Phonetic Subspace Features for Improved Query by Example Spoken Term Detection

Dhananjay Ram[a,b,*], Afsaneh Asaei[a], Hervé Bourlard[a,b]

{*dhananjay.ram, afsaneh.asaei, herve.bourlard*}*@idiap.ch*

[a]*Idiap Research Institute, Martigny, Switzerland*
[b]*Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland*

Abstract

This paper addresses the problem of detecting speech utterances from a large audio archive using a simple spoken query, hence referring to this problem as "Query by Example Spoken Term Detection" (QbE-STD). This still open pattern matching problem has been addressed in different contexts, often based on variants of the Dynamic Time Warping (DTW) algorithm. In the work reported here, we exploit Deep Neural Networks (DNN) and the so inferred phone posteriors to better model the phonetic subspaces and, consequently, improve the QbE-STD performance. Those phone posteriors have indeed been shown to properly model the union of the underlying low-dimensional phonetic subspaces. Exploiting this property, we investigate here two methods relying on sparse modeling and linguistic knowledge of sub-phonetic components. Sparse modeling characterizes the phonetic subspaces through a dictionary for sparse coding. Projection of the phone posteriors through reconstruction on the corresponding subspaces using their sparse representation enhance those phone posteriors. On the other hand, linguistic knowledge driven sub-phonetic structures are identified using phonological posteriors which consists of the probabilities of phone attributes estimated by DNNs, resulting in a new set of feature vectors. These phonological posteriors provide complementary information and a distance fusion method is proposed to integrate information from phone and phonological posterior features. Both posterior features are used for query detection using DTW and evaluated on AMI database. We demonstrate that the subspace enhanced phone posteriors obtained using sparse reconstruction outperforms the conventional DNN posteriors. The distance fusion technique gives further improvement in QbE-STD performance.

Keywords: Deep neural network, Phone posterior, Phonological posterior, Sparse representation, Dictionary learning, Query by Example, Spoken Term Detection

1. Introduction

Query by Example Spoken Term Detection (QbE-STD) refers to the task of detecting all audio documents from a database such that the documents contain a spoken query provided by a user. This enables the users to search over spoken audio archives using their own speech. The primary difference between QbE-STD and keyword spotting is that the user provides one or more examples of a spoken query instead of a textual query. In general, the query examples as well as test utterances can be spoken by different speakers in varying acoustic conditions without any constraints on the language and corresponding vocabulary. Since no training data is required nor provided, QbE-STD is a particular case of a zero-resource task.

A QbE-STD system is useful for searching through audio data generated by news channels, radio broadcasts, internet etc. These audio contents are produced everyday in multiple languages by a large number of diverse users. Due to the lack of knowledge about the language of interest and corresponding training data,

*Corresponding author

it is difficult to build an automatic speech recognition (ASR) system and integrate it to a text based retrieval system to perform QbE-STD (Lee et al., 2015). Therefore, recent advances in QbE-STD are largely dominated by template matching techniques for its superior performance in zero-resource condition (Anguera et al., 2014; Rodriguez-Fuentes et al., 2014). The template based QbE-STD system primarily involves two steps: (1) extraction of feature vectors from the spoken query and the test audio, and (2) alignment of the query and test features using dynamic time warping (DTW) (Rabiner et al., 1978) or one of its variants (Müller, 2007; Zhang and Glass, 2009). Phone posterior features (posterior probabilities of a set of phonetic classes) (Hazen et al., 2009; Rodriguez-Fuentes et al., 2014) and bottleneck features (representation obtained from the bottleneck layer of a deep neural network) (Szöke et al., 2014; Chen et al., 2017) have been successful for QbE-STD. These features are extracted from deep neural networks (DNN) trained using multiple well-resourced languages. The bottleneck features can also be extracted in an unsupervised manner using labels generated from clustering techniques (Chen et al., 2016).

Our earlier attempts at QbE-STD rely on low-dimensional subspace structure of speech signal (Ram et al., 2015, 2016, 2017, 2018a). This structure of speech can be attributed to the constrained configuration of the human speech production system, leading to the generation of speech signals lying on low-dimensional, non-linear manifolds (Deng, 2004; King et al., 2007). The low-dimensional structure is exploited using sparse representation of speech data and QbE-STD is cast as a subspace detection problem between query and non-query speech (Ram et al., 2015, 2016, 2018a). This property of speech has also been exploited to perform robust speech recognition (Sainath et al., 2011; Gemmeke et al., 2011) as well as enhanced acoustic modeling (Dighe et al., 2016b). Our method presented a faster approach than template matching, however it lacked a framework to capture the temporal information inherent to speech. In contrast, we propose here to exploit the low-dimensional properties to obtain a better representation of the speech signal, before performing QbE-STD using DTW based template matching. In this way, we exploit the temporal information as well as low-dimensional structure of speech signal. To achieve this goal, we propose a data-driven and a knowledge-based approach to obtain better representation of speech and a fusion technique to combine information from different kinds of representations as discussed below.

(i) *Phonetic Subspace Representation - A data-driven approach (Section 4)*: We propose to use sparse modeling as an unsupervised data-driven method to characterize the low-dimensional structures of sub-phonetic components (Elhamifar and Vidal, 2013; Rish and Grabarnik, 2014). To that end, we model the underlying phonetic subspaces using dictionary learning for sparse coding. The dictionaries are used to obtain sparse representation of the phone posteriors and we project them onto the phonetic subspaces through reconstruction. This approach leads to subspace enhanced phone posteriors such that the query and test posteriors are represented on a common subspace and reduces the effect of unstructured phonetic variations.

(ii) *Phonetic Subspace Representation - A knowledge-based approach (Section 5)*: Alternative to the data-driven sparse modeling approach, we utilize linguistic knowledge for identifying the sub-phonetic attributes or phonological features (Chomsky and Halle, 1968). The phonological features are recognized as the atomic components of phone construction. The linguists define a binary mapping between the phone and phonological categories. We exploit DNN in probabilistic characterization of the phonological features, referred to as the *phonological posteriors* (Cernak et al., 2017). Due to the sub-phonetic nature of these features, they are less language dependent (Lee and Siniscalchi, 2013; Sahraeian et al., 2015) and can be helpful for a zero resource task like QbE-STD.

(iii) *Distance fusion (Section 6)*: The proposed representations are exploited for QbE-STD using the DTW method presented in (Rodriguez-Fuentes et al., 2014) (see Section 3 for details). To integrate the information from multiple feature representations, we propose to update the distance matrix for DTW by fusing the distances between the query and test utterance obtained from different kinds of feature representations. In contrast to (Wang et al., 2013), we use non-uniform weights which are optimized using development queries.

The proposed methods are evaluated on two subsets of AMI database (IHM and SDM) with challenging conditions as presented in Section 8. The improvements obtained by our approach over the baseline system indicate the significance of subspace structure of speech for QbE-STD.

