IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 22, 2019, accepted May 12, 2019, date of publication May 23, 2019, date of current version June 5, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2918638

Query-by-Example Speech Search Using Recurrent Neural Acoustic Word Embeddings With Temporal Context

YOUGEN YUAN[1], (Student Member, IEEE), CHEUNG-CHI LEUNG[2], (Member, IEEE), LEI XIE[1], (Senior Member, IEEE), HONGJIE CHEN[2], (Member, IEEE), AND BIN MA[2], (Senior Member, IEEE)

[1]School of Computer Science, Northwestern Polytechnical University, Xi'an 710129, China
[2]Machine Intelligence Technology, Alibaba Group

Corresponding author: Lei Xie (lxie@nwpu-aslp.org)

This work was supported in part by the National Natural Science Foundation of China, under Grant 61571363.

ABSTRACT Acoustic word embeddings (AWEs) have been popular in low-resource query-by-example speech search. They are using vector distances to find the spoken query in search content, which has much lower computation than the conventional dynamic time warping (DTW)-based approaches. The AWE networks are usually trained using variable-length isolated spoken words, while they are applied to fixed-length speech segments obtained by shifting an analysis window on speech content. There is an obvious mismatch between the learning of AWEs and its application on search content. To mitigate such mismatch, we propose to include temporal context information on spoken word pairs to learn recurrent neural AWEs. More specifically, the spoken word pairs are represented by multi-lingual bottleneck features (BNFs) and padded with the neighboring frames of the target spoken words to form fixed-length speech segment pairs. A deep bidirectional long short-term memory (BLSTM) network is then trained with a triplet loss using the speech segment pairs. Recurrent neural AWEs are obtained by concatenating the BLSTM backward and forward outputs. During QbE speech search stage, both spoken query and search content are converted into recurrent neural AWEs. Cosine distances are then measured between them to find the spoken query. The experiments show that using temporal context is essential to alleviate the mismatch. The proposed recurrent neural AWEs trained with temporal context outperform the previous state-of-art features with 12.5% relative mean average precision (MAP) improvement on QbE speech search.

INDEX TERMS Acoustic word embeddings, temporal context, bidirectional long short-term memory network, spoken word pairs, query-by-example spoken term detection.

I. INTRODUCTION

Query-by-Example (QbE) speech search or spoken term detection (STD) is the task of searching for the occurrence of a spoken query in a collection of audio archives [1], [2]. This task has received much attention as it involves matching spoken queries directly on speech, without the need of a large vocabulary continuous speech recognition (LVCSR) system. The spoken query is an audio example of the keyword of interest. Besides, the system building does not necessarily need language-specific knowledge, such as phoneme definition and pronunciation lexicon. Hence the task is particularly promising for *low-resource* speech search scenarios. In such scenarios, we do not have a sizable amount of labeled data to build a decent speech recognizer. A series of related benchmark evaluations, such as spoken web search (SWS) [3]–[5] and QbE search on speech task (QUESST) [6], [7], have recently focused on this low-resource task.

In low-resource settings, a typical approach for QbE speech search is to learn efficient frame-level feature representations and perform dynamic time warping (DTW) to find the matching spoken query [8], [9]. However, DTW is inefficient to do the search on massive audio archives. As an alternative to DTW, *acoustic word embeddings (AWEs)* are

The associate editor coordinating the review of this manuscript and approving it for publication was Moayad Aloqaily.

becoming more and more popular. They aim to map variable-length speech segments in a fixed-dimensional vector space where the word discrimination power of these speech segments is preserved as much as possible [10], [11]. For the QbE speech search task, a neural network is usually trained with a set of spoken word examples to embed both the spoken query and search content into the same space [12], [13]. With AWEs, a simple vector distance (e. g., cosine distance) can be measured between the spoken query and search content with much lower computation than DTW.

One of the keys to such an AWE-based QbE approach is how to achieve effective embeddings. Studies have shown that learning AWEs with deep neural architectures has been successful in isolated word discrimination [10], [11]. With the recent advance in acoustic modeling, recurrent neural networks (RNNs) have been proven to be more capable of capturing temporal dependency with variable-length sequential speech data in a fixed-dimensional space, leading to good performances in downstream speech applications [14]–[17].

Another key factor to the success of AWE-based QbE approach is to find an appropriate way to embed both the spoken query and search content. In previous study, AWEs are usually learned using isolated spoken words [10], [11], [14], [18]. As the word boundaries are not readily available in QbE speech search, a sliding window is usually applied to search content to get a sequence of fixed-dimensional speech segments. Without a speech recognizer or a segmentation technology, it is hard to segment search content into isolated words or other meaningful units to generate AWEs. These fixed-dimensional speech segments, without clear boundaries, unavoidably contain a partial word and more words. Hence there is an obvious mismatch between the learning of AWEs and its application on search content, which affects the search quality.

To mitigate the mismatch, in this paper, we propose to include the neighboring frames of each target spoken words as *temporal context* to learn recurrent neural AWEs. With the temporal context, we learn the important neighboring information around the target words in recurrent neural AWEs. Our approach only requires a limited amount of spoken word pairs (pairs of different realizations of the same spoken word) as weak supervision. These spoken word pairs are much easier to access in low-resource settings. Feeding the AWE network with paired examples from the same and different context-padded spoken words, we enable the network with desired discriminative ability.

In our AWE-based QbE approach, the spoken word pairs are firstly represented by multi-lingual bottleneck features (BNFs) as they capture rich information of phonetic discrimination that is even useful for an unseen language. Then, these spoken word pairs are padded with the same length of temporal context on both sides to form fixed-length speech segment pairs. A deep bidirectional long short-term memory (BLSTM) network is trained with a triplet loss using the fixed-length speech segment pairs. Our proposed recurrent neural AWEs are learned by concatenating the BLSTM backward and forward outputs. During QbE speech search stage, both spoken query and search content are converted into recurrent neural AWEs, and then cosine distances are measured between them to find the spoken query.

Experiments show that using temporal context is essential to alleviate the mismatch. As compared with the previous state-of-art features, the proposed recurrent neural AWEs achieve superior performance in terms of both search accuracy and search time. Our study also indicates that sufficient speech segment pairs with rich vocabulary coverage and discriminative input features are both important to learn recurrent neural AWEs for QbE speech search.

The rest of this paper is organized as follows. Section II reviews the prior works on QbE speech search and AWEs. Section III details our proposed method of learning recurrent neural AWEs with temporal context for QbE speech search. Section IV reports the experimental results and describes the significance of our findings. Section V concludes the paper and shows our future work.

