t(x_i, Q)$ are sufficiently close, $R_g^t(x_i, Q)$ can be taken as the scores $R_g'(x_i, Q)$ satisfying (16).

$R_g'(x_i, Q)$ is finally integrated with $R(x_i, Q)$ to become a new relevance score for re-ranking,

$$R_g(x_i, Q) = R(x_i, Q)^{1-\delta_2} R_g'(x_i, Q)^{\delta_2}, \tag{19}$$

where δ_2 is a parameter between 0 and 1. The final retrieval results ranked according to $R_g(x_i, Q)$ in (19) are then displayed to the user.[8]

2.4.3. Complexity of graph-based re-ranking

There are two stages in graph-based re-ranking: graph construction in Section 2.4.1 and random walk in Section 2.4.2. In the following, the complexity of these two stages is analyzed.

During the graph construction in Section 2.4.1, the system first constructs a fully connected graph, and then prunes the edges with low weights. Suppose the number of spoken segments retrieved in the first pass is G, or there are G nodes in the graph. The system should compute the distances between the G nodes, or the weights of $G(G-1)$ edges. G is usually small because the number of segments retrieved in the first pass is usually limited, although there are a large amount of spoken segments in the spoken archive. In real implementation, it is possible to construct the graph in more effective way. The system can use the coarse but fast approaches (Jansen and Durme, 2012) to first compute the approximate weights for all of the $G(G-1)$ edges in the fully connected graph to decide the edges to be pruned, and then use the fine but slow ways to exactly compute the weights of the remaining edges.

The complexity of the random walk in Section 2.4.2 is low. For each iteration in power method, there are G equations like (18) for every x_i, and in general each equation has at most G additions (because the number of elements in A_i can never exceed G). Therefore, if power method has T iterations, the complexity of random walk is less than $O(G^2T)$. G is a small number as described in the last paragraph, and T is also small because empirically tens of iterations is sufficient for the power method to converge (even if there are millions of nodes in a graph) (Langville and Meyer, 2005). In fact, in the experiments here, $O(G^2T)$ excessively overestimates the complexity. For example, based on the graph construction with *Fixed Number of Incoming Edges* (IN) in Section 2.4.1, because the size of A_i is always K, there are only $K+1$ additions in (18), and K is usually much smaller than G. Therefore, one of the G in $O(G^2T)$ should be replaced by K, and thereby the complexity of random walk is only $O(KGT)$ in such case. There are other approaches to speed power method (Kamvar et al., 2003; Manaskasemsak and Rungsawang, 2005), but it is out of the scope here.

3. Experimental setup

The testing spoken archive is a corpus of 45 h of recorded lectures for a course offered at National Taiwan University taught by a single instructor; the corpus is quite noisy and spontaneous (Lee et al., 2009).

[6] It is also possible to obtain $R_g'(x_i, Q)$ in (16) by searching for the eigenvalues of a matrix, but this approach has much larger time complexity than power method.

[7] The initial values would not influence the final results (Meye, 2000).

[8] Although the original scores $R(x_i, Q)$ have been considered when computing $R_g'(x_i, Q)$ in (16), integrating $R(x_i, Q)$ and $R_g'(x_i, Q)$ again in (19) empirically lead to better performance. Because $R(x_i, Q)$ and $R_g'(x_i, Q)$ are added in (16) but multiplied in (19), considering both integration mechanism leads to optimal performance.

The spoken archive was divided into about 23,000 spoken segments based on silences, and the lengths of the segments were 3.6 s on average.

The lectures were given primarily in Mandarin Chinese but with some English terms and phrases embedded within the Mandarin utterances.

Those embedded English words or phrases are usually very short, very often with a length of only one to three words. The English words or phrases occurred in the utterances in the following two conditions. In the first case, almost all terminologies for this course are directly produced by the instructor in English without trying to translate them into Chinese. For example, in the utterance, "除了 speech recognition 的技術之外, 我們還需要 indexing 跟 retrieval 的技術" (Except for speech recognition technology, we also need technologies about indexing and retrieval.)", the phrase "speech recognition" and the words "indexing" and "retrieval" were produced in English, while other parts of the utterance are in Mandarin. In the second case, the instructor may prefer to use some commonly used English words in his utterances, which are not terminologies at all, probably because the concept can be easily or naturally expressed in English in this way. For example, in the utterance, "我可以 somehow handle 這個問題 (I can somehow handle this problem)", the words "somehow handle" were produced in English, but other parts of the utterance were in Mandarin.

The ratio of the number of Mandarin characters to that of English words is nine to one in the spoken archive we used. Many course lectures are presented in this code-switching way in Taiwan. In fact, such code-switching speech is very common for speakers whose native languages are not English but speak fluent English in the daily lives. They naturally speak the native languages as the daily language, but spontaneously embed some English words in their native language utterances. For example, many Asian whose native languages are not English speak in this way. Hence, the task domain considered here is very important and representative, although not yet investigated extensively.

We split the corpus into two parts: 12 hours for acoustic and language model training and 33 h for retrieval testing.

In the following experiments, mean average precision (MAP) (Garofolo et al., 2000) was used as the retrieval performance measure. The pair-wise t-test with a significance level of 0.05 was used to gauge the significance of performance improvements. Here the STD system only returns the spoken segments containing the query terms without locating their exact positions in the spoken archive, because the spoken segments here were short enough to be used as the pointer to the positions. This task definition is the same as the STD task in the 9th NTCIR workshop (Akiba et al., 2011), but slightly different from that in NIST 2006 (http://www.itl.nist.gov/iad/mig/tests/std/2006/index.html), in which the positions of the query terms in the spoken archive should be located. For computing the DTW distances in Section 2.2, MFCCs were used as the acoustic features, and Euclidean distance was applied as the distance measure between two acoustic features. Some parameters in the experiments were set empirically as below. a_n and $a_n{}'$ in (3) and (6) were both set to 10^{5n} to favor longer n-grams. The expected term frequencies of longer n-grams should have more influence on the relevance scores in (3) and (6) because the observation of a query's longer n-grams in the lattices provides more confidence about the existence of the query than shorter n-grams. Due to the same reason, b_n and $b_n{}'$ in (9) and (12) were set equal to a_n and $a_n{}'$.

The influence of δ_1 in (15), δ_2 in (19) and α in (16) has been explored in previous studies on the same audio but with another query set (Chen, 2011; Chen et al., 2011). In the previous studies (Chen, 2011), larger δ_1 and δ_2 implied better results unless they were too close to 1 (for example, larger than 0.99) because $SIM'(x_i)$ in (14) and $R_g^t(x_i, Q)$ in (16) were more reliable than the original relevance scores $R(x_i, Q)$. Therefore, δ_1 and δ_2 were both set to 0.9 here. For graph-based re-ranking, α close to 1 was optimal for not only spoken term detection (Chen et al., 2011) but also video search (Hsu et al., 2007), so α was set to 0.9 here.