2. Related Works

In this section, we summarize different techniques proposed for QbE-STD. The first set of methods consists of a two step approach: feature extraction and template matching as discussed earlier. The spoken queries as well as test utterances can be represented using mel frequency cepstral coefficient (MFCC) or perceptual linear prediction (PLP) based spectral features. These spectral features were initially investigated for template matching task (Sakoe and Chiba, 1978). However these features were outperformed by posterior features, which can be estimated from models trained in both supervised and unsupervised manner (Hazen et al., 2009; Rodriguez-Fuentes et al., 2014; Zhang and Glass, 2009). Gaussian mixture model (GMM) based posteriors are estimated from a GMM trained in an unsupervised manner where the feature dimensions correspond to posterior probabilities of different Gaussian components in the model (Zhang and Glass, 2009; Park and Glass, 2008). On the other hand, a deep boltzman machine (DBM) trained in unsupervised as well as semi-supervised manner can be used to extract posterior features. The unsupervised training of DBM can capture hierarchical structural information from unlabeled data. In (Zhang et al., 2012), the authors first train a DBM using unlabeled data and then fine tune it using small amount of labeled data. In another approach, GMM based posteriors were used as labels for the DBM training (Zhang et al., 2012). Posteriors from DBM in both cases perform better than GMM posteriors for QbE-STD.

The supervised approach to extract posterior features primarily relies on training a DNN using labeled data. In case of zero resource languages, the DNN is first trained using data from different well resourced languages where the labels can indicate mono-phones, context dependent phones or senones (Hazen et al., 2009; Rodriguez-Fuentes et al., 2014). The DNN is then used to extract posterior features to perform template matching for QbE-STD. In this approach, the posteriors are interpreted as a characterization of instantaneous content of the speech signal, irrespective of the underlying language (Rodriguez-Fuentes et al., 2014). DNNs with bottleneck layer have also been trained in a similar multilingual setting to compute bottleneck features for QbE-STD (Szöke et al., 2014; Chen et al., 2017).

Features extracted from the spoken query and test utterance are used to compute a frame-level distance matrix and a DTW algorithm is used to find the degree of similarity between them. Standard DTW algorithm performs an end-to-end comparison between two temporal sequences, making it difficult to use for QbE-STD because the query can occur anywhere in the test utterance as a sub-sequence. In segmental DTW (Park and Glass, 2008), the distance matrix is segmented into overlapping diagonal bands where the width of the band indicates temporal distortion allowed for matching. But the width of each band limits its capability to deal with signals of widely varying speaking rate. Slope-constrained DTW (Zhang and Glass, 2009) was proposed to deal with this problem by penalizing the slope of warping path which maps the spoken query within a test utterance. It limits the number of frames to be mapped in the test audio corresponding to a frame in the query and vice versa. In sub-sequence DTW (Müller, 2007), the cost of insertion is forced to be 0 in the beginning and end of a query, which enables the warping path to begin and end at any point in the test audio and finds a sub-sequence best matching the query.

More recent approaches are aimed at minimizing the computational cost or memory footprints of the DTW based search techniques. Information retrieval based DTW (Anguera, 2013) proposes to index the frames of test utterance and uses hashing techniques to reduce the search space. On the other hand, memory efficient DTW (Anguera and Ferrarons, 2013) proposed an improvement over subsequence DTW by using a lookup table for faster backtracking and an alternative normalization of the warping path. Alternative to DTW, subspace detection based approach relying on frame level detection scores have been proposed to make the search faster (Ram et al., 2015, 2016, 2018a). DTW based template matching can be replaced by a Convolutional Neural Network (CNN) with the distance matrix as input images to achieve better performance (Ram et al., 2018b). Additionally, model based approaches have been proposed to deal with acoustic and speaker mismatch conditions. These methods depend on unsupervised acoustic unit discovery, followed by the use of hidden Markov models (HMM) to model those units. These HMMs are then used

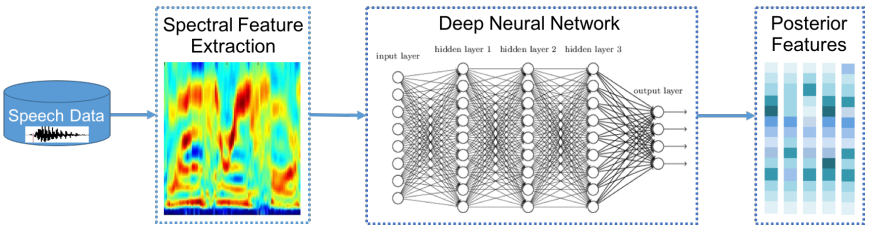


Figure 1: Phone posterior estimation using a deep neural network (DNN): First, MFCC based spectral features are extracted from windowed speech. These features are then fed to a DNN with some acoustic context (left and right) to estimate the phone posterior probabilities.

to find symbolic representation of the query and test utterance, and symbolic search techniques are used to retrieve the query.

3. Baseline System

In this section, we briefly describe the spoken query detection system presented in (Rodriguez-Fuentes et al., 2014) (best system in MediaEval challenge (Anguera et al., 2013)) which we use as our baseline system for its superior performance. In this approach, phone posterior features are extracted from both query and test audio followed by a template matching technique to find the similarity score as discussed in the following.

3.1. Phone Posteriors

Phone posterior feature is a vector representation consisting of phonetic class conditional posterior probabilities given a short window of the acoustic features, typically estimated using a DNN (Bourlard and Morgan, 1994; Hinton et al., 2012). These posteriors are recognized as one of the most efficient speech representation for QbE-STD (Hazen et al., 2009; Rodriguez-Fuentes et al., 2014). The setup for phone posterior estimation is depicted in Figure 1. In the first step, Mel Frequency Cepstral Coefficient (MFCC) features are extracted from the speech signal over a sliding temporal window. Those MFCC features along with some acoustic context (left and right) are used as input to train a DNN for estimating the phone posterior probabilities. MFCC features of test data are then forward passed through the trained DNN to compute corresponding phone posterior vectors.

3.2. Template Matching

Phone posteriors, from both queries and test utterances, are used to compute frame-level distance matrices employing logarithm of cosine distance as distance measure. These distances are further normalized to vary between 0 and 1 before using DTW to obtain a warping path. The DTW based template matching proposed in (Rodriguez-Fuentes et al., 2014) is similar to the slope constrained DTW (Hazen et al., 2009). It enforces the cost of insertion at the first and last frames of the query to be 0. This helps the warping path to start and end at any frame of the test utterance, which gives us a sub-sequence matching the spoken query. At each step, the DTW algorithm compares the accumulated distances normalized by the corresponding path lengths to avoid preference for shorter paths. The normalized distance of a hypothesized sub-sequence from the test utterance is considered as its scores. The sub-sequences with durations less than half of the query duration are rejected to reduce the false alarm rate. The normalized distances are compared with a pre-defined threshold to take the final decision. A block diagram of this resulting system is presented in Figure 2.

4. Phonetic Subspace Representation: A data-driven approach

In this section, we present a data-driven approach for modeling the underlying low-dimensional subspaces which constitute different phonetic units in a language. These models are then used to enhance the phone posterior features by reducing the effect of unstructured noise present in the data.