II. RELATED WORKS

A. QUERY-BY-EXAMPLE SPEECH SEARCH

In low-resource settings, many approaches perform acoustic pattern matching with DTW on frame-level feature representations for QbE speech search. The feature representations can be learned in an unsupervised or supervised manner. In unsupervised manners, many studies investigated learning posteriorgrams from Gaussian mixture models (GMMs) [2], [19]–[21], deep Boltzmann machines (DBMs) [8] and acoustic segment models (ASMs) [22]–[25]. These generated posteriorgrams can discriminate phoneme patterns more accurately than spectral features including mel-frequency cepstral coefficients (MFCCs), perceptual linear prediction (PLP) and filter-bank (Fbank) features. In supervised manners, studies have shown that BNFs extracted from a bottleneck-shaped deep neural network (DNN) with phonetic targets outperformed the above spectral features by a large margin in phonetic discrimination [26], [27]. Thus some studies investigated cross- or multi-lingual BNFs for QbE speech search with promising results [9], [28], [29]. The BNFs are usually extracted from a supervised DNN which is trained using a sizable amount of labeled data from resource-rich non-target languages.

In practical low-resource scenarios, it is easy to access a limited amount of paired word examples, no matter whether they are obtained from annotation [11], [30] or unsupervised clustering [31]–[33]. Previous studies show that using limited paired examples with deep neural architectures can learn efficient feature representations [34]–[36]. Learning representations from paired examples is originally proposed in computer vision [34], and it has been adopted in natural language processing [35] and speech processing [36]. As for our low-resource QbE speech search scenario, the same spoken words are naturally set as paired examples. Learning frame-level feature representation with spoken word pairs has been successfully used in phonetic or word discrimination [11], [30], [31], [37] and QbE speech search [29] as well.

B. ACOUSTIC WORD EMBEDDINGS

Acoustic embeddings were originally proposed in [18], where Laplacian eigenmaps were used to learn variable-length speech segments in a fixed-dimensional space where the learned compact vectors have a reasonable discriminative ability. Later, acoustic embeddings have been successfully applied in lexical clustering [38], unsupervised ASR [39], [40] and QbE speech search [41]. When the embedding unit is a word or word-like unit, we refer to such acoustic embeddings as AWEs.

Recently, many neural networks have been intensively used to learn AWEs. For example, deep convolutional neural networks (CNNs) have been used to learn AWEs for isolated word discrimination [10], [11] and ASR lattice rescoring [42], [43]. These previous works have shown that convolutional neural AWEs are able to achieve superior performance over the frame-level representation based DTW approaches. To take advantages of important temporal dependency in speech, RNNs have been adopted for learning AWEs as well. Studies have demonstrated that RNNs are more flexible to deal with variable-length input sequences. They can discriminate isolated spoken words more accurately than DTW-based approaches with frame-level speech representations [14], [15], [44], [45].

There are three prior approaches that are most related to our study of QbE speech search. In [46], an LSTM network is used with an analysis window on search content to generate AWEs for finding the speech keyword. This approach requires a large amount of transcribed speech data from the target language for training the LSTM network. Hence it cannot be applied in low-resource settings. In [12], the authors proposed a siamese network to learn AWEs using limited isolated spoken words with word labels, and search content is spliced into a large number of overlapping speech segments to generate AWEs. However, these speech segments may contain a partial word, one or more isolated words, there is a mismatch between the learning of AWEs and its application on search content. To mitigate such mismatch, in our previous work [13], we introduced that including temporal context information to learn CNN-based AWEs was helpful, but the embeddings were not much better than frame-level feature representations. Hence it is still plenty of space to improve the performance of learning AWEs with temporal context for QbE speech search.

III. METHODS

A. TASK DESCRIPTION

In low-resource scenarios, although we usually do not have a sizable amount of labeled speech data, phoneme definition, and pronunciation lexicon, we are still able to get limited word-like pairs from annotation [13] or unsupervised term detection (UTD) [31]. As for our case, we only know whether the two speech segments belong to the same word-like unit or not. Based on the idea that different examples of the same word should have similar feature representations, we can use spoken word pairs to learn discriminative AWEs [10], [11]. However, as the word boundaries are not available in QbE speech search, shifting analysis window to generate speech segments unavoidably contain a partial word and more words. Hence there is a clear mismatch between the learning of AWEs and its application on search content. Temporal context padding is a feasible way to mitigate such kind of mismatch.

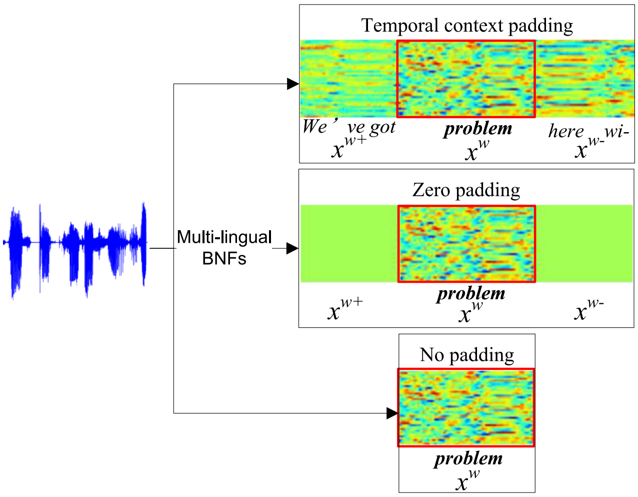


FIGURE 1. The diagram of temporal context padding. We also depict zero or no padding as a contrast.

B. TEMPORAL CONTEXT PADDING

Temporal context padding adopts the neighboring frames of each variable-length target word x^w as the temporal context to form a fixed-length speech segment. Following our previous study [13], multi-lingual BNFs are used to represent the speech segment, and the fixed length is set to the maximum duration of all target words in the training set. As shown in Figure 1, for each target word x^w, temporal context padding is to add its original previous information (denoted as x^{w+}) in front and its subsequent information (denoted as x^{w-}) behind with the same number of frames. Notice that the temporal context may contain a partial word (e. g., "wi-" in "with"), a whole word (e. g., "here"), or even multiple words (e. g., "We've got"). It is also used together with the target word x^w to train the neural networks. With the temporal context, we can reduce the mismatch between the learning of AWEs and its application on search content, leading to improved search performance.

As a contrast, zero padding is to directly pad zeros on both sides of each target word x^w to form a fixed-length input sequence, which has been previously used to learn AWEs with feed-forward neural networks for isolated word discrimination [10], [11]. In addition, when a BLSTM network is used for learning recurrent neural AWEs, each variable-length target word x^w can be directly taken as an