In order to evaluate the retrieval performance with respect to acoustic models of different matched conditions, we used three sets of acoustic models:

- Speaker-independent models (SI) trained on a Mandarin corpus of 24.6 h of read speech, produced by 100 male and 100 female speakers, plus the Sinica L2 Taiwanese English corpus with 59.7 h of English read speech, produced by 229 male and 256 female Taiwanese speakers.
- Speaker-adaptive models (SA) adapted by MLLR with 256 classes cascaded with the maximum a posterior estimation from the above SI model based on 500 utterances taken from the training set of the lecture corpus.
- Speaker-dependent models (SD) trained on the 12-h training set of the lecture corpus.

For all three sets of acoustic models, we trained a set of state-tied triphone models spanned from 37 Mandarin monophones and 35 English monophones based on the recently developed state mapping and recovery techniques (Yeh et al., 2011).

Two sets of experiments respectively with in-vocabulary (IV) and OOV queries were performed as mentioned below.

3.1. IV query experiments

The IV query set included 275 Chinese queries, each composed of 1–3 words, or 2–7 Chinese characters. The number of relevant spoken segments for each IV query ranged from 5 to 714 with an average of 38.2. In the experiments here, a language model trained with the manual transcriptions of the training set of the lecture corpus was used. A close-to-oracle lexicon was used which included 11K Chinese words plus 2K English words covering all words in the testing archive. Each utterance was transcribed into a bilingual word lattice. Then we transformed each Chinese word arc into a sequence of concatenated corresponding Chinese character and Mandarin syllable arcs to respectively form character and syllable lattices. The English word arcs remained unchanged. Therefore, for each utterance there were three lattices: word-, character-, and syllable-based. The word recognition accuracies for Chinese characters and English words evaluated together were 49.7%, 80.8%, and 88.0% respectively for the SI, SA, and SD models, and the inclusion rates of the lattices[9] were 72.7%, 86.8%, and 92.0% respectively for the SI, SA, and SD models. Note that the three different sets of acoustic models together with the relatively matched language model and lexicon gave different levels of recognition accuracies. In this way, we wish to show that the proposed approaches can offer performance improvements regardless of whether the recognition accuracies are lower or higher.

3.2. OOV query experiments

For the OOV query set we used 110 English queries, each consisting of a single word. The number of relevant spoken segments for each OOV query ranged from 2 to 268 with an average of 39.8. First, we assume we do not know the pronunciation of these OOV words, so we trained a 6-gram joint-sequence model from the CMU dictionary with 130K words (http://www.speech.cs.cmu.edu/cgi-bin/cmudict) to be used as the grapheme-to-phoneme converter to predict the pronunciations for the OOV queries (Bisani and Ney, 2008).[10] The canonical pronunciation for each OOV query was also used in the experiments for comparison. Using the canonical pronunciation as the reference, the pronunciation was estimated perfectly (exactly the same as the reference) for 81 of the 110 OOV queries, or with an accuracy of 73.6%, while the pronunciation estimation accuracies on syllable and phoneme levels were 85.8% and 93.8% respectively.

We used a word/subword hybrid system to transcribe each spoken segment, which is a widely used approach for handling the OOV problem (Rastrow et al., 2009; Akbacak et al., 2008; Szoke et al., 2008). In this experiment, we used a lexicon composed of 11K Chinese words, 5K English words from the standard Aurora-4 lexicon,[11] and 10K English syllables automatically generated from the CMU dictionary based on some syllable segmentation rules. 20,000 English documents from the 20Newsgroups corpus[12] were then used to train an English language model based on the above lexicon of 5K English words and 10K English syllables, which included mixed trigrams for English words and syllables (for example, a word following two concatenated syllables can be a trigram item). In other words, those words in the English training documents but not within the above selected lexicon of 5K English words were segmented into syllable sequences to be used together with the other words in the 5K English word lexicon to train an English trigram language model for mixed words and syllables. A Chinese word-based trigram language model was trained on the lecture corpus training set. These two language models were then interpolated to produce the lattices composed of a mixture of arcs for Chinese words, English words, and English syllables. We further substituted the Chinese and English word arcs in the lattices with their corresponding syllables to obtain a set of syllable-based lattices. Thus for each spoken segment we generated two lattices: one composed of Chinese and English words plus English syllables,

[9] The highest accuracy among the path hypotheses in each lattice.
[10] The terms used in OOV queries were excluded from the CMU dictionary during training.
[11] This lexicon is quite disjoint from the content of the target corpus for retrieval, so the 110 English queries were not included in this lexicon.
[12] http://people.csail.mit.edu/jrennie/20Newsgroups/.

and the other composed solely of Chinese and English syllables. Because the English recognition accuracies for SI and SA were not good enough to offer reasonable results, we used only SD models for the OOV query experiments. Since the accuracy for the word/syllable hybrid recognition output is not easy to define, we only evaluated the English syllable accuracy here. The English syllable accuracy of the one-best transcriptions and English syllable inclusion rate on the lattices were respectively 43.6% and 59.5% for the SD models.

4. Experimental results

Sections 4.1, 4.2 and 4.3 are the results for the IV queries. In Section 4.1, we demonstrate the performance of PRF with different numbers of pseudo-relevant/-irrelevant spoken segments. In Section 4.2, we investigate different graph construction approaches for graph-based re-ranking, and the results of PRF and graph-based re-ranking for word lattices are compared. In Section 4.3, PRF and graph-based re-ranking were applied on the subword-based lattices, and the results of word and subword units were further integrated. Finally, to investigate the usefulness of PRF and graph-based re-ranking for OOV queries, the results of OOV queries are reported in Section 4.4.

4.1. Pseudo-relevance feedback (PRF) for word lattices with IV queries

Table 1 shows the MAP performance for word lattices with IV queries yielded by PRF with different numbers of pseudo-relevant segments (different y in (13)) and 40 pseudo-irrelevant segments ($z = 40$ in (13)). The first-pass retrieval results are taken as the baseline. The three columns SI, SA and SD correspond to the three sets of acoustic models with different quality. The superscript * indicates significantly better than the baselines. First of all, we found that PRF outperformed the baselines except when $y = 1$. We also observed that as the number of pseudo-relevant segments was raised the MAP first increased and then decreased. This is reasonable because a larger y implies that more segments are considered when computing the similarities, and therefore disturbances caused by noisy pseudo-relevant segments (irrelevant segments assumed to be relevant) are diluted. However, when y was too large, more irrelevant segments were inevitably included in the pseudo-relevant segment set, and the MAP naturally degraded.

Because for the IV queries tested here the number of relevant segments ranged from 5 to hundreds, the pseudo-relevant sets including more than 10 segments inherently cover more than 50% incorrect examples for some queries. This may explains why the optimal y for the three recognition conditions, SI, SA and SD, was all around 10.

Notwithstanding we did not develop techniques to automatically decide the optimal y, PRF is still an useful approach because no matter the value of y, significant improvements were always observed (except $y = 1$).

Table 2 shows similar MAP performance as those in Table 1 with the number of pseudo-relevant segments fixed at 9 ($y = 9$ in (13)) but with different numbers of pseudo-irrelevant segments (different z in (13)). The superscript * indicates significantly better than the baselines. In contrast to Table 1, we observed that as the number of pseudo-irrelevant segments was raised the MAP first increased and then saturated without too much degradation. This may be

Table 1
MAP performance of PRF for word lattices with IV queries with different numbers of pseudo-relevant segments and 40 pseudo-irrelevant segments. The first-pass retrieval results are considered as the baselines. SI, SA, and SD correspond to the three sets of acoustic models.