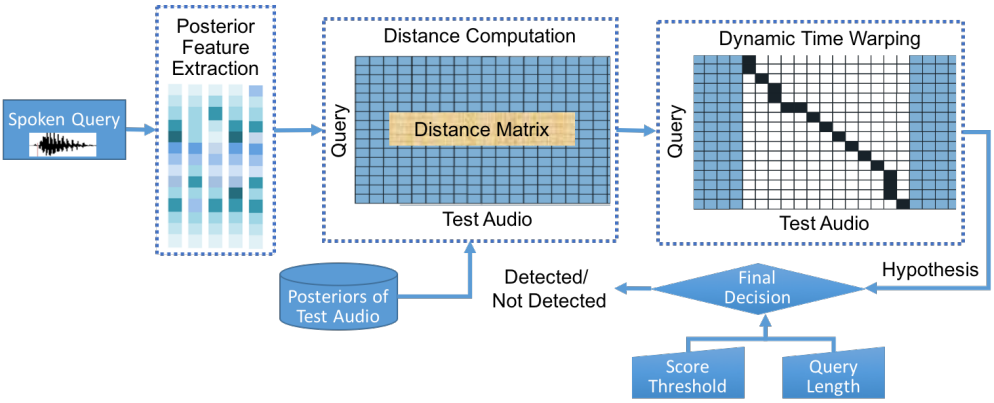


Figure 2: Block diagram of the baseline DTW system: Phone posterior features are estimated for both spoken query and test utterance to compute a normalized distance matrix. DTW algorithm is then used to obtain a hypothesized sub-sequence of the test utterance matching the query. A hypothesis of length less than half the query length is discarded to reduce false alarm rate. Finally, the score of a valid hypothesis is compared to a threshold to yield final decision.

4.1. Sparse Subspace Modeling

We consider sparse coding as a data-driven unsupervised technique to characterize the subspace structure of phone posteriors. Earlier studies have shown that the phone posterior vectors belong to a union of low-dimensional subspaces (Ram et al., 2015, 2016, 2018a; Dighe et al., 2016a). Any data point in these subspaces can be efficiently reconstructed using a sparse linear combination of other points in that space. This property is referred to as the self-expressiveness (Elhamifar and Vidal, 2013) of data. In practice, an over-complete set of basis vectors, called dictionary is learned from the training data to model the underlying subspaces. It is learned in a manner such that each training vector can be reconstructed as a sparse linear combination of its columns. The columns of a dictionary are known as the atoms forming the molecular structure of the phone posteriors.

Formally speaking, any data point $\mathbf{y}_t$ belonging to the space of phone k can be expressed as a sparse linear combination of the atoms present in the corresponding dictionary as $\mathbf{y}_t = \mathbf{D}_k \boldsymbol{\alpha}_t$ where $\mathbf{D}_k \in \mathbb{R}^{K \times M_k}$ consists of the over-complete basis vectors (atoms) used to model the subspaces of phone k and $\boldsymbol{\alpha}_t$ is the sparse weight vector indicating the significance of each atom to construct the posterior vector. Here, M_k is the number of atoms in the k-th dictionary and K is the dimension of each atom as well as the number of phone classes.

In order to obtain a sparse representation of a posterior vector, we require dictionaries modeling the underlying subspaces. For this purpose, we learn phone-specific dictionaries using the training posteriors. The data used to train the DNN for phone posterior estimation are used here (after forward passing through the DNN) to train these dictionaries. Let us consider a set of T_k training posterior vectors, $\mathbf{Y}_k = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{T_k}\}$ belonging to phone class k. Their sparse representations are denoted by $\mathbf{A}_k = \{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_{T_k}\}$. The objective function for dictionary learning is expressed as

$$\mathbf{D}_k = \arg\min_{\mathbf{D}, \mathbf{A}_k} \frac{1}{T_k} \sum_{t=1}^{T_k} \left(\frac{1}{2} \|\mathbf{y}_t - \mathbf{D}\,\boldsymbol{\alpha}_t\|_2^2 + \lambda \|\boldsymbol{\alpha}_t\|_1 \right) \tag{1}$$

where λ is the regularization parameter. The first term in this expression represents the reconstruction error. The second term denotes the ℓ_1-norm of $\boldsymbol{\alpha}$ defined as $\|\boldsymbol{\alpha}\|_1 = \sum_i |\alpha_i|$ which quantifies the level of sparsity of $\boldsymbol{\alpha}_t$. The joint optimization of this objective function with respect to both $\mathbf{D}$ and $\mathbf{A}_k$ simultaneously is non-convex, however it can be solved as a convex objective by optimizing for one while keeping the other fixed (Mairal et al., 2010).

In this work, we have used the fast online algorithm proposed in (Mairal et al., 2010) which was found to be effective for sparse modeling of the phone posterior (Dighe et al., 2016a; Ram et al., 2016). This algorithm is based on stochastic gradient descent optimization and is summarized in Algorithm 1; it alternates between

Algorithm 1 Dictionary Learning for Sparse Modeling of Phone k

Require: : $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{T_k}\}, \lambda, \mathbf{D}^{(0)}$ (initialization)

1: **for** $t = 1$ `to` T_k **do**

2: Sparse representation of $\mathbf{y}_t$ to determine $\boldsymbol{\alpha}_t$:
$\boldsymbol{\alpha}_t = \arg\min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2}\|\mathbf{y}_t - \mathbf{D}^{(t-1)}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1 \right\}$

3: Updating $\mathbf{D}^{(t)}$ with $\mathbf{D}^{(t-1)}$ as warm restart:
$\mathbf{D}^{(t)} = \arg\min_{\mathbf{D}} \left\{ \frac{1}{t}\sum_{i=1}^{t} (\frac{1}{2}\|\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1) \right\}$

4: **end for**

5: **return** $\mathbf{D}_k = \mathbf{D}^{(T_k)}$

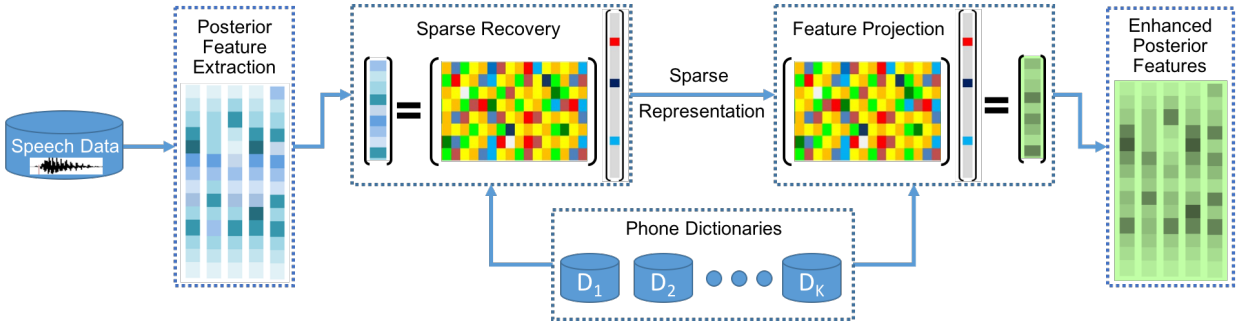


Figure 3: Block diagram to obtain subspace enhanced phone posteriors using sparse representation based reconstruction: Phone posterior features are extracted from speech signal. Sparse representations for these feature vectors are obtained using phone dictionaries. These sparse representations are then multiplied to the corresponding dictionary matrix in order to compute the enhanced phone posteriors.

a step of sparse representation for the current training feature $\mathbf{y}_t$ and then optimizes the previous estimate of dictionary $\mathbf{D}^{(t-1)}$ to determine the new estimate $\mathbf{D}^{(t)}$ using stochastic gradient descent.