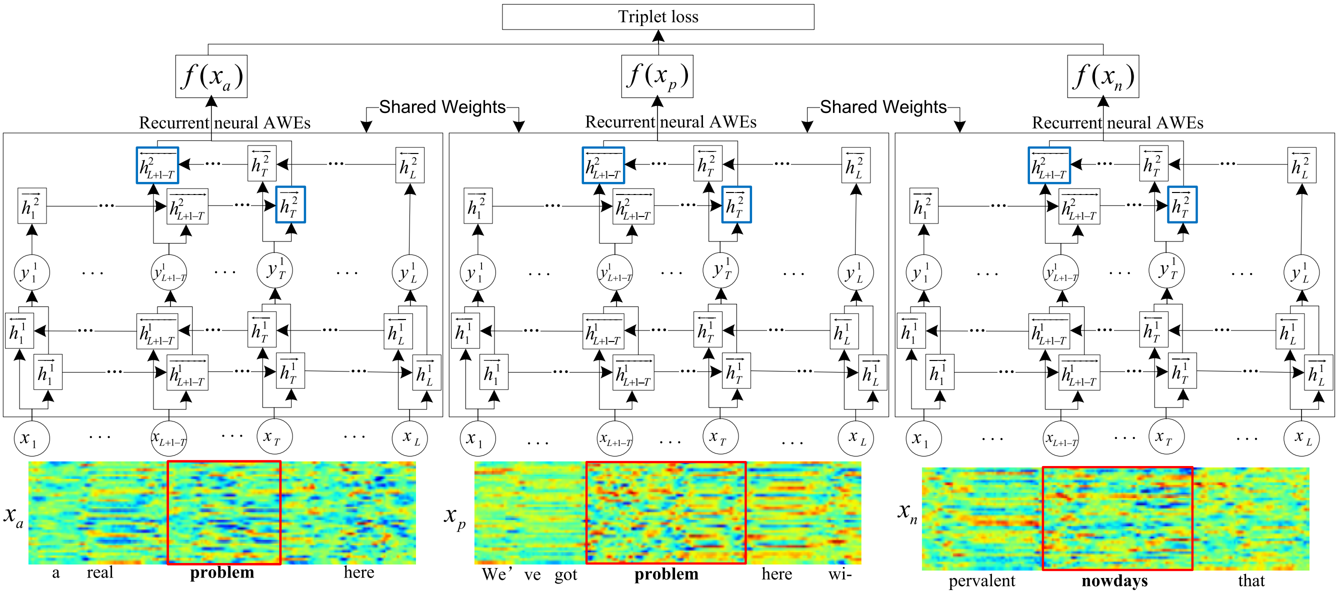


FIGURE 2. The diagram of learning recurrent neural AWEs with temporal context. Recurrent neural AWEs are learned by concatenating the BLSTM backward and forward outputs (the blue box) from the first and last frames of target spoken word (the red box) respectively in the speech segments.

input sequence. We refer to this approach as no padding. Notice that there is no temporal context in both zero padding and no padding. The three different padding methods are investigated for QbE speech search in the experiments.

C. RECURRENT NEURAL ACOUSTIC WORD EMBEDDINGS

Through temporal context padding, we aim to let the BLSTM network learn the previous and subsequent speech sequences (x^{w-}, x^{w+}) of the target word x^w and then alleviate the mismatch between the learning of AWEs and its application on search content. As shown in Figure 2, the input of the BLSTM network is a triplet of 3 examples (x_p, x_a, x_n). We use a pair of speech segments as an anchor example x_a and a positive example x_p, respectively. At the beginning of each training epoch, we randomly sample another speech segment that is different from the anchor example in the training dataset as a negative example x_n. In this way, the speech segment x_a and x_p contain the same target word (e. g., "problem" in the red box of Figure 2) in the middle, while the speech segment x_n contains a different target word (e. g., "nowadays").

The BLSTM network stacks multiple bidirectional LSTM layers, and each bidirectional LSTM layer consists of a forward layer and a backward layer. The forward and backward layers access different portions of input speech segments. The length of each portion T of input speech segments, also refered to as the effective input length, is calculated by:

$$T = \begin{cases} len(x^w) & \text{no padding} \\ (L - len(x^w))/2 + len(x^w) & \text{zero/context padding} \end{cases} \quad (1)$$

where $len(x^w)$ is the length of the target word x^w. L is the length of speech segment x. Here L is set as the maximum length of all target words in the training set. $(L - len(x^w))/2$ frames with zeros or temporal context are padded with zeros or temporal context on both sides to form each speech segment x with the fixed-length L.

The forward layers compute from the first vector of the fixed-length speech segment (x_1), and reach the last vector of the target word (x_T). Thus the forward hidden sequences $\overrightarrow{h}_t^n$ from layer $n = 1$ to $N - 1$ and time step $t = 1$ to T are calculated by:

$$\begin{aligned}
\overrightarrow{i}_t^n &= \sigma(\overrightarrow{W}_i^n \overrightarrow{y}_t^n + \overrightarrow{U}_i^n \overrightarrow{h}_{t-1}^n + \overrightarrow{b}_i^n) \\
\overrightarrow{f}_t^n &= \sigma(\overrightarrow{W}_f^n \overrightarrow{y}_t^n + \overrightarrow{U}_f^n \overrightarrow{h}_{t-1}^n + \overrightarrow{b}_f^n) \\
\overrightarrow{o}_t^n &= \sigma(\overrightarrow{W}_o^n \overrightarrow{y}_t^n + \overrightarrow{U}_o^n \overrightarrow{h}_{t-1}^n + \overrightarrow{b}_o^n) \\
\overrightarrow{c}_t^n &= \overrightarrow{f}_t^n \overrightarrow{c}_{t-1}^n + \overrightarrow{i}_t^n \, \sigma(\overrightarrow{W}_c^n \overrightarrow{y}_t^n + \overrightarrow{U}_c^n \overrightarrow{h}_{t-1}^n + \overrightarrow{b}_c^n) \\
\overrightarrow{h}_t^n &= \overrightarrow{o}_t^n \, \sigma(\overrightarrow{c}_t^n)
\end{aligned} \quad (2)$$

On the other hand, the backward layers compute from the last vector of the fixed-length speech segment (x_L), and reach the first vector of target word (x_{L+1-T}). The backward hidden sequence $\overleftarrow{h}_t^n$ from layer $n = 1$ to $N - 1$ and time step $t = L$ to $L + 1 - T$ are calculated by:

$$\begin{aligned}
\overleftarrow{i}_t^n &= \sigma(\overleftarrow{W}_i^n \overleftarrow{y}_t^n + \overleftarrow{U}_i^n \overleftarrow{h}_{t+1}^n + \overleftarrow{b}_i^n) \\
\overleftarrow{f}_t^n &= \sigma(\overleftarrow{W}_f^n \overleftarrow{y}_t^n + \overleftarrow{U}_f^n \overleftarrow{h}_{t+1}^n + \overleftarrow{b}_f^n) \\
\overleftarrow{o}_t^n &= \sigma(\overleftarrow{W}_o^n \overleftarrow{y}_t^n + \overleftarrow{U}_o^n \overleftarrow{h}_{t+1}^n + \overleftarrow{b}_o^n) \\
\overleftarrow{c}_t^n &= \overleftarrow{f}_t^n \overleftarrow{c}_{t+1}^n + \overleftarrow{i}_t^n \, \sigma(\overleftarrow{W}_c^n \overleftarrow{y}_t^n + \overleftarrow{U}_c^n \overleftarrow{h}_{t+1}^n + \overleftarrow{b}_c^n) \\
\overleftarrow{h}_t^n &= \overleftarrow{o}_t^n \, \sigma(\overleftarrow{c}_t^n)
\end{aligned} \quad (3)$$

where σ denotes the hyperbolic tangent function, and i, f, o and c are the *inputgate*, *forgetgate*, *outputgate* and *cell* vectors. All the W or U, and b denote the weights and bias of the BLSTM network respectively. y_t^n represents the BLSTM outputs at the layer of n and the time step of t, which is calculated by:

$$y_t^n = \begin{cases} x_t & n = 0 \\ [\overrightarrow{h}_t^n \overleftarrow{h}_t^n] & 1 \leq n \leq N-1 \end{cases} \quad (4)$$

where y^0 corresponds to input sequence x. [] represents the vector concatenation operation. Notice that the effective input length T is computed in each hidden layer by iterating the forward layer from $t = 1$ to T and the backward layer from $t = L$ to $L + 1 - T$, and the other hidden sequences are set to zeros for simple calculation.