Acoustic model		SI	SA	SD
Baseline		0.5596	0.7956	0.8424
Number of pseudo-relevant segments	1	0.5972*	0.7946	0.8392
	3	0.6146*	0.8126*	0.8529*
	5	0.6199*	0.8204*	0.8583*
	7	0.6223*	0.8216*	0.8605*
	9	0.6235*	0.8220*	0.8601*
	11	0.6208*	0.8205*	0.8611*
	13	0.6219*	0.8192*	0.8606*
	15	0.6185*	0.8172*	0.8591*
	17	0.6157*	0.8157*	0.8574*
	19	0.6136*	0.8149*	0.8557*

* Significantly better than the baselines.

Table 2
MAP performance of PRF as those in Table 1, except with 9 pseudo-relevant segments but different numbers of pseudo-irrelevant segments.

Acoustic model		SI	SA	SD
Baseline		0.5596	0.7956	0.8424
Number of pseudo-irrelevant segments	5	0.6083*	0.8174*	0.8557*
	10	0.6150*	0.8194*	0.8569*
	15	0.6197*	0.8196*	0.8580*
	20	0.6222*	0.8215*	0.8579*
	25	0.6228*	0.8221*	0.8602*
	30	0.6234*	0.8225*	0.8599*
	35	0.6235*	0.8222*	0.8603*
	40	0.6235*	0.8220*	0.8601*
	45	0.6235*	0.8218*	0.8600*

* Significantly better than the baselines.

because the irrelevant segments form the majority of the retrieved segments, so most pseudo-irrelevant segments are truly irrelevant even when we took very large number of them.

In this section, the experimental results show that PRF was helpful for the IV queries with one or several words, and the influence of the sizes of pseudo-relevant/-irrelevant sets, y and z, was investigated.

4.2. Graph-based re-ranking for word lattices with IV queries

Table 3 shows the results of graph-based re-ranking started with word lattices yielded by the different graph construction strategies of Section 2.4.1 with the IV query set and SI models. The four columns of the table correspond to the *Fixed Number of Outgoing* (OUT) and *Incoming* (IN) *Edges*, *K-nearest Neighbor* (KNN), and *Mutual K-nearest Neighbor* (M-KNN); K in Table 3 is the number of outgoing edges for OUT, incoming edges for IN, or the number of nearest neighbors considered for KNN and M-KNN. The best results in each column are in bold. Clearly *Fixed number of Incoming Edges* (IN) is the best graph construction strategy. Consider Eq. (16). We mentioned that a spoken segment x_i can have larger $R'_g(x_i, Q)$ in (16) if x_i's first-pass score $R(x_i, Q)$ is higher (the first term in (16)), or if the nodes connected by incoming edges to x_i have large scores $R_g{}'(x_j, Q)$ (the second term in (16)). There is however another implicit factor with the latter which may influence the score of $R_g{}'(x_i, Q)$: given a large number of nodes connected by incoming edges to x_i (or a large set B_i), even though the scores of the individual nodes in B_i are small, $R_g{}'(x_i, Q)$ can still become high. In our task, this phenomenon is undesirable because an irrelevant segment x_i may be somehow similar to many other irrelevant segments, and that may result in a higher score $R_g{}'(x_i, Q)$ through the many incoming

Table 3
MAP performance of graph-based re-ranking started with word lattices yielded by the four different graph construction strategies as summarized in Section 2.4.1: *Fixed Number of Outgoing* (OUT) or *Incoming* (IN) *Edges*, *K-nearest Neighbor* (KNN) or *Mutual K-nearest Neighbor* (M-KNN), with the IV query set and SI models. K is the number of outgoing edges for OUT, incoming edges for IN, or the number of nearest neighbors considered for KNN and M-KNN. The best results in each column are in bold.

Graph construction method	OUT	IN	KNN	M-KNN
$K=1$	0.5885	0.5873	0.5286	0.5283
$K=2$	0.6303	0.6463	0.5395	0.5641
$K=3$	0.6388	0.6679	0.5496	0.5940
$K=4$	**0.6400**	0.6753	0.5617	0.6111
$K=5$	0.6394	**0.6783**	**0.5640**	0.6266
$K=10$	0.6254	0.6699	0.5620	**0.6469**
$K=15$	0.6132	0.6612	0.5548	0.6460
$K=20$	0.6029	0.6550	0.5453	0.6416

Table 4
MAP performance as those in Table 3 for graph-based re-ranking for *Fixed Number of Incoming Edges* (IN) with different values of K using different sets of acoustic models. The best results in each column are in bold.

Acoustic model		SI	SA	SD
Baseline		0.5596	0.7956	0.8424
PRF		0.6261*	0.8239*	0.8621*
Graph	$K=1$	0.5873*	0.7884	0.8315
	$K=2$	0.6463*,†	0.8127*	0.8566*
	$K=3$	0.6679*,†	0.8239*	0.8666*
	$K=4$	0.6753*,†	0.8281*	0.8690*
	$K=5$	**0.6783***,†	0.8328*	0.8711*,†
	$K=10$	0.6699*,†	**0.8337***,†	**0.8717***,†
	$K=15$	0.6612*,†	0.8301*	0.8700*,†
	$K=20$	0.6550*,†	0.8271*	0.8678*,†

* Significantly better than the baselines.
† Significantly better than the PRF.

edges from many other irrelevant segments.[13] By fixing the size of B_i, *Fixed number of Incoming Edges* (IN) solved this problem, yielding the best results in our task here; it is therefore used in all following experiments.

Table 4 shows the results of graph-based re-ranking using the graph with *Fixed Numbers of Incoming Edges* (IN) for different values of K with three different sets of acoustic models, one for each column. The graphs constructed for SI, SA and SD models had respectively 313.4, 171.1 and 146.7 nodes on average, that is, there were 313.4, 171.1 and 146.7 spoken segments respectively in the first pass for SI, SA and SD models on average. Because in Table 4 the number of incoming edges for each node was fixed to K, the number of edges in a graph was simply K times the number of nodes. The superscripts * and † respectively indicate significantly better than the baselines and PRF. The best results of PRF are reported here in the second row, in which the numbers of pseudo-relevant (y) and -irrelevant (z) segments used were tuned to maximize the MAP values on the testing query set, resulting in unrealistically high performance for PRF (higher than all numbers in Tables 1 and 2).[14] The results in Table 4 are represented in Fig. 4 as well. The blue and red lines are respectively for the first-pass retrieval results and PRF, and the green curves are for graph-based re-ranking. The horizontal scales in the figures are the numbers of incoming edges K. Fig. 4(a), (b) and (c) are respectively for SI, SA and SD models. From Table 4 and Fig. 4, we found that graph-based re-ranking outperformed the baseline in all cases except when $K=1$. We also found that graph-based re-ranking was so powerful that even though the parameters for PRF were carefully tuned, graph-based re-ranking still outperformed PRF significantly if K was large enough.