To learn the phone-specific subspaces, individual dictionaries are learned for each phonetic class separately using the corresponding training data. These phone-specific dictionaries are then used to construct the subspace enhanced posteriors based on the procedure explained in the following section.

4.2. Subspace Enhanced Phone Posteriors

The dictionaries learned for sparse modeling characterize the phonetic subspaces using a large amount of training posteriors. Sparse representation of posterior vectors obtained using those dictionaries enables projection of the posteriors to the space of training data. This projection enhances the posterior vectors by better matching the structure of phonetic subspaces modeled using training data and has been successfully used in speech recognition (Sainath et al., 2011; Dighe et al., 2016b). In this work, we use a similar approach to enhance phone posterior and use it for the task of QbE-STD.

In order to enhance the phone posteriors, we first obtain a sparse representation of a posterior vector using the dictionaries. Then, we use this sparse representation to obtain the enhanced posterior by multiplying it with the corresponding dictionary. The process of obtaining the sparse representation is different for training and test posterior vectors due to fact that, in case of training posteriors, the corresponding phonetic class is known while, the phonetic class is unknown for test posteriors. The methods are discussed in the following.

1. *Enhancement of Training Posteriors*: Given a training posterior vector $\mathbf{x}_t$ from phone class k and a dictionary $\mathbf{D}_k$, we can obtain its sparse representation by solving the following optimization problem (Tibshirani, 1996),

$$\boldsymbol{\alpha}_t = \arg\min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2}\|\mathbf{x}_t - \mathbf{D}_k\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1 \right\} \tag{2}$$

The enhanced posterior vector is computed by multiplying the sparse representation with the corresponding dictionary matrix as $\mathbf{D}_k\boldsymbol{\alpha}_t$. This procedure is used to enhance the posteriors of spoken queries extracted from the training data.

2. *Enhancement of Test Posteriors*: To enhance the testing posteriors, where the underlying phonetic class k is unknown, a global dictionary is constructed by concatenation of the individual phone dictionaries as $\mathfrak{D} = [\mathbf{D}_1 \quad \mathbf{D}_2 \quad \dots \quad \mathbf{D}_K]$. Due to the partitioning in construction of $\mathfrak{D}$, there is an inherent block structure underlying the space of $\mathfrak{D}$. This structure is exploited using group sparse recovery algorithm where block sparsity is encouraged during the optimization (Sprechmann et al., 2011). Sparse representation of a test posterior $\mathbf{x}_t$ is then obtained by solving the following optimization problem,

$$\boldsymbol{\beta}_t = \arg\min_{\boldsymbol{\beta}} \left\{ \frac{1}{2}\|\mathbf{x}_t - \mathfrak{D}\boldsymbol{\beta}\|_2^2 + \lambda f_{\mathfrak{D}}(\boldsymbol{\beta}) \right\} \tag{3}$$

where $f_{\mathfrak{D}}$ is the Group-Lasso regularizer (Sprechmann et al., 2011) defined as: $f_{\mathfrak{D}} = \sum_{i=1}^{K} \|\boldsymbol{\beta}_{\{\mathbf{D}_i\}}\|_2$ and $\boldsymbol{\beta}_{\{\mathbf{D}_i\}}$ indicates the sparse representation coefficients corresponding to a sub-dictionary $\mathbf{D}_i$ inside $\mathfrak{D}$. The function $f_{\mathfrak{D}}$ can be interpreted as a generalization of the ℓ_1 regularization used in (1) and (2), where each atom of the dictionary is considered to be a sub-dictionary. The group sparsity regularizer, $f_{\mathfrak{D}}$ forces the atoms of the dictionary to be activated in groups.

The quantity $\mathfrak{D}\boldsymbol{\beta}_t$ yields a projection of the test posterior onto the phonetic subspace using sparse representation. The posterior vectors thus share common subspaces, which mitigates the effect of unstructured noise (Dighe et al., 2016b). The posteriors of test utterances used as search space for QbE-STD experiments are enhanced using the technique discussed above. A block diagram to obtain the subspace enhanced phone posteriors is depicted in Figure 3. We study the effectiveness of this approach for QbE-STD in Section 8.

5. Phonetic Subspace Representation: A knowledge-based approach

In this section, we describe a linguistic knowledge-based approach to identify the sub-phonetic attributes composing different phonetic units. We further present a setup to exploit this knowledge for extracting new feature vectors (other than phone posteriors) to be used for QbE-STD.

5.1. Phonological Subspaces

The linguistic theory states that the elements of phonological structures can be represented as a vector of sub-phonetic, binary-valued attributes (Chomsky and Halle, 1968; Clements, 1985). Each phone attribute represents the minimal distinction between groups of phonemes that share some set of articulatory, perceptual, and/or acoustic properties. The attributes thus cover both articulator-free and articulator-bound distinctions.

Articulator-free attributes describe the high-level properties that can be used to specify the broad classes of speech sounds. These attributes determine the details of the sub-phonetic variations, including whether a periodic source is used and whether nasal effect is occurred. For instance, the articulator-free attribute [son] distinguishes sonorant sounds ([+son]), produced with a largely open vocal tract (e.g. vowels), from obstruent sounds such as fricatives, produced with a tract constriction.

Articulator-bound attributes describe the articulatory configurations of the vocal tract. For instance, the feature [high] distinguishes between those vowels produced with the tongue close to the roof of the mouth (e.g., the in beat) from those that are not.

The articulator-bound and articulator-free sub-phonetic attributes possess a natural hierarchical structure (McCarthy, 1988). Articulator-free distinctions are more perceptually salient and their acoustic correlates are less context dependent whereas the articulator-bound features make finer distinctions between individual or small groups of phonemes (Jansen and Niyogi, 2013). Every component of the phonological features characterizes a subspace such that the phonemes are formed through the composition of the underlying phonological components (Cernak et al., 2016, 2017). In this paper, we use DNNs to generate a probabilistic representation of the phonological features, as described in the following section.

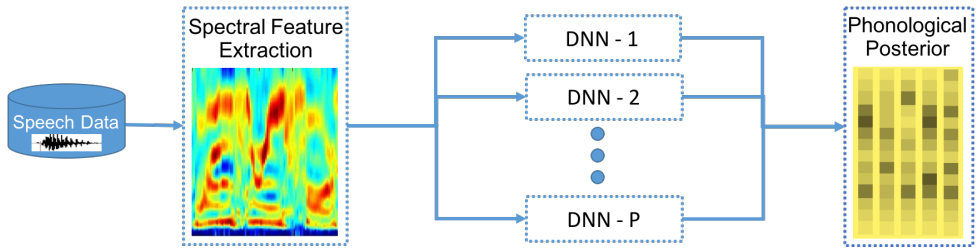


Figure 4: Phonological posterior estimation using a bank of deep neural networks (DNN): First, MFCC based spectral features are extracted from windowed speech. These features are then fed with some acoustic context (left and right) to each DNN modeling a phonological class to estimate the corresponding class conditional posterior probabilities. These probabilities are concatenated to form posterior feature vector.