In the last layer of BLSTM network, the forward and backward directions with the time step t from $t = 1$ to T are computed to obtain the forward hidden sequence $\overrightarrow{h}_T^N$ and backward hidden sequence $\overleftarrow{h}_{L+1-T}^N$ respectively. Then the hidden sequences from both directions are concatenated as our learned recurrent neural AWEs $f(x)$.

$$f(x) = [\overrightarrow{h}_T^N \overleftarrow{h}_{L+1-T}^N] \quad (5)$$

Using such concatenation of BLSTM outputs as the final recurrent neural AWEs $f(x)$ can retain important sequential context information of speech, leading to improved QbE speech search. We have used the BLSTM outputs with one or the last few time steps as in [46], but the resulting recurrent neural AWEs do not bring improvement in our preliminary test.

After forwarding the BLSTM network, a triplet loss [47] is employed to generate recurrent neural AWEs. The triplet loss is defined as

$$Loss(x_p, x_a, x_n) = max\{0, \delta + d^+ - d^-\} \quad (6)$$

$$d^+ = \frac{1 - \frac{f(x_p) * f(x_a)}{\|f(x_p)\|_2 \|f(x_a)\|_2}}{2} \quad (7)$$

$$d^- = \frac{1 - \frac{f(x_n) * f(x_a)}{\|f(x_n)\|_2 \|f(x_a)\|_2}}{2} \quad (8)$$

where δ is a margin constraint that regularizes the gap between the cosine distance of same speech content d^+ and the cosine distance of different speech content d^-. After training the deep BLSTM network, it can be used as an extractor to generate recurrent neural AWEs for QbE speech search.

D. EMBEDDINGS BASED QBE SPEECH SEARCH

Figure 3 illustrates the process of QbE speech search stage based on the recurrent neural AWEs. A fixed-length analysis window (the black box) is shifted on the search content y along the time axis. The size of the fixed-length analysis window is the same as the training speech segment pairs. The window shift size is set to 5 frames as it is the optimal choice of efficiency in our preliminary test.

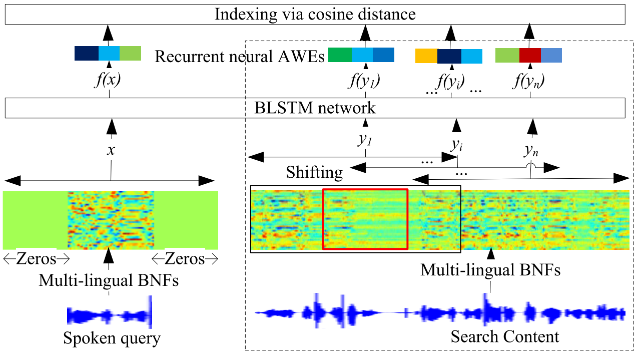


FIGURE 3. The process of QbE speech search based on recurrent neural AWEs.

For the input sequence in the analysis window, word boundaries (the red box) are not available to obtain the length of target word $len(y_i^w)$, so that the effective input length T cannot be calculated by Eq. 1. In our experiments, $len(y_i^w)$ is set as the average length of all target words in the training set (63 frames). Then the speech segment with the constant effective input length T in the analysis window is converted into recurrent neural AWEs via the trained deep BLSTM network. As a result, search content is represented by a sequence of recurrent neural AWEs as $(f(y_1), \ldots, f(y_i), \ldots, f(y_n))$. The whole process of generating recurrent neural AWEs on search content (the black dashed line) can be pre-calculated to save search run-time.

As no context information is available in the spoken query x, zeros are padded on both sides of x to form the same length as the analysis window. This operation is also to mitigate the mismatch between the learning of AWEs and its application for the spoken query x and it is an inevitable mismatch in our approach. The padded spoken query is then converted into recurrent neural AWEs $f(x)$ via the same deep BLSTM network. In this way, cosine distances over the recurrent neural AWEs can be measured between the spoken query and search content. A minimum cost is calculated by:

$$Cost(x, y) = min(1 - \frac{f(x) * f(y_i)}{\|f(x)\|_2 \|f(y_i)\|_2}), \quad i = 1, \ldots, n \quad (9)$$

Finally, given a spoken query x, all the minimum distance costs in search content are returned by the QbE speech search.

IV. EXPERIMENTS AND DISCUSSION

A. EXPERIMENTAL SETUP

In our experiments, English is considered as the low-resource target language. In such a low-resource scenario, only limited spoken word pairs are available to train the recurrent neural AWEs for QbE speech search. Following our previous study [13], [29], these spoken word pairs are from the English Switchboard telephone speech corpus (LDC97S62). We regarded Mandarin Chinese and Spanish as the resource-rich non-target languages for multi-lingual bottleneck feature extraction. The Chinese data is from the HKUST Mandarin Chinese telephone speech corpus (LDC2005S15) and the Spanish data is from the Fisher Spanish telephone speech corpus (LDC2010S01). With the Chinese and Spanish data, a multi-lingual BNF extractor is trained using a feed-forward

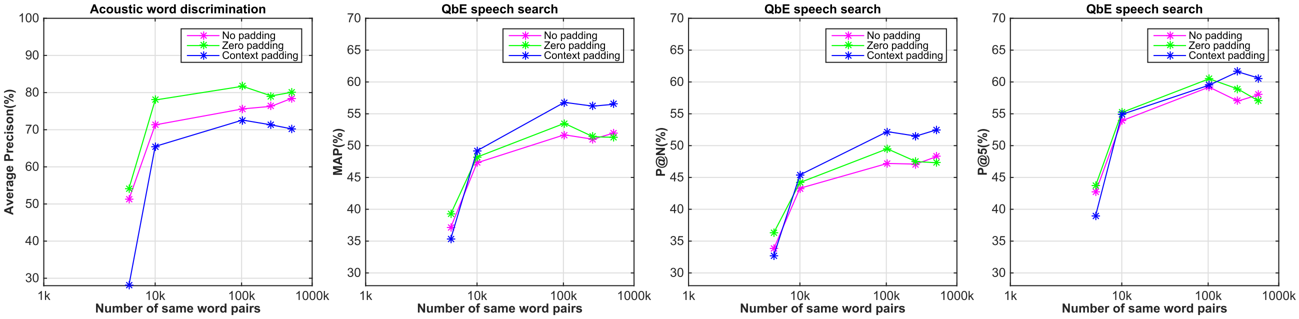


FIGURE 4. Comparison the effect of different temporal padding in the learning of recurrent neural AWEs for QbE speech search.

network with the node configuration of 1500-1500-40-1500-[412,420], where 39-dimensional FBanks with pitch features are used as network input and [412,420] denotes the number of tied triphone states of Mandarin Chinese and Spanish, respectively. The bottleneck layer with the size of 40 is in the middle of the feed-forward network.