In graph-based re-ranking, it is ideal if the nodes representing relevant segments only connect to other relevant segments. If K was too large, since the number of relevant segments is limited for each query, a relevant spoken segments would unavoidably connect to some irrelevant segments. Because some queries only had 5 relevant segments, when K was larger than 10, for such queries the relevant segments had to connect to more nodes representing irrelevant segments than the relevant ones. This may be why in Table 4 the optimal K for different models ranged from 5 to 10.

Then we analyzed the execution time of the graph-based re-ranking. As mentioned in Section 2.4.3, there are two stages in graph-based re-ranking: graph construction and random walk. In the first stage, the DTW distances between all the hypothesized regions are computed. Since the acoustic feature sequences for hypothesized regions were usually short, the computation of the DTW distance for a region pair took less than one millisecond on a regular Linux machine.[15] Moreover, in real implementation, the computation of the distances can be parallel. In the second stage, power method took less than a second on a regular machine.

Fig. 5 is the detection error trade-off (DET) curves of the first-pass retrieval results (baselines), PRF and graph-based re-ranking for SI, SA and SD models (all results have been shown in Table 4 in terms of MAP). The red and blue curves are respectively for the first-pass retrieval results and PRF, and the green curves are for graph-based re-ranking

[13] In the PageRank scenario, this is desired because a web page that many other pages link to is deemed to be a famous page.
[14] We tuned the parameters y and z in this way to emphasize the power of graph-based re-ranking.
[15] With 2.66 GHz Intel processor.

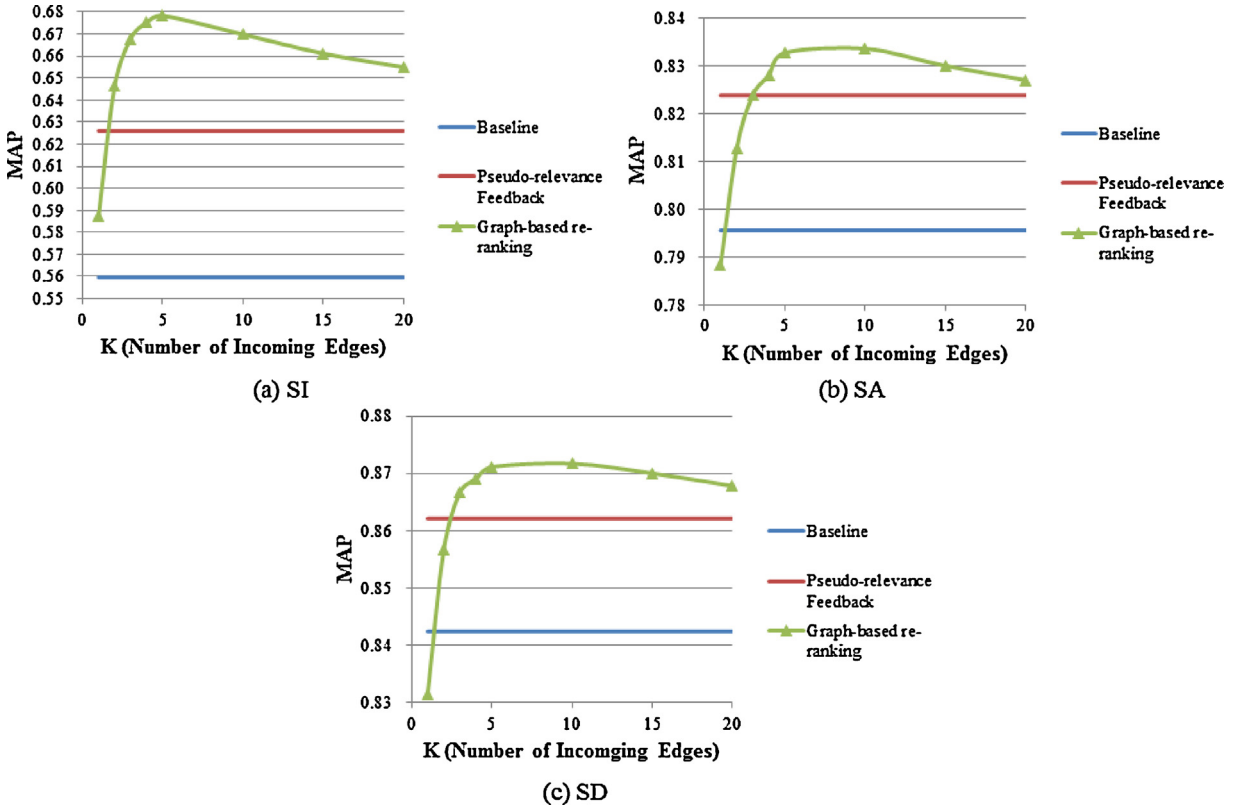

Fig. 4. MAP performance of the first-pass retrieval results (baselines), PRF and graph-based re-ranking with *Fixed Number of Incoming Edges* (IN). The horizontal scales in the figures are the numbers of incoming edges K. (a), (b) and (c) are respectively for SI, SA and SD models.

with *Fixed Numbers of Incoming Edges* (IN) and $K = 10$. The horizontal and vertical scales are respectively for false alarm rate and missed detection rate in log scale between 0 and 50%. Each point on the curve represents an operation point, or a specific threshold. The spoken segments with relevance scores higher than the threshold would be taken as relevant otherwise irrelevant. Thus each threshold corresponds to a set of false alarm and missed detection rates, or a point on the curves.

From Fig. 5(a), we found that for SI models PRF and graph-based re-ranking outperformed the baselines regardless of the operation points in terms of missed detection and false alarm rates. However, in Fig. 5(b) and (c), for SA and SD models when operating at higher threshold with higher missed detection rates (or only a few top-ranked segments were regarded as relevant), the baselines had lower false alarm rates than PRF and graph-based re-ranking. In other words, PRF and graph-based re-ranking actually increased the false alarm rates for the top-ranked segments. Because SA and SD models were of high quality, based on these models, the top-ranked segments given large original relevance scores from lattices were already ranked well, so PRF and graph-based re-ranking only slightly disturbed the perfect ranking under such conditions. On the contrary, in Fig. 5(b) and (c), when operated at low missed detection rates (or taking more segments as relevant), PRF and graph-based re-ranking offered much lower false alarm rates than the baselines. This shows that the segments besides the top-ranked ones were ranked better after PRF or graph-based re-ranking. Therefore, nevertheless the acoustic models were already of high quality, PRF and graph-based re-ranking still helped discriminate the segments whose relevance was not confidence enough based on the recognition results.