5.2. Phonological Posteriors

A vector representing the class conditional posterior probabilities of all phonological classes is referred to as phonological posteriors. They are considered to be capable of providing a language independent representation of speech, and any sound can be decomposed into a subset of phonological classes. In contrast, for sparse modeling approach, there is no linguistic knowledge guiding the discovery of the underlying subspaces. Hence, both representations may bear complimentary information on the sub-phonetic structure of the speech representation.

The setup to extract phonological posteriors (Cernak et al., 2016, 2017) from speech signal is similar to the phone posterior estimation as described in Section 3.1. In contrast to the system presented in Figure 1, one DNN is trained per phonological class (see Section 8 for more details). This is due to the fact that multiple phonological classes can be active per speech frame, unlike the case of phone posteriors where only one class is active per frame. Once the DNNs are trained, MFCC features are forward passed through each of them to estimate the corresponding phonological class probability. These probabilities are then concatenated to obtain the phonological posterior vectors. The setup for this process is depicted in Figure 4.

So, we have presented two different approaches to capture the subspace information of speech data in Sections 4 and 5 respectively. Our first approach is aimed at enhancing the phone posteriors using the subspace information from a data-driven method. In a similar manner, we need a method to integrate the information obtained from speech data using phonological posteriors with that of the phone posteriors. Hence, we propose a new approach to fuse information from different feature representations of same speech signal as described in the following section.

6. Distance Fusion

In this section, our goal is to develop a mechanism to integrate information from multiple feature representations of speech data for spoken query detection which we use to fuse the phone and phonological posteriors. Different types of feature vectors have variable dimensions and represent different characteristics of speech, making it difficult to fuse them in the domain of feature vectors. Instead, we use the features independently to construct a distance matrix for DTW and fuse these matrices into a single distance matrix which can be used for query detection using the DTW algorithm explained in Section 3.2.

More formally, let $\mathbf{U}^s = [\,\mathbf{u}_1^s, \mathbf{u}_2^s, \ldots, \mathbf{u}_m^s\,]$ denote the feature vectors extracted from a spoken query and $\mathbf{V}^s = [\,\mathbf{v}_1^s, \mathbf{v}_2^s, \ldots, \mathbf{v}_n^s\,]$ the feature vectors of a test utterance; m and n represents the number of frames in the spoken query and test utterance respectively, and $s = 1, 2, \ldots, S$ indicates the source of feature vector where S is the number of different types of features extracted from the same speech. The pairwise distances of any two feature vectors can be calculated using a distance measure:

$$\begin{aligned} \delta_s\,(i,j) = \text{distance}\left(\mathbf{u}_i^s, \mathbf{v}_j^s\right) \qquad & \forall\ s = 1, 2, \ldots, S \\ & i = 1, 2, \ldots, m \text{ and } j = 1, 2, \ldots, n \end{aligned} \tag{4}$$

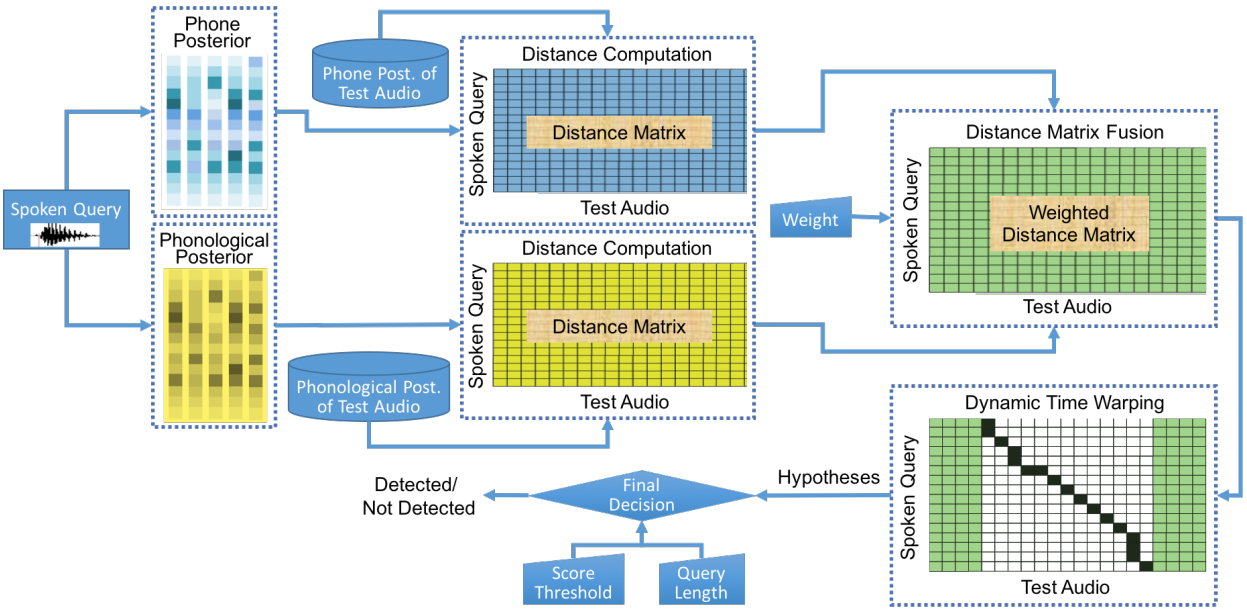


Figure 5: Block diagram of distance fusion for DTW system using phone and phonological posteriors: Both phone and phonological posterior based representations are obtained for spoken query and test utterance to compute corresponding normalized distance matrices. We fuse these distance matrices by taking a weighted combination of each element from corresponding distance matrices. DTW algorithm (Rodriguez-Fuentes et al., 2014) is then used to obtain a hypothesized sub-sequence matching the query. A hypothesis of length less than half the length of the query is discarded to reduce false alarm. Finally, the score of a valid hypothesis is compared to a threshold to yield a decision.

The distance function can be chosen to best match the properties of corresponding feature vectors. In this paper, we consider logarithm of cosine distance as the distance function for its superior performance in posterior-based query detection (Rodriguez-Fuentes et al., 2014). A simple range normalization technique is used to have the distance values between 0 and 1. We fuse the distance matrices (δ_s) for S different sources of feature representations by taking a weighted combination as follows:

$$\Delta(i,j) = \sum_{s=1}^{S} w_s \times \delta_s(i,j) \quad \text{s.t.} \quad \sum_{s=1}^{S} w_s = 1 \\ \forall\, i = 1,2,\ldots,m \text{ and } j = 1,2,\ldots,n \tag{5}$$

where the weight parameters w_s are optimized over an independent development set. The weight associated to a feature representation indicates its importance in conjunction with the rest of feature vectors being fused for a particular task (see Section 8.3 for our experimental analysis on the choice of w_s). After fusion, we simply follow the process described in Section 3.2 to generate QbE-STD hypotheses.

In this work we consider two types of features, namely phone posterior and phonological posterior and show that the fusion technique discussed here improves the detection performance (see Table 4 and Figure 6). The block diagram of the system integrating these features (phone and phonological posteriors) is presented in Figure 5. Note that the distance fusion method is independent of the type of features and distance measure being used, and it is applicable to other DTW based pattern matching.