In order to compare the proposed approach with our previous best deep CNN approach [13], experiments are conducted on the same two training sets (named as Set 1 and Set 2 respectively). Set 1 has the same vocabulary size (the number of unique spoken words) of 1,687 as in [11], and it involves 37k spoken word instances. Set 2 increases the vocabulary size to 5,476 and it involves 53k spoken word instances. Each instance has the speech duration between 0.5 and 2 seconds, and it is padded to 2 seconds with the same length of its original temporal context on both sides to form the fixed-length speech segment. Both sets can make up to 500k speech segment pairs. Five subsets are also selected from each training set. These subsets consist of $N = [M, 10k, 100k, 250k, 500k]$ speech segment pairs, where M represents the minimum number of speech segment pairs in Set 1 and Set 2 respectively. The speech segment pairs are represented by multi-lingual BNFs that were extracted from the bottleneck layer of a previously trained extractor.

Our deep BLSTM network consists of 2 layers with 512 hidden units per direction and per layer. The CNN network [13] consists of two convolutional layers, two max pooling layers, one fully connected layer with 2,048 hidden units and one fully connected layer with 1,024 hidden units. Both networks take a triplet of speech segments as input. The number of parameters in the BLSTM network is kept as comparable to that of the CNN network. Our proposed BLSTM network was implemented using the Tensorflow toolkit [48] with the configuration based on [15]. The network weights were initialized from −0.05 to 0.05. Adam optimizer [49] was used for updating the weights with the mini-batch size of 100 and the initial learning rate of 0.0001. The dropout rate was set to 0.4 at each layer and the margin in our triplet loss was 0.4. The performance of BLSTM network was fine-tuned using the development set of 10,966 speech segments every five epochs. The BLSTM network which gave the best performance on the development set was used to extract recurrent neural AWEs. After training the BLSTM network, both the keywords set of 346 spoken queries and the search content of 10 hours are used for QbE speech search stage. The duration of all the spoken queries is also between 0.5 and 2 seconds.

As the same in [23], [29], [50], the performance of search accuracy in QbE speech search stage is evaluated by three different metrics: 1) mean average precision (MAP), which is the mean of average precision for each query in search content; 2) Precision of the top N utterances in search content (P@N), where N is the number of target utterances involving the query term; 3) Precision of the top 5 utterances in search content (P@5). High precision represents better performance. In addition, the search time is recorded during QbE speech search stage, which is used to calculate the minimum cost by Eq. 9 between all the spoken queries and search content using the learned feature representations. All the tests were performed by using a computation thread on a workstation equipped with an Intel Xeon E5-2680 @ 2.7GHz CPU.

B. COMPARISON OF DIFFERENT PADDINGS

To validate the efficiency of our proposed temporal context padding in the learning of recurrent neural AWEs, three different padding methods were compared on Set 2 for both isolated word discrimination task and QbE speech search task. The isolated word discrimination task aims to calculate the distance between speech segment pairs and decide whether they contain the same or different words in the middle. We used average precision as the evaluation metric and conducted the experiments on the same test set of 11,024 speech segments as in [13].

The evaluation results on both tasks are shown in Figure 4. With limited speech segment pairs ($\leq 250k$), the performance of recurrent neural AWEs trained with zero padding is slightly better than those trained with no padding in terms of MAP/P@N/P@5, but their results get a little bit worse when speech segment pairs are increasing from 250k to 500k.

Most importantly, with sufficient speech segment pairs ($\geq 10k$ in MAP/P@N, or $\geq 250k$ in P@5), if zero padding is replaced by temporal context padding, we notice a large improvement with the best performance in QbE speech search. The relative improvements in MAP/P@N/P@5 are up to 9.3%/8.7%/4.6%, respectively. Similar results are also obtained in Set 1 with a small vocabulary. The experimental

TABLE 1. Comparison of different feature representations for QbE speech search.

Representation	Input features of paired examples	Use temporal padding?	Similarity measure	QbE speech search stage			
				MAP	P@N	P@5	Run-time (seconds)
Multi-lingual BNFs	N/A	N/A	DTW	0.400	0.365	0.485	4,752
Autoencoder features [29]	Multi-lingual BNFs	N/A	DTW	0.485	0.446	0.566	9,506
Convolutional neural AWE [13]	Multi-lingual BNFs	context	cosine	0.502	0.462	0.567	1,017
Recurrent neural AWEs	Multi-lingual BNFs	context	cosine	**0.565**	**0.525**	**0.606**	**718**
Recurrent neural AWEs	Multi-lingual BNFs	no	cosine	0.520	0.483	0.580	823

results suggest that using temporal context padding with sufficient speech segment pairs is the most efficient way to learn recurrent neural AWEs for QbE speech search. This observation is consistent with our previous approach to convolutional neural AWEs [13].

Although the recurrent neural AWEs trained using temporal context padding perform better than those trained with zero/no padding in the QbE speech search task, they achieve the worst results in the isolated word discrimination task. This result confirms our argument that there exists a clear mismatch between the learning of AWEs and its application on search content, where the actual units used for embeddings are quite different. The addition of temporal context information increases the confusion of discriminating isolated spoken words, while this information is very useful for QbE speech search instead. With the temporal context, the important neighboring information around the target words is also learned in recurrent neural AWEs via a deep BLSTM network, which reduces the mismatch for QbE speech search considerably.

C. COMPARISON OF DIFFERENT FEATURE REPRESENTATIONS

The performance of different feature representations was also compared in QbE speech search. These representations can be classified as frame-level and word-level feature representations. The frame-level feature representations include 40-dimensional multi-lingual BNFs and 100-dimensional autoencoder features. They rely on DTW at run-time QbE speech search. Learning autoencoder features is the same as [29]. The word-level feature representations include convolutional neural AWEs learned from [13] and our proposed recurrent neural AWEs, where cosine distances over these fixed-dimensional AWEs can be measured to find the matching spoken query. Here, 500*k* speech segment pairs in Set 2 were used for obtaining these pairwise learned feature representations. The performance of both search accuracy and search time are summarized in Table 1.

Multi-lingual BNFs were used as the initial input features to learn efficient frame-level or word-level feature representations. When paired examples on the target language are used as weak supervision, the learned feature representations, including autoencoder features, convolutional neural AWEs, and recurrent neural AWEs, bring significant performance improvements as compared with the baseline DTW approach with multi-lingual BNFs. This demonstrates that using paired examples as weak supervision can learn efficient frame-level or word-level feature representations.

More promisingly, when speech segment pairs are used to encode a word or segmental level speech, the generated AWEs, including both convolutional neural AWEs and recurrent neural AWEs, outperform the initial frame-level feature representations by a large margin. Meanwhile, they significantly reduce a lot of search run-time because they avoid time-consuming DTW in QbE speech search.