Compare the DET curves of PRF and graph-based re-ranking in Fig. 5. We found that when operating at low missed detection rates, the results of PRF were closer to the baselines than graph-based re-ranking for all the models, SI, SA and SD. Because PRF took the top-ranked segments as pseudo-relevant, in general the ranking of the top-ranked segments did not modify dramatically after PRF. In Fig. 5(a), although the results of PRF and graph-based re-ranking were close at the operation points of low missed detection rates (the right hand side of the figure), at the operation points of high missed detection rates (the upper left corner of the figure), graph-based re-ranking had lower false alarm

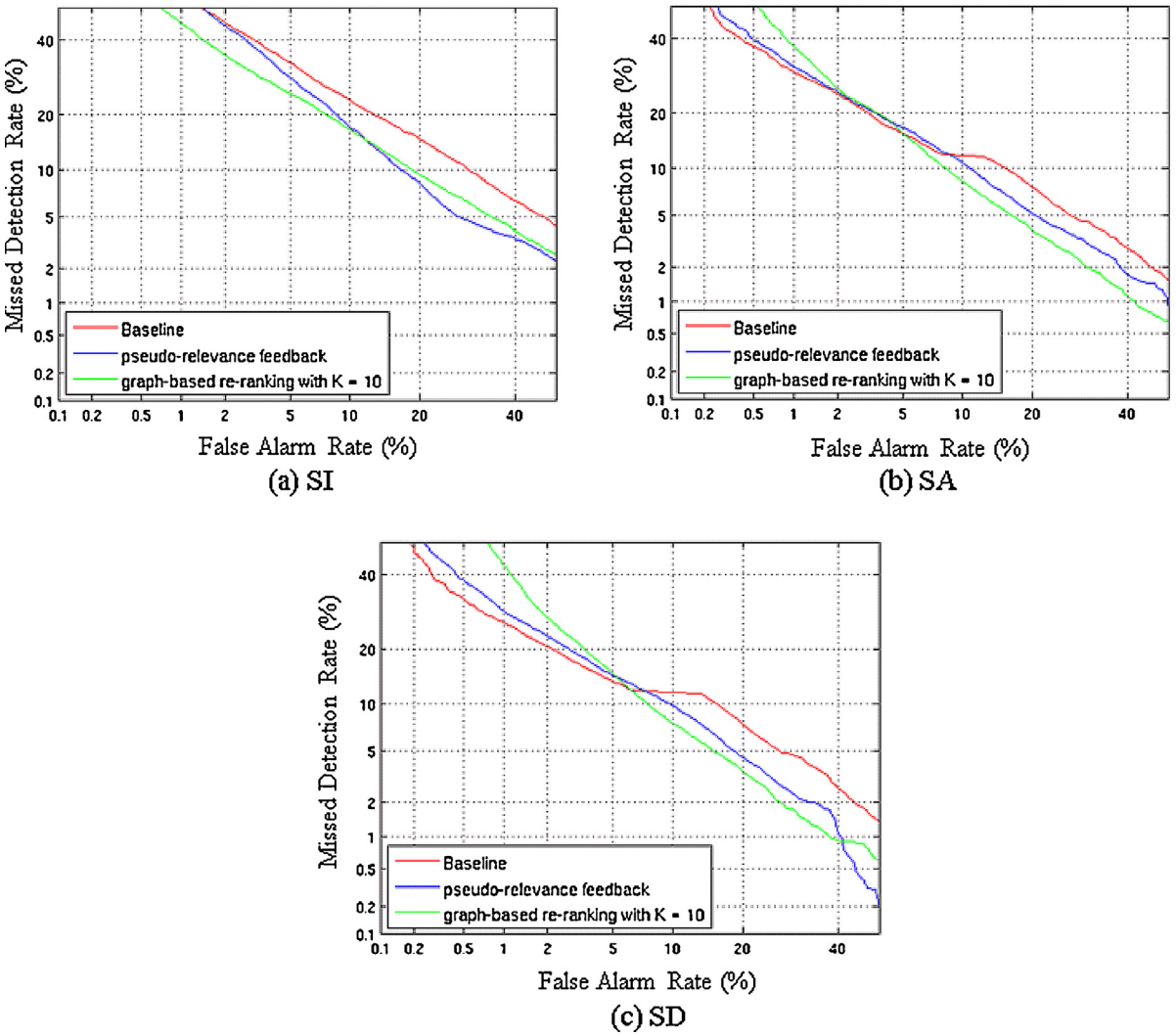

Fig. 5. Detection error trade-off (DET) curves of the first-pass retrieval results (baselines), PRF and graph-based re-ranking (with *Fixed Numbers of Incoming Edges* (IN) and $K = 10$). (a), (b) and (c) are respectively for SI, SA and SD models.

rates than PRF. This reveals that due to the low quality of the SI models the top-ranked segments with high relevance scores from the lattices still left some rooms for improvement, but PRF which taking the top-ranked segments as pseudo-relevant missed this opportunities. This explains why graph-based re-ranking yielded greatest improvements over PRF for SI models in Table 4.

In this section, on a set of IV queries, we found that fixed the number of incoming edges for each node is the best graph construction approach, and the comparison of PRF and graph-based re-ranking found that graph-based re-ranking outperformed PRF regardless of whether the sizes of pseudo-relevant/-irrelevant sets were tuned to be optimal.

4.3. Subword units for IV queries

Table 5 shows the results for IV queries with lattices of word and subword units. Parts (a), (b), and (c) are respectively with word-, character- or syllable-based lattices, and columns SI, SA, and SD correspond to the different sets of acoustic models. For each case we report the results of the first pass (baseline), PRF, and graph-based re-ranking (Graph). The superscripts * and † respectively indicate significantly better than the baselines and PRF. The numbers of pseudo-relevant (y) and irrelevant (z) segments for PRF were tuned on the testing queries. Compared the first pass retrieval

Table 5
MAP results of first pass (baseline), PRF, and graph-based re-ranking (Graph) for IV queries under different acoustic models with word-, character-, or syllable-based lattices.

	Approach	SI	SA	SD
(a) Word	Baseline	0.5596	0.7956	0.8424
	PRF	0.6261*	0.8239*	0.8621*
	Graph	0.6783*,†	0.8328*	0.8711*,†
(b) Character	Baseline	0.4733	0.7216	0.7507
	PRF	0.5761*	0.8209*	0.8595*
	Graph	0.6462*,†	0.8349*	0.8666*
(c) Syllable	Baseline	0.4329	0.6737	0.6941
	PRF	0.5281*	0.7797*	0.8182*
	Graph	0.5739*	0.8014*	0.8308*

* Significantly better than the baselines.
† Significantly better than the PRF.

results of the three units. It is clear that PRF always yielded improvements over the baseline, and the graph-based re-ranking always offered still further improvements regardless of the acoustic model set or the units selected. Note also that even though the word-based first-pass results were much better than the subword-based results, PRF and graph-based re-ranking yielded larger improvements for subword lattices. Because both PRF and graph-based re-ranking only re-rank the first-pass retrieved results, segments that were not retrieved in the first pass could never be retrieved. Since subword units offered higher recall than words in the lattices, the proposed approach yielded greater improvements for subword units.

Since different units contain complementary information, the integration of the results based on different units after graph-based re-ranking may outperform each individual. Fig. 6 shows the integration of the results from word-, character-, syllable-based lattices. The four curves in Fig. 6 are the MAP performances of graph-based re-ranking for graphs constructed with *Fixed Number of Incoming Edges* (IN) from word-, character-, syllable-based lattices and the integration of the three. Fig. 6(a), (b), and (c) are respectively for SI, SA, and SD models. The horizontal scales in the figures are the numbers of incoming edges K for the graphs. In Fig. 6, the results for the integration were always significantly better than the individuals regardless of the acoustic model set or the number of incoming edges K with only one exception in Fig. 6(b), that is, for SA models when $K=8$ the improvement of the integration over the character was not significant.