7. Benchmarking Setup

In this section, we present a brief description of the database and query selection process used for our QbE-STD experiments. Then we discuss the setup used for extracting phone and phonological posteriors and the pre-processing steps involved to perform the experiments. Finally, we describe the evaluation metrics used to compare different systems proposed in this work.

7.1. Database: AMI meeting corpus

The experiments are conducted on AMI meeting corpus (McCowan et al., 2005) using individual headset microphone (IHM) recordings as well as single distant microphone (SDM) recordings with mic-id 1 to evaluate the performance of our system for close talk and far field speech respectively. Both datasets are partitioned into three groups[1] to train the corresponding DNN. The partition consists of about 81 hours of speech for training and about 9 hours for each of the development and evaluation. Although, the meeting language was English, many participants were non-native speakers. In addition, the recordings contain considerable amount of overlapping speech (competing speakers).

7.2. Query Selection

We select different words from the database to construct the query set for our QbE-STD experiments. We use two different strategies to extract queries for experiments on IHM and SDM set. For experiments on IHM, we extract 200 more frequent words (excluding functional words) including very short words such as 'BUY' to long words such as 'TELEVISION'. In case of SDM queries, we compute term frequency-inverse document frequency (TF-IDF) statistic for all words in the dataset, which indicates the importance of a word to a document in a collection of documents. We consider each meeting recording as a document for computing TF-IDF statistic. We arrange these words with decreasing TF-IDF values and choose top 200 words giving us important, content bearing words. The spoken examples corresponding to these words are extracted from randomly chosen utterances in the training data. These queries are divided into two sets of 100 queries each. These sets become our development and evaluation query set. The development queries are used to optimize parameters for different systems. The test set of each dataset is used as the search space for queries from corresponding dataset.

7.3. Phone Posterior Estimation

The setup presented in Section 3.1 is implemented using Kaldi toolkit (Povey et al., 2011) to estimate phone posterior features. We have used Mel frequency cepstral coefficients (MFCC) features as input to the DNN. The MFCC features are calculated from the speech signal over a sliding temporal window of 25ms with a shift of 10ms. These MFCC features together with their 'delta' and 'double delta' coefficients constitute a 39 dimensional vector. The feature vectors along with a context of 4 frames from both left and right (total 9 frames) are fed to the DNN, which gives us 351 dimensional input vector to the DNN. The DNN has 3 hidden layers of 1024 neurons each to estimate 43 dimensional phone posterior probability vector (Ram et al., 2016, 2017, 2018a). Out of these 43 phones, there are 39 non-silence phones obtained from the CMU pronunciation dictionary[2] which is used for lexical modeling. Rest of the 4 phones are considered for representing silence and non-speech sounds in the utterances. To obtain the phone labels for DNN training, we trained a GMM-HMM based speech recognizer. This recognizer is then used to force align (Gales and Young, 2008) the speech frames in an utterance with one of the phones using corresponding transcription. The parameters of the DNN were randomly initialized and were trained by minimizing cross-entropy loss.

7.4. Phonological Posterior Estimation

We have used the open-source phonological vocoding platform presented in (Cernak et al., 2016) to obtain the phonological posteriors. It uses the phonological system of extended Sound Pattern of English (eSPE) for phonological representation (Cernak et al., 2017). It has 21 DNNs corresponding to each phonological class including one class for silence. Each DNN consists of 3 hidden layers of 1024 neurons each. These DNNs were trained using MFCC features with a context of 9 frames giving us a 351 dimensional input vector and 2 output labels indicating whether the phonological class occurs for the segment or not. In other words, each DNN performs binary classification of the target class vs the rest. All the DNNs were randomly initialized and were trained by minimizing cross-entropy loss. The output probabilities from all DNNs are concatenated to form the phonological posterior feature used for QbE-STD.

[1] http://groups.inf.ed.ac.uk/ami/corpus/datasets.shtml
[2] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

7.5. Speech Activity Detection (SAD)

We have implemented a simple speech activity detector following the setup presented in (Rodriguez-Fuentes et al., 2014). This SAD relies on the posterior probabilities to choose silence or noisy frames from speech data. The probability of noise/ silence is obtained by summing up posterior probabilities corresponding to all non-speech phones. If the probability of silence/noise is highest in any frame, it is thrown away from the utterance without performing any kind of smoothing. Moreover, if the length of any utterance is less than 10 frames after SAD, we reject the whole utterance to reduce computational complexity and false alarm rate of the system. Finally, we remove the dimensions corresponding to silence/noise phones from the posterior vector as it is unlikely to help in retrieving utterances containing a given query (Rodriguez-Fuentes et al., 2014). We use phone posteriors to perform SAD for all the experiments with corresponding dataset.

7.6. Evaluation Metric

We use several metrics to evaluate the detection performance of different systems. We consider Maximum Term Weighted Value ($MTWV$) (Rodriguez-Fuentes and Penagarikano, 2013) as our primary metric and use it to optimize the parameters of different systems. $MTWV$ indicates the maximum value of Actual Term Weighted Value ($ATWV$), which is achieved with a well calibrated system. $ATWV$ relies on hard decisions taken by a system by fixing the operating point while obtaining the detection score. It takes into account the prior probability of query occurrence in the test set as well as the costs of missed detection and false alarm. We consider the cost of missed detection (C_{miss}) to be 100 and the cost of false alarm (C_{fa}) to be 1 for our experiments.

We also present minimum normalized cross entropy ($minCnxe$) (Rodriguez-Fuentes and Penagarikano, 2013) for these systems as a secondary evaluation metric. Normalized cross entropy provides the knowledge that a QbE-STD system has on the ground truth. To be more precise, it computes the information which is not provided by the detection scores generated by a given system. $minCnxe$ indicates the minimum normalized cross entropy which can be attained by calibrating a system. A perfect system produces $minCnxe \approx 0$, whereas a non-informative system gives $minCnxe = 1$. Additionally, we present detection error trade-off (DET) curves corresponding to each system to analyze their performance and compare them in a range of false alarm probabilities.

To measure the statistical significance of the improvements in $MTWV$ and $minCnxe$ scores, we perform one-tailed paired-samples t-test. The test is conducted by considering the scores per query ($MTWV$ or $minCnxe$ whichever applicable) as samples and the corresponding p-values are indicated with the results.

8. Experimental Analysis

This section describes different QbE-STD experiments conducted to analyze and evaluate the performance of the proposed approaches exploiting information from various phonetic subspace representations. In all experiments, only one spoken instance of each query is provided, and the test utterances are conversational speech produced by competing speakers.

8.1. Phone and Phonological Posteriors

We have used the QbE-STD system discussed in Section 3 as our baseline system. We use both phone posteriors and phonological posteriors as feature representation for template matching. This work is the first attempt at using phonological posteriors for QbE-STD. The detection performance using development queries of IHM and SDM dataset is presented in Table 1. Clearly, the phonological posteriors perform worse than the phone posteriors for both datasets in a stand-alone system. This can be attributed to the shared sub-phonetic properties of different phonemes. However, we expect that they bear complementary sub-phonetic information guided by the knowledge of linguistics. Hence, we study the performance of the detection system where both features are integrated using the distance fusion technique presented in Section 6.

Table 1: Performance of the DTW based QbE-STD system using phone and phonological posteriors as feature vectors evaluated on the development queries of IHM and SDM dataset.