Most importantly, recurrent neural AWEs outperform convolutional neural AWEs on QbE speech search no matter what temporal padding is used. The relative improvements in MAP/P@N/P@5 are up to 12.5%/13.6%/6.9%, respectively. This suggests that recurrent neural AWEs are more capable of modeling temporal dependency in speech than convolutional neural AWEs and thus result in better performance in QbE speech search.

Moreover, our proposed recurrent neural AWEs trained with temporal context padding hold the best performance in terms of both search accuracy and search time. The relative improvements in MAP/P@N/P@5 are 8.7%/8.7%/4.5%, respectively. The small reduction of search time (from 823 to 718 seconds) is due to fewer speech segment candidates generated by the temporal context padding when shifting on search content. In summary, the experiments demonstrate that the temporal context is critical to the use of recurrent neural AWEs for QbE speech search.

D. EFFECT OF NUMBERS OF SPEECH SEGMENT PAIRS

We further investigated how the number of speech segment pairs affect the performance of our proposed recurrent neural AWEs for QbE speech search. The evaluation results of all the subsets from both Set 1 and Set 2 are plotted in Figure 5.

With more speech segment pairs, the learned AWEs have a better performance for QbE speech search in most instances. More importantly, no matter how many numbers of speech pairs are used, our proposed recurrent neural AWEs consistently outperform convolutional neural AWEs [13]. When the number of speech segment pairs is fixed (e. g., $\geq 10k$ QbE speech search), the recurrent neural AWEs trained on Set 2 outperform those trained on Set 1. These observations suggest that it is important to use more speech segment pairs with a larger vocabulary size for learning recurrent neural AWEs.

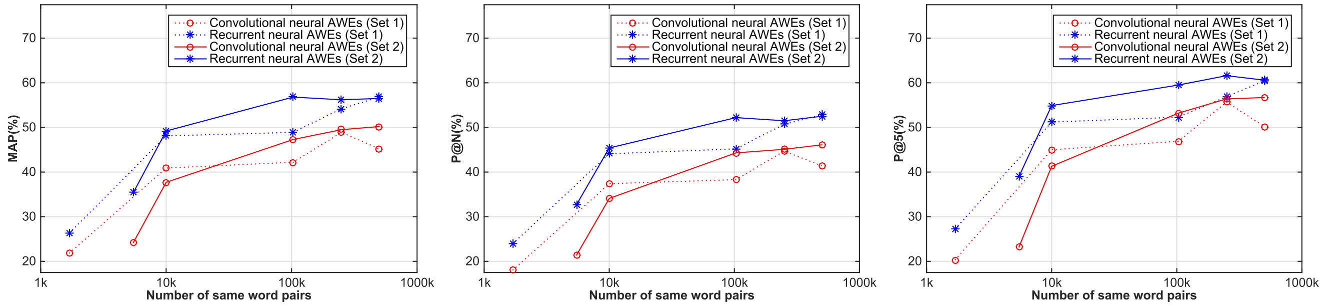


FIGURE 5. Comparison the effect of using different speech segment pairs in the learning of AWEs for QbE speech search.

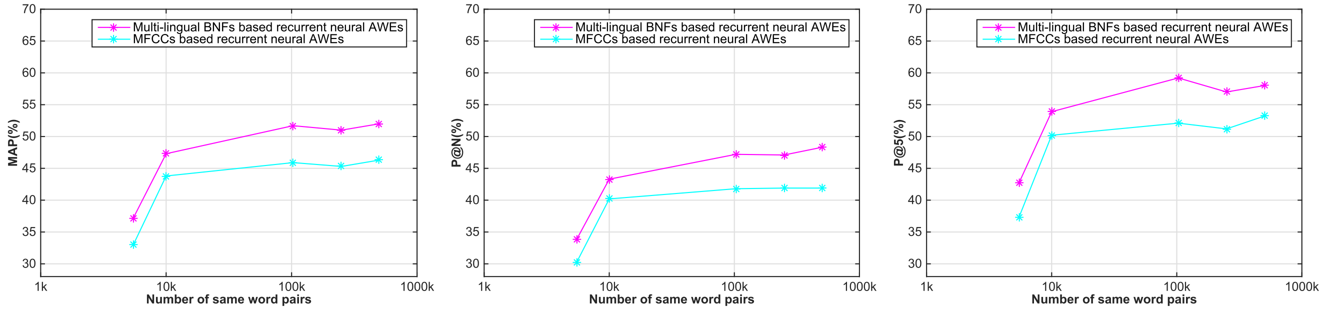


FIGURE 6. Comparison the effect of using different input features in the learning of recurrent neural AWEs for QbE speech search.

E. BNFs VS MFCCs AS NEURAL INPUT

Finally, we compared 40-dimensional multi-lingual BNFs with 39-dimensional MFCCs to investigate how the network input features affect the learned recurrent neural AWEs. Notice that the MFCCs are commonly used as features in acoustic modeling. Results in Figure 6 show that multilingual BNFs consistently perform MFCCs in the learning of recurrent neural AWEs for QbE speech search, no matter the training data size. These results suggest that choosing efficient input features is also important. Multi-lingual BNFs, which have more capability in phonetic discrimination than MFCCs, result in better recurrent neural AWEs for QbE speech search.

F. DISCUSSION

Our above findings suggest that leanring recurrent neural AWEs with temporal context is beneficial to QbE speech search. It learns the important neighboring information around the target word, by reducing the mismatch between the AWE network training and its application on search content. As compared with the previous state-of-art features, the proposed recurrent neural AWEs achieve superior performance in terms of both search accuracy and efficiency.

V. CONCLUSION

We have proposed to include the temporal context information in a deep BLSTM network to learn recurrent neural AWEs with strong discrimination ability for QbE speech search. The introduction of temporal context aims to reduce the mismatch between the AWE network training and its application on search content. More specifically, the AWE networks are usually trained using isolated spoken words, while without the word boundary information during speech search, a fixed length window, which may contain a partial word or more words, is used to slide over the search content for similarity measure. Our study has shown that leanring recurrent neural AWEs with temporal context is essential to alleviate the mismatch. Our study has also indicated that sufficient speech segment pairs with rich vocabulary coverage and more discriminative input features are both important to the AWE based QbE speech search. In the future, we will try learning AWEs using word pairs discovered in an automatic manner from untrascribed speech corpus. In this way, the AWEs can be learned in a fully unsupervised way for QbE speech search.

REFERENCES

[1] T. J. Hazen, W. Shen, and C. White, “Query-by-example spoken term detection using phonetic posteriorgram templates,” in *Proc. ASRU*, 2009, pp. 421–426.

[2] Y. Zhang and J. R. Glass, “Unsupervised spoken keyword spotting via segmental dtw on Gaussian posteriorgrams,” in *Proc. ASRU*, 2009, pp. 398–403.

[3] N. Rajput and F. Metze, “Spoken web search,” in *Proc. MediaEval Workshop*, 2011, pp. 1–2.

[4] F. Metze, E. Barnard, M. Davel, C. Van Heerden, X. Anguera, G. Gravier, and N. Rajput, “The spoken web search task,” in *Proc. MediaEval Workshop*, 2012, pp. 1–2.