For all the recognition conditions, SI, SA, and SD, the integration in Fig. 6 was achieved by a weighted sum of the relevance scores obtained from word-, character-, and syllable-based lattices with weights 1.0, 0.2 and 0.04[16] respectively. These weights were set based on the following reasons. Because in Mandarin Chinese a word consists of one to several subwords, in Sections 2.1 and 2.2, a query's word sequence representation $Q_w = \{w_j, j = 1, 2, \ldots, N\}$ would be much shorter than the subword version $Q_s = \{s_j, j = 1, 2, \ldots, M\}$, or $N < M$. Because a_n and a_n' respectively in $R(x_i, Q_w)$ in (3) and $R(x_i, Q_s)$ in (6) raised as n was increased, $R(x_i, Q_w)$ is inherently smaller than $R(x_i, Q_s)$ due to $N < M$. Therefore, to let the results from word and subword-based lattices have comparable influence for the integration, it is reasonable to give the results from word lattices larger weights than subword lattices. On the other hand, each character is produced as a monosyllable in Mandarin Chinese, so very often many different characters with very different meanings may correspond to the same syllable (much less number of distinct syllables than distinct characters). Since in Mandarin Chinese different terms may have the same syllable sequence, and syllable is not as discriminative as character, syllable was given smaller weights than character during the integration. To achieve better integration results, the weights of different units can be learned by the learning-to-rank techniques from a set of training queries as in the previous studies (Meng et al., 2009), but this is out of the scope here because we only aim at showing that after graph-based re-ranking the integration of different units can yield further improvements.

[16] That is, $(0.2)^2$.

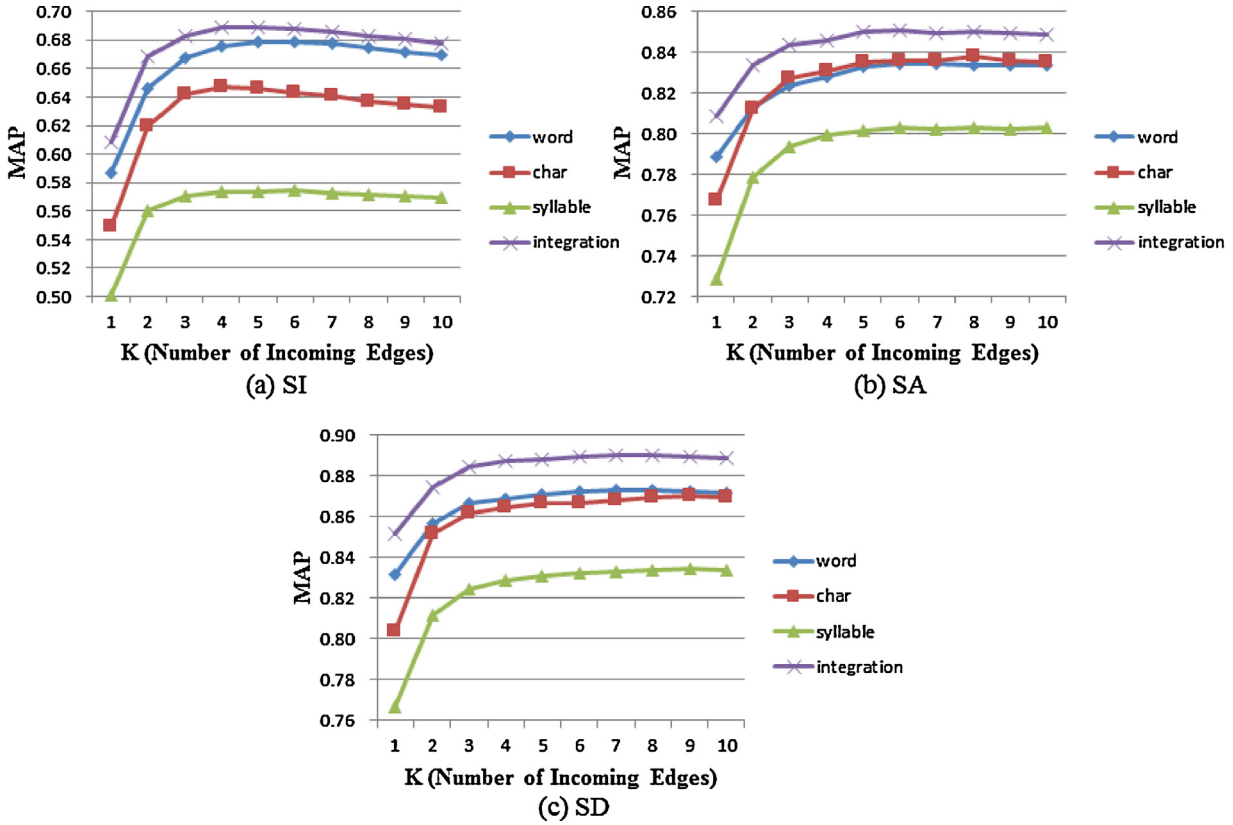

Fig. 6. MAP performance of the graph-based re-ranking approaches for graphs constructed from word-, character-, and syllable-based lattices with *Fixed Number of Incoming Edges* (IN), and the integration of the scores for the three different units. The horizontal scales in the figures are the numbers of incoming edges *K*. (a), (b) and (c) are respectively for SI, SA and SD models.

In this subsection, we show that PRF and graph-based re-ranking not only improved word-based retrieval but also character- and syllable-based retrieval, and the integration of the results based on word and different subword units yielded further improvement.

4.4. Experimental results for OOV queries

Table 6 shows the results for the OOV query experiments as summarized in Section 3.2 on lattices for words plus syllables (Hybrid) or for syllables only (Syllable), with PRF or the graph-based re-ranking applied. Section (a) (canonical) is the case assuming the canonical pronunciation of each OOV query was known, while Section (b) (g2p) shows the results based on the pronunciation estimated using the grapheme-to-phoneme approach, each including Column (1) (Hybrid) for lattices composed of Chinese and English word arcs plus English syllable arcs, and Column (2) (Syllable) for lattices containing Chinese and English syllable arcs only. In PRF approach, the numbers of pseudo-relevant (y) and -irrelevant (z) segments were determined by 4-fold cross validation. That is, the testing queries were first separated into 4 parts. In each trial, one part was selected as the development query set for parameter tuning, while the other three parts tested, and this was repeated 4 times.

We can see that the grapheme-to-phoneme-based pronunciations yielded reasonable performance, although naturally lower compared with the canonical pronunciation (Sections (b) vs (a)). Also, the lattices composed of syllables only outperformed the hybrid lattices (Columns (2) vs (1)). Because some of the English queries were incorrectly recognized as words with similar pronunciations in the lexicon in the hybrid case, transforming those words into corresponding syllable sequences increased the recall rates and thus improved the results.

It is clear that remarkable improvements were achieved by both PRF and graph-based re-ranking in all cases. However, we also observed that the graph-based re-ranking did not outperform PRF on the OOV query set. This is

Table 6
MAP results for PRF and graph-based re-ranking (with different K) for OOV queries on lattices for words plus syllables (Hybrid) or syllables only (Syllable) assuming canonical pronunciations (canonical), or pronunciations estimated with grapheme-to-phoneme (g2p).