Feature Representation	IHM		SDM	
	MTWV	minCnxe	MTWV	minCnxe
Phone Posterior	0.4646	0.6558	0.1976	0.8083
Phonological Posterior	0.3105	0.8118	0.0276	0.9118

Table 2: Variation of MTWV and minCnxe for Subspce Enhanced Phone Posterior features with varying regularizarion, λ evaluated on development queries of IHM and SDM dataset

λ	IHM		SDM	
	MTWV	minCnxe	MTWV	minCnxe
0.05	0.4399	0.6827	0.1923	0.8035
0.1	0.4633	0.6712	**0.2497**	**0.7726**
0.2	**0.4705**	**0.6653**	0.2439	0.7910
0.3	0.4563	0.6812	0.2054	0.8094

8.2. Subspace Enhanced Phone Posteriors

We evaluate the data-driven approach to enhance phone posteriors using the subspace structure of speech as discussed in Section 4. The phonetic subspaces are characterized via dictionary learning for sparse representation. We use Algorithm 1 to learn phone-specific subspaces (dictionaries) from corresponding training phone posteriors. For this purpose, we collect all posterior vectors corresponding to a phonetic class, and randomly choose 50 posteriors to initialize the dictionary. Rest of the posteriors are used to train the dictionary.

The posteriors from training data are enhanced using the sparse representation obtained from (2). The enhancement is achieved by multiplying the sparse representation coefficients with the corresponding dictionary. On the other hand, to enhance the test posteriors, all the dictionaries are concatenated to form a single dictionary of all phones. This dictionary is then used to obtain sparse representation of test posteriors using (3). The enhanced posterior is obtained by multiplying the sparse representation with the corresponding dictionary as discussed in Section 4.2.

To evaluate this approach, the regularization parameter λ is tuned on the development set. It controls the number of dictionary atoms (sub-phonetic components) in subspace reconstruction, and its optimal value depends on the size of the dictionary. To evaluate the sensitivity of the QbE-STD system to λ, we compare the $MTWV$ and $minCnxe$ scores corresponding to different values measured on the development queries. The results for IHM and SDM datasets are presented in Table 2. We can see that $\lambda = 0.2$ and $\lambda = 0.1$ yields the best performance for IHM and SDM set respectively, so these values are chosen for the corresponding experiments using evaluation queries.

8.3. Distance Fusion Performance

We analyze the effectiveness of different feature vectors for QbE-STD using the distance fusion technique presented in Section 6. We performed two different experiments for both IHM and SDM datasets to integrate subspace information of speech data presented in the form of phonological posteriors. The first one (Phone + Phonological) integrates phone posteriors with phonological posteriors whereas the second one (Enhanced Phone + Phonological) combines enhanced phone posteriors (data-driven) with phonological posteriors (knowledge-based). In both cases, we construct distance matrices between the query and test utterance using corresponding feature representation. The two matrices are then fused to form a single distance matrix following (5), and is used to perform DTW to detect queries.

Table 3: Variation of MTWV and minCnxe for fusion of two sets of feature representations (Phone+Phonological and Enhanced Phone+Phonological) evaluated on development queries of the IHM set. The optimal weight indicates the contribution of corresponding feature representation to attain the best performance.

Fusion Weights	Phone + Phonological		Enhanced Phone + Phonological	
	MTWV	minCnxe	MTWV	minCnxe
0.4	0.4343	0.6896	0.5197	0.6322
0.5	0.4916	0.6358	**0.5270**	**0.6217**
0.6	**0.4988**	**0.6307**	0.5225	0.6232
0.7	0.4982	0.6323	0.5243	0.6280
0.8	0.4880	0.6359	0.5135	0.6357

In both experiments we merge two matrices, resulting in weights w and $(1-w)$ corresponding to (enhanced) phone and phonological posterior respectively. These weights indicate the significance of the corresponding feature vectors for the detection stage. The results using the development queries of the IHM dataset for different weights are shown in Table 3 as an example. The optimal value of $w = 0.6$ for 'Phone + Phonological' indicates a contribution of 0.6 and 0.4 for phone and phonological posteriors respectively to attain the best performance. In 'Enhanced Phone + Phonological', the optimal value of $w = 0.5$ shows equal contribution of enhanced phone and phonological posteriors for best performance. The optimized weight for SDM set is $w = 0.8$ for both 'Phone + Phonological' and 'Enhanced Phone + Phonological' cases. The higher weight indicate smaller contribution from phonological posteriors compared to IHM set. It can be attributed to the worse performance of the phonological posteriors in a stand alone system for SDM set compared to IHM set as indicated in Table 1. These optimized weights are used for final assessment on evaluation queries as presented in the following section. As part of our experiments, we have also tried a score fusion technique (Brümmer and De Villiers, 2013) as used in (Rodriguez-Fuentes et al., 2014). However, this resulted in worse QbE-STD performance compared to the baseline system using phone posteriors.

8.4. Final Performance

In this section, we present the performance of different systems (optimized using development queries) on evaluation queries. The *MTWV* and *minCnxe* scores are presented in Table 4 for both IHM and SDM datasets, whereas the corresponding DET curves are shown in Figure 6 to indicate the miss probabilities for a given range of false alarm probabilities. The performance using phone posteriors is our baseline system and it gets increasingly better using the subspace enhanced posteriors and distance fusion techniques. It can be observed that the performance of both phone posteriors and enhanced phone posteriors improves while a distance fusion is performed with phonological posteriors. This indicates that, both phone posteriors and enhanced phone posteriors are not able to capture all information present in the speech signal and phonological posteriors are one way of capturing finer details of sub-phonetic components.

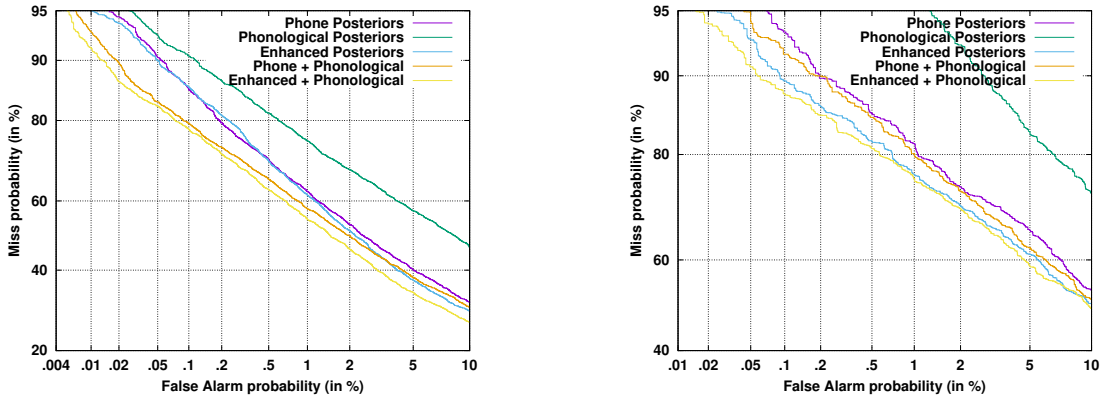
9. Conclusions

QbE-STD benefits from the advanced features capturing the subspace structure of the speech signal. We exploit the low-dimensional subspace structures of speech through a data-driven and a knowledge-based approach. The data-driven approach relies on sparse modeling of phonetic posteriors to characterize the sub-phonetic components in an unsupervised manner and we use these models to enhance the phone posteriors. The knowledge-based approach utilizes linguistic information to identify the sub-phonetic units and represent them using phonological posteriors. To integrate information from multiple representations of speech, a distance fusion technique is proposed. We show that the phone posteriors and phonological posteriors represent complementary information to improve the performance of the QbE-STD system. The QbE-STD solution developed in this paper makes no assumption on the underlying language, thus it is