[5] X. Anguera, F. Metze, A. Buzo, I. Szoke, and L. J. Rodriguez-Fuentes, “The spoken web search task,” in *Proc. MediaEval Workshop*, 2013.

[6] X. Anguera, L. J. Rodriguez-Fuentes, and I. Szöke, A. Buzo, and F. Metze, “Query by example search on speech at mediaeval 2014,” in *Proc. MediaEval Workshop*, 2014, pp. 1–2.

[7] I. Szöke, L. J. Rodriguez-Fuentes, A. Buzo, X. Anguera, F. Metze, J. Proenca, M. Lojka, and X. Xiong, “Query by example search on speech at mediaeval 2015,” in *Proc. MediaEval Workshop*, 2015, pp. 1–3.

[8] Y. Zhang, R. Salakhutdinov, H.-A. Chang, and J. Glass, ''Resource configurable spoken query detection using deep boltzmann machines,'' in *Proc. ICASSP*, Mar. 2012, pp. 5161–5164.

[9] J. Tejedor, M. Fapšo, I. Szöke, J. Černocký, and F. Grézl, ''Comparison of methods for language-dependent and language-independent query-by-example spoken term detection,'' *ACM Trans. Inf. Syst.*, vol. 30, no. 3, p. 18, 2012.

[10] H. Kamper, W. Wang, and K. Livescu, ''Deep convolutional acoustic word embeddings using word-pair side information,'' in *Proc. ICASSP*, 2016, pp. 4950–4954.

[11] Y. Yuan, C.-C. Leung, L. Xie, B. Ma, and H. Li, ''Learning neural network representations using cross-lingual bottleneck features with word-pair information,'' in *Proc. INTERSPEECH*, 2016, pp. 788–792.

[12] S. Settle, K. Levin, H. Kamper, and K. Livescu, ''Query-by-example search with discriminative neural acoustic word embeddings,'' in *Proc. INTERSPEECH*, 2017, pp. 2874–2878.

[13] Y. Yuan, C.-C. Leung, L. Xie, H. Chen, B. Ma, and H. Li, ''Learning acoustic word embeddings with temporal context for query-by-example speech search,'' in *Proc. INTERSPEECH*, 2018, pp. 97–101.

[14] S. Settle and K. Livescu, ''Discriminative acoustic word embeddings: Tecurrent neural network-based approaches,'' in *Proc. SLT*, 2016, pp. 503–510.

[15] W. He, W. Wang, and K. Livescu, ''Multi-view recurrent neural acoustic word embeddings,'' in *Proc. ICLR*, 2017, pp. 1–12.

[16] Y.-A. Chung, C.-C. Wu, C.-H. Shen, H.-Y. Lee, and L.-S. Lee, ''Audio word2vec: Unsupervised learning of audio segment representations using sequence-to-sequence autoencoder,'' in *Proc. INTERSPEECH*, 2016, pp. 765–769.

[17] Y.-A. Chung and J. Glass, ''Learning word embeddings from speech,'' Nov. 2017, *arXiv:1711.01515*. [Online]. Available: https://arxiv.org/abs/1711.01515

[18] K. Levin, K. Henry, A. Jansen, and K. Livescu, ''Fixed-dimensional acoustic embeddings of variable-length segments in low-resource settings,'' in *Proc. ASRU*, 2013, pp. 410–415.

[19] Y. Zhang and J. R. Glass, ''Towards multi-speaker unsupervised speech pattern discovery,'' in *Proc. ICASSP*, 2010, pp. 4366–4369.

[20] G. Mantena, S. Achanta, and K. Prahallad, ''Query-by-example spoken term detection using frequency domain linear prediction and non-segmental dynamic time warping,'' *IEEE Trans. Audio, Speech, Language Process.*, vol. 22, no. 5, pp. 946–955, May 2014.

[21] H. Chen, C.-C. Leung, L. Xie, B. Ma, and H. Li, ''Parallel inference of Dirichlet process Gaussian mixture models for unsupervised acoustic modeling: A feasibility study,'' in *Proc. INTERSPEECH*, 2015, pp. 3189–3193.

[22] H. Wang, T. Lee, and C.-C. Leung, ''Unsupervised spoken term detection with acoustic segment model,'' in *Proc. COCOSDA*, 2011, pp. 106–111.

[23] H. Wang, C.-C. Leung, T. Lee, B. Ma, and H. Li, ''An acoustic segment modeling approach to query-by-example spoken term detection,'' in *Proc. ICASSP*, 2012, pp. 5157–5160.

[24] H. Wang, T. Lee, C.-C. Leung, B. Ma, and H. Li, ''Using parallel tokenizers with dtw matrix combination for low-resource spoken term detection,'' in *Proc. ICASSP*, 2013, pp. 8545–8549.

[25] C.-A. Chan and L.-S. Lee, ''Model-based unsupervised spoken term detection with spoken queries,'' *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 7, pp. 1330–1342, Jul. 2013.

[26] F. Grézl, M. Karafiát, S. Kontar, and J. Černocký, ''Probabilistic and bottle-neck features for LVCSR of meetings,'' in *Proc. ICASSP*, 2007, pp. 753–757.

[27] K. Veselỳ and M. Karafiát, and F. Grézl, ''Convolutive bottleneck network features for lvcsr,'' in *Proc. ASRU*, 2011, pp. 42–47.

[28] C.-C. Leung, L. Wang, H. Xu, J. Hou, V. T. Pham, H. Lv, L. Xie, X. Xiao, C. Ni, B. Ma, E. S. Chng, and H. Li, ''Toward high-performance language-independent query-by-example spoken term detection for mediaeval 2015: Post-evaluation analysis,'' in *Proc. INTERSPEECH*, 2016, pp. 3703–3707.

[29] Y. Yuan, C.-C. Leung, L. Xie, H. Chen, B. Ma, and H. Li, ''Pairwise learning using multi-lingual bottleneck features for low-resource query-by-example spoken term detection,'' in *Proc. ICASSP*, 2017, pp. 5645–5649.

[30] H. Kamper, M. Elsner, A. Jansen, and S. Goldwater, ''Unsupervised neural network based feature extraction using weak top-down constraints,'' in *Proc. ICASSP*, 2015, pp. 5818–5822.

[31] A. Jansen and B. Van Durme, ''Efficient spoken term discovery using randomized algorithms,'' in *Proc. ASRU*, 2011, pp. 401–406.

[32] D. Renshaw, H. Kamper, A. Jansen, and S. Goldwater, ''A comparison of neural network methods for unsupervised representation learning on the zero resource speech challenge,'' in *Proc. INTERSPEECH*, 2015, pp. 3199–3203.

[33] Y. Yuan, C.-C. Leung, L. Xie, H. Chen, B. Ma, and H. Li, ''Extracting bottleneck features and word-like pairs from untranscribed speech for feature representation,'' in *Proc. ASRU*, 2017, pp. 734–739.

[34] J. Bromley, I. Guyon, Y. LeCun, and E. Säckinger, and R. Shah, ''Signature verification using a 'siamese' time delay neural network,'' in *Proc. NIPS*, 1994, pp. 737–744.