		(a) Canonical		(b) g2p	
		(1) Hybrid	(2) Syllable	(1) Hybrid	(2) Syllable
Baseline		0.3611	0.3806	0.3092	0.3288
PRF		0.4699*	0.4967*	0.4127*	0.4362*
Graph	$K=1$	0.4246*	0.4423*	0.3621*	0.3888*
	$K=2$	0.4504*	0.4659*	0.3874*	0.4177*
	$K=3$	0.4613*	0.4697*	0.4020*	0.4232*
	$K=4$	0.4666*	0.4757*	0.4087*	0.4287*
	$K=5$	0.4654*	0.4760*	0.4087*	0.4293*
	$K=10$	0.4644*	0.4775*	0.4114*	0.4320*
	$K=15$	0.4684*	0.4794*	0.4175*	0.4340*
	$K=20$	0.4742*	0.4823*	0.4215*	0.4349*
	$K=100$	0.4766*	0.4840*	0.4213*	0.4328*

* Significantly better than the baselines.

probably because of the relatively poor recognition results for the OOV terms, or the relevance scores $R(x_i, Q_s)$ in (6) could be unreliable. Since the graph-based re-ranking in (16) was directly applied on these relevance scores, the random walk may be relatively sensitive to the noisy relevance scores for the individual segments from the first pass. On the other hand, PRF considered the pseudo-relevant and -irrelevant groups of segments as a whole which were obtained based on the ranking of the first pass. As a result, the disturbances of the individual scores did not necessarily change the pseudo-relevant and -irrelevant groups.

In this section, we found that both PRF and graph-based re-ranking can improve the performance of the OOV queries, but different from the results in IV queries, PRF was comparable to graph-based re-ranking here.

5. Conclusion

In this paper, we tested approaches that take into account acoustic feature similarity including PRF and graph-based re-ranking, and extended them to the retrieval based on subword-based lattices and OOV queries. Different graph construction approaches were investigated, and we found that fixed the number of incoming edges for each node in the graphs was the best approach. For the results of IV queries under different recognition conditions, both PRF and graph-based re-ranking yielded remarkable improvements for the results obtained from word-, character- and syllable-based lattices, and the integration of the results from different units after graph-based re-ranking gave further improvements over each individual. In addition, graph-based re-ranking outperformed PRF in most cases for the IV queries tested here. Finally, PRF and graph-based re-ranking were also applied on an open-vocabulary spoken content retrieval system based on a hybrid language model of words and syllables, and it was found that both approaches offered significant improvements for a set of OOV queries.

References

Akbacak, M., 2009. Robust Spoken Document Retrieval in Multilingual and Noisy Acoustic Environments. University of Colorado (Ph.D. thesis).
Akbacak, M., Vergyri, D., Stolcke, A., 2008. Open-vocabulary spoken term detection using graphone-based hybrid recognition systems. In: ICASSP.
Akiba, T., Nishizaki, H., Aikawa, K., Kawahara, T., Matsui, T., 2011. Overview of the IR for spoken documents task in NTCIR-9 workshop. In: Proceedings of NTCIR-9 Workshop.
Alberti, C., Bacchiani, M., Bezman, A., Chelba, C., Drofa, A., Liao, H., Moreno, P., Power, T., Sahuguet, A., Shugrina, M., Siohan, O., 2009. An audio indexing system for election video material. In: ICASSP.
Allauzen, C., Mohri, M., Saraclar, M., 2004. General indexation of weighted automata: application to spoken utterance retrieval. In: Proceedings of the Workshop on Interdisciplinary Approaches to Speech Indexing and Retrieval at HLT-NAACL 2004.
Aradilla, G., Vepa, J., Bourlard, H., 2006. Using posterior-based features in template matching for speech recognition. In: ICSLP.
Bisani, M., Ney, H., 2008. Joint-sequence models for grapheme-to-phoneme conversion. Speech Communication 50, 434–451.

Brin, S., Page, L., 1998. The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems 30 (1-7), 107–117.
Cao, G., Nie, J.-Y., Gao, J., Robertson, S., 2008. Selecting good expansion terms for pseudo-relevance feedback. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.
Chelba, C., Acero, A., 2005. Position specific posterior lattices for indexing speech. In: ACL.
Chelba, C., Silva, J., Acero, A., 2007. Soft indexing of speech content for search in spoken documents. Computer Speech & Language 21, 458–478.
Chelba, C., Hazen, T., Saraclar, M., 2008. Retrieval and browsing of spoken content. IEEE Signal Processing Magazine 25, 39–49.
Chen, C.-P., 2011. Improved Speech Information Retrieval by Acoustic Feature Similarity. National Taiwan University (Master's thesis).
Chen, C.-P., Lee, H.-Y., Yeh, C.-F., Lee, L.-S., 2010. Improved spoken term detection by feature space pseudo-relevance feedback. In: Interspeech.
Chen, Y.-N., Chen, C.-P., Lee, H.-Y., Chan, C.-A., Lee, L.-S., 2011. Improved spoken term detection with graph-based re-ranking in feature space. In: ICASSP.
Chan, C.-A., Lee, L.-S., 2011. Unsupervised hidden Markov modeling of spoken queries for spoken term detection without speech recognition. In: Interspeech.
Garcia, A., Gish, H., 2006. Keyword spotting of arbitrary words using minimal speech resources. In: ICASSP.
Garofolo, J.S., Auzanne, C.G.P., Voorhees, E.M., 2000. The TREC spoken document retrieval track: a success story. In: Text Retrieval Conference (TREC), vol. 8.
Glass, J., Hazen, T., Cyphers, S., Malioutov, I., Huynh, D., Barzilay, R., 2007. Recent progress in the MIT spoken lecture processing project. In: Interspeech.
Goto, M., Ogata, J., Eto, K., 2007. Podcastle: a web 2.0 approach to speech recognition research. In: Interspeech.
Hansen, J.H., Huang, R., Mangalath, P., Zhou, B., Seadle, M., Deller, J.R., 2004. SPEECHFIND: Spoken Document Retrieval for a National Gallery of the Spoken Word.
Hazen, T.J., Shen, W., White, C., 2009. Query-by-example spoken term detection using phonetic posteriorgram templates. In: ASRU.
Hori, T., Hetherington, I., Hazen, T., Glass, J., 2007. Open vocabulary spoken utterance retrieval using confusion networks. In: ICASSP.
Hsu, W.H., Kennedy, L.S., Chang, S.-F., 2007. Video search reranking through random walk over document-level context graph. In: Proceedings of the 15th International Conference on Multimedia, pp. 971–980.
Itoh, Y., Iwata, K., Kojima, K., Ishigame, M., Tanaka, K., Wook Lee, S., 2007. An integration method of retrieval results using plural subword models for vocabulary-free spoken document retrieval. In: Interspeech.
Jansen, A., Durme, B.V., 2012. Indexing raw acoustic features for scalable zero resource search. In: Interspeech.
Kamvar, S.D., Haveliwala, T.H., Manning, C.D., Golub, G.H., 2003. Extrapolation methods for accelerating pagerank computations. In: Proceedings of the 12th International Conference on World Wide Web, WWW'03, pp. 261–270.
Kong, S.-Y., Wu, M.-R., Lin, C.-K., Fu, Y.-S., Lee, L.-S., 2009. Learning on demand – course lecture distillation by information extraction. In: ICASSP.
Kurland, O., Lee, L., Domshlak, C., 2005. Better than the real thing? Iterative pseudo-query processing using cluster-based language models. In: Proceedings of the 28th Annual international ACM SIGIR Conference on Research and Development in Information Retrieval.
Langville, A.N., Meyer, C.D., 2005. A survey of eigenvector methods for web information retrieval. SIAM Review 47, 135–161.
Lee, L.-S., Chen, B., 2005. Spoken document understanding and organization. IEEE Signal Processing Magazine 22, 42–60.
Lee, H.-Y., Lee, L.-S., 2013. Enhanced spoken term detection using support vector machines and weighted pseudo examples. IEEE Transactions on Audio, Speech, and Language Processing 21 (6), 1272–1284.
Lee, H.-Y., Tang, Y.-L., Tang, H., Lee, L.-S., 2009. Spoken term detection from bilingual spontaneous speech using code-switched lattice-based structures for words and subword units. In: ASRU.
Lee, H.-Y., Chen, Y.-N., Lee, L.-S., 2011. Improved speech summarization and spoken term detection with graphical analysis of utterance similarities. In: APSIPA.
Lee, H.-Y., Wen, T.-H., Lee, L.-S., 2012a. Improved semantic retrieval of spoken content by language models enhanced with acoustic similarity graph. In: SLT.
Lee, H.-Y., Chou, P.-W., Lee, L.-S., 2012b. Open-vocabulary retrieval of spoken content with shorter/longer queries considering word/subword-based acoustic feature similarity. In: Interspeech.
Logan, B., Moreno, P., van Thong, J.-M., Whittaker, E., 2000. An experimental study of an audio indexing system for the web. In: ICSLP.
Logan, B., Van Thong, J.-M., Moreno, P., 2005. Approaches to reduce the effects of OOV queries on indexed spoken audio. IEEE Transactions on Multimedia 7, 899–906.
Lv, Y., Zhai, C., 2009. A comparative study of methods for estimating query language models with pseudo feedback. In: Proceeding of the 18th ACM Conference on Information and Knowledge Management.
Lv, Y., Zhai, C., 2010. Positional relevance model for pseudo-relevance feedback. In: Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval.
Mangu, L., Brill, E., Stolcke, A., 2000. Finding consensus in speech recognition: word error minimization and other applications of confusion networks. Computer Speech & Language 14 (4), 373–400.
Manaskasemsak, B., Rungsawang, A., 2005. An efficient partition-based parallel pagerank algorithm. In: Proceedings of the 11th International Conference on Parallel and Distributed Systems, 2005, vol. 1, pp. 257–263.
Meng, C.-H., Lee, H.-Y., Lee, L.-S., 2009. Improved lattice-based spoken document retrieval by directly learning from the evaluation measures. In: ICASSP.
Meye, C.D., 2000. Matrix Analysis and Applied Linear Algebra. SIAM, pp. 661–706 (Chapter 8).

Montgomery, J., Si, L., Callan, J., Evans, D.A., 2004. Effect of varying number of documents in blind feedback: analysis of the 2003 NRRC RIA workshop "bf_numdocs" experiment suite. In: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'04, pp. 476–477.

Ng, K., 2000. Subword-based Approaches for Spoken Document Retrieval. Massachusetts Institute of Technology (Ph.D. thesis).

Oard, D.W., Soergel, D., Doermann, D., Huang, X., Murray, G.C., Wang, J., Ramabhadran, B., Franz, M., Gustman, S., Mayfield, J., Kharevych, L., Strassel, S., 2004. Building an information retrieval test collection for spontaneous conversational speech. In: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'04, pp. 41–48.

Otterbacher, J., Erkan, G., Radev, D.R., 2009. Biased LexRank: passage retrieval using random walks with question-based priors. Information Processing & Management 45 (1), 42–54.

Pan, H.-L.C.Y.-C., Lee, L.-S., 2007. Analytical comparison between position specific posterior lattices and confusion networks based on words and subword units for spoken document indexing. In: ASRU.

Pan, Y.-C., Chang, H.-L., Lee, L.-S., 2007. Subword-based position specific posterior lattices (S-PSPL) for indexing speech information. In: Interspeech.

Pan, Y.-C., Lee, H.-Y., Lee, L.-S., 2012. Interactive spoken document retrieval with suggested key terms ranked by a Markov decision process. IEEE Transactions on Audio, Speech and Language Processing 20 (2), 632–645.

Parada, C., Sethy, A., Ramabhadran, B., 2009. Query-by-example spoken term detection for OOV terms. In: ASRU.

Rastrow, A., Sethy, A., Ramabhadran, B., Jelinek, F., 2009. Towards using hybrid word and fragment units for vocabulary independent LVCSR systems. In: Interspeech.

Saraclar, M., 2004. Lattice-based search for spoken utterance retrieval. In: Proceedings of HLT-NAACL 2004, pp. 129–136.

Szoke, I., Fapso, M., Burget, L., Cernocky, J., 2008. Hybrid word–subword decoding for spoken term detection. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.

Tao, T., Zhai, C., 2006. Regularized estimation of mixture models for robust pseudo-relevance feedback. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.

Tian, X., Yang, L., Wang, J., Yang, Y., Wu, X., Hua, X.-S., 2008. Bayesian video search reranking. In: Proceedings of the 16th ACM International Conference on Multimedia, pp. 131–140.

Tu, T.-W., Lee, H.-Y., Lee, L.-S., 2011. Improved spoken term detection using support vector machines with acoustic and context features from pseudo-relevance feedback. In: ASRU.

Turunen, V.T., 2008. Reducing the effect of OOV query words by using morph-based spoken document retrieval. In: Interspeech.

Turunen, V.T., Kurimo, M., 2007. Indexing confusion networks for morph-based spoken document retrieval. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'07, pp. 631–638.

Wallace, R., Vogt, R., Sridharan, S., 2007. A phonetic search approach to the 2006 NIST spoken term detection evaluation. In: Interspeech.

Wang, D., Frankel, J., Tejedor, J., King, S., 2008. A comparison of phone and grapheme-based spoken term detection. In: ICASSP.

Wang, H., Leung, C.-C., Lee, T., Ma, B., Li, H., 2012. An acoustic segment modeling approach to query-by-example spoken term detection. In: ICASSP.

Yeh, C.-F., Sun, L.-C., Huang, C.-Y., Lee, L.-S., 2011. Bilingual acoustic modeling with state mapping and three-stage adaptation for transcribing unbalanced code-mixed lectures. In: ICASSP.

Zhang, Y., Glass, J., 2009. Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams. In: ASRU.

Zhang, Y., Glass, J., 2010. Towards multi-speaker unsupervised speech pattern discovery. In: ICASSP.

Zhou, B., 2003. Audio Parsing and Rapid Speaker Adaptation in Speech Recognition for Spoken Document Retrieval. University of Colorado (Ph.D. thesis).