[35] J. Mueller and A. Thyagarajan, ''Siamese recurrent architectures for learning sentence similarity,'' in *Proc. AAAI*, 2016, pp. 2786–2792.

[36] A. Jansen, S. Thomas, and H. Hermansky, ''Weak top-down constraints for unsupervised acoustic model training,'' in *Proc. ICASSP*, 2013, pp. 8091–8095.

[37] G. Synnaeve, T. Schatz, and E. Dupoux, ''Phonetics embedding learning with side information,'' in *Proc. SLT*, 2014, pp. 106–111.

[38] H. Kamper, A. Jansen, S. King, and S. Goldwater, ''Unsupervised lexical clustering of speech segments using fixed-dimensional acoustic embeddings,'' in *Proc. SLT*, 2014, pp. 100–105.

[39] H. Kamper, A. Jansen, and S. Goldwater, ''Fully unsupervised small-vocabulary speech recognition using a segmental Bayesian model,'' in *Proc. INTERSPEECH*, 2015, pp. 678–682.

[40] H. Kamper, A. Jansen, and S. Goldwater, ''A segmental framework for fully-unsupervised large-vocabulary speech recognition,'' *Comput. Speech Lang.*, vol. 46, pp. 154–174, Nov. 2017.

[41] K. Levin, A. Jansen, and B. Van Durme, ''Segmental acoustic indexing for zero resource keyword search,'' in *Proc. ICASSP*, 2015, pp. 5828–5832.

[42] A. L. Maas, S. D. Miller, T. M. O'Neil, A. Y. Ng, and P. Nguyen, ''Word-level acoustic modeling with convolutional vector regression,'' in *Proc. ICML*, 2012, pp. 1–8.

[43] S. Bengio and G. Heigold, ''Word embeddings for speech recognition,'' in *Proc. INTERSPEECH*, 2014, pp. 1053–1057.

[44] Y.-A. Chung, C.-C. Wu, C.-H. Shen, and H.-Y. Lee, ''Unsupervised learning of audio segment representations using sequence-to-sequence recurrent neural networks,'' in *Proc. INTERSPEECH*, 2016, pp. 765–769.

[45] C.-W. Ao and H.-Y. Lee, ''Query-by-example spoken term detection using attention-based multi-hop networks,'' in *Proc. ICASSP*, 2018, pp. 6264–6268.

[46] G. Chen, C. Parada, and T. N. Sainath, ''Query-by-example keyword spotting using long short-term memory networks,'' in *Proc. ICASP*, 2015, pp. 5236–5240.

[47] F. Schroff, D. Kalenichenko, and J. Philbin, ''FaceNet: A unified embedding for face recognition and clustering,'' in *Proc. CVPR*, 2015, pp. 815–823.

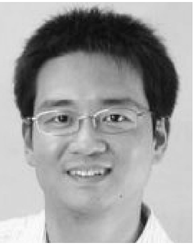
[48] M. Abadi *et al.*, ''TensorFlow: Large-scale machine learning on heterogeneous distributed systems,'' Mar. 2016, *arXiv:1603.04467*. [Online]. Available: https://arxiv.org/abs/1603.04467

[49] D. Kingma and J. Ba, ''Adam: A method for stochastic optimization,'' in *Proc. ICLR*, 2015, pp. 1–15.

[50] H. Chen, C.-C. Leung, L. Xie, B. Ma, and H. Li, ''Unsupervised bottleneck features for low-resource query-by-example spoken term detection,'' in *Proc. INTERSPEECH*, 2016, pp. 923–927.

YOUGEN YUAN received the B.E. degree in computer science and technology from Chongqing University, China, in 2014. He is currently pursuing the Ph.D. degree with the School of Computer Science, Northwestern Polytechnical University, China. In 2016, he visited the Institute for Infocomm Research, Singapore, as an intern, for a year. In 2017, he was a Joint-Training Student with the Department of Electrical and Computer Engineering, National University of Singapore. His current research interests include automatic speech recognition and spoken document retrieval.

CHEUNG-CHI LEUNG received the B.E degree from The University of Hong Kong, in 1999, and the M.Phil. and Ph.D. degrees from The Chinese University of Hong Kong, in 2001 and 2004, respectively. From 2004 to 2008, he was with the Spoken Language Processing Group, CNRS-LIMSI, France, as a Postdoctoral Researcher. In 2008, he joined the Institute for Infocomm Research, Singapore, where he was with the Human Language Technology Department, for ten years. In 2018, he joined the A.I. Research Lab, Alibaba, Singapore, where he is currently a Research Scientist. His current research interests include automatic speech recognition, spoken document retrieval, and spoken language recognition.



LEI XIE received the Ph.D. degree in computer science from Northwestern Polytechnical University (NPU), Xi'an, China, in 2004, where he is currently a Professor with the School of Computer Science. From 2001 to 2002, he was with the Department of Electronics and Information Processing, Vrije Universiteit Brussel (VUB), Brussels, Belgium, as a Visiting Scientist. From 2004 to 2006, he was a Senior Research Associate with the School of Creative Media, City University of Hong Kong, Hong Kong. From 2006 to 2007, he was a Postdoctoral Fellow with the Human Computer Communications Laboratory (HCCL), Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong. He has published over 200 papers in major journals and proceedings, such as the IEEE Transactions on Audio, Speech, and Language Processing, the IEEE Transactions on Multimedia, the *Pattern Recognition*, the *ACM Multimedia*, the ACL, INTERSPEECH, and ICASSP. He is currently an Associate Editor of the IEEE/ACM Transactions on Audio, Speech and Language Processing. His current research interests include audio, speech, and language processing, multimedia and human–computer interaction.

HONGJIE CHEN received the B.E. degree from the School of Computer Science, Northwestern Polytechnical University, China, in 2013, where he is currently pursuing the Ph.D. degree with the School of Computer Science. In 2014, he visited the Institute for Infocomm Research, Singapore, as an Intern. In 2017, he served as a research student with the Department of Electrical and Computer Engineering, National University of Singapore. His current research interests include automatic speech recognition and spoken document retrieval.

BIN MA received the Ph.D. degree in computer engineering from The University of Hong Kong, in 2000. He joined Lernout & Hauspie Asia Pacific, in 2000, as a Researcher, working on speech recognition. From 2001 to 2004, he worked for InfoTalk Corp., Ltd., as a Senior Researcher, and a Senior Technical Manager for speech recognition. In 2004, he joined the Institute for Infocomm Research, Singapore, where he worked as a Senior Scientist and the Lab Head of Speech Recognition. He has served as the Subject Editor for *Speech Communication* (2009–2012) and an Associate Editor for the IEEE/ACM Transactions on Audio, Speech, and Language Processing (2014–2017). He has also served as the Technical Program Co-Chair for INTERSPEECH 2014 and the Technical Program Chair for ASRU 2019. He is currently a Principal Engineer with the R&D Center Singapore, Machine Intelligence Technology, Alibaba. His current research interests include robust speech recognition, speaker and language recognition, spoken document retrieval, natural language processing, and machine learning.