Table 4: Performance of the DTW based QbE-STD system using different posteriors as feature representations computed on the evaluation queries. Enhanced phone posteriors improve the QbE-STD performance. Also, distance fusion technique is effective in integration of the multiple sources of information through different set of features

Feature Representation	IHM		SDM	
	MTWV	minCnxe	MTWV	minCnxe
Phone Posterior	0.4758	0.6526	0.2319	0.8398
Phonological Posterior	0.3044	0.7780	0.0459	0.8924
Enhanced Phone Posterior	0.5052*	0.6136*	0.2774*	0.8202*
Phone + Phonological Posterior	0.4969*	0.6287*	0.2555*	0.8288*
Enhanced Phone + Phonological Posterior	**0.5414***	**0.6051***	**0.2944***	**0.8102***

* significant at $p < 0.001$



(a) IHM dataset

(b) SDM dataset

Figure 6: DET curves of the DTW based QbE-STD system using different (phone and phonological) posteriors as feature representations computed on evaluation queries for both IHM and SDM dataset. We see that the performance improvement is consistent throughout the given range of false alarm probabilities.

applicable to multilingual scenarios involving low-resource or zero-resource languages. To that end, we plan to use posteriors from DNNs trained on multiple different languages for QbE-STD.

10. Acknowledgments

We acknowledge Dr. Milos Cernak for kind help with the phonological posteriors estimation. We would also like to acknowledge the authors of Rodriguez-Fuentes et al. (2014) for providing their code of DTW. The research leading to these results has received funding from the Swiss NSF project on "Parsimonious Hierarchical Automatic Speech Recognition and Quary Detection (PHASER-QUAD)" grant agreement number 200020-169398.

References

Anguera, X., 2013. Information retrieval-based dynamic time warping. In: Seventeenth Annual Conference of the International Speech Communication Association (INTERSPEECH). pp. 1–5.

Anguera, X., Ferrarons, M., 2013. Memory efficient subsequence DTW for query-by-example spoken term detection. In: 2013 IEEE International Conference on Multimedia and Expo (ICME). IEEE, pp. 1–6.

Anguera, X., Metze, F., Buzo, A., Szoke, I., Rodriguez-Fuentes, L. J., 2013. The Spoken Web Search task. In: the MediaEval 2013 Workshop.

Anguera, X., Rodriguez-Fuentes, L. J., Szoke, I., Buzo, A., Metze, F., Penagarikano, M., 2014. Query-by-example spoken term detection evaluation on low-resource languages. In: The 4th International Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU'14).

Bourlard, H., Morgan, N., 1994. Connectionist speech recognition: A hybrid approach.

Brümmer, N., De Villiers, E., 2013. The bosaris toolkit: Theory, algorithms and code for surviving the new dcf. arXiv preprint arXiv:1304.2865.

Cernak, M., Asaei, A., Bourlard, H., 2016. On structured sparsity of phonological posteriors for linguistic parsing. Speech Communication 84, 36–45.

Cernak, M., Benus, S., Lazaridis, A., 2017. Speech vocoding for laboratory phonology. Computer Speech & Language 42, 100–121.

Chen, H., Leung, C.-C., Xie, L., Ma, B., Li, H., 2016. Unsupervised bottleneck features for low-resource query-by-example spoken term detection. In: INTERSPEECH. pp. 923–927.

Chen, H., Leung, C. C., Xie, L., Ma, B., Li, H., Dec 2017. Multitask feature learning for low-resource query-by-example spoken term detection. IEEE Journal of Selected Topics in Signal Processing 11 (8), 1329–1339.

Chomsky, N., Halle, M., 1968. The Sound Pattern of English. Harper & Row.

Clements, G. N., 1985. The geometry of phonological features. Phonology 2 (01), 225–252.

Deng, L., 2004. Switching dynamic system models for speech articulation and acoustics. In: Mathematical Foundations of Speech and Language Processing. Springer New York, pp. 115–133.

Dighe, P., Asaei, A., Bourlard, H., 2016a. Sparse modeling of neural network posterior probabilities for exemplar-based speech recognition. Speech Communication 76, 230–244.

Dighe, P., Luyet, G., Asaei, A., Bourlard, H., 2016b. Exploiting low-dimensional structures to enhance DNN based acoustic modeling in speech recognition. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 5690–5694.

Elhamifar, E., Vidal, R., 2013. Sparse subspace clustering: Algorithm, theory, and applications. Pattern Analysis and Machine Intelligence, IEEE Transactions on 35 (11), 2765–2781.

Gales, M., Young, S., 2008. The application of hidden markov models in speech recognition. Foundations and trends in signal processing 1 (3), 195–304.

Gemmeke, J. F., Virtanen, T., Hurmalainen, A., 2011. Exemplar-based sparse representations for noise robust automatic speech recognition. Audio, Speech, and Language Processing, IEEE Transactions on 19 (7), 2067–2080.

Hazen, T. J., Shen, W., White, C., 2009. Query-by-example spoken term detection using phonetic posteriorgram templates. In: Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on. IEEE, pp. 421–426.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al., 2012. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. Signal Processing Magazine, IEEE 29 (6), 82–97.

Jansen, A., Niyogi, P., 2013. Intrinsic spectral analysis. IEEE Transactions on Signal Processing 61 (7), 1698–1710.

King, S., Frankel, J., Livescu, K., McDermott, E., Richmond, K., Wester, M., 2007. Speech production knowledge in automatic speech recognition. The Journal of the Acoustical Society of America 121 (2), 723–742.

Lee, C.-H., Siniscalchi, S. M., 2013. An information-extraction approach to speech processing: Analysis, detection, verification, and recognition. Proceedings of the IEEE 101 (5), 1089–1115.

Lee, L.-s., Glass, J., Lee, H.-y., Chan, C.-a., 2015. Spoken content retrieval - beyond cascading speech recognition with text retrieval. IEEE/ACM Transactions on Audio, Speech, and Language Processing 23 (9), 1389–1420.

Mairal, J., Bach, F., Ponce, J., Sapiro, G., 2010. Online learning for matrix factorization and sparse coding. Journal of Machine Learning Research (JMLR) 11, 19–60.

McCarthy, J. J., 1988. Feature geometry and dependency: A review. Phonetica 45 (2-4), 84–108.

McCowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., et al., 2005. The AMI meeting corpus. In: Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research. Vol. 88.

Müller, M., 2007. Dynamic time warping. Information retrieval for music and motion, 69–84.

Park, A. S., Glass, J. R., 2008. Unsupervised pattern discovery in speech. IEEE Transactions on Audio, Speech, and Language Processing 16 (1), 186–197.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., et al., 2011. The Kaldi speech recognition toolkit. In: IEEE 2011 workshop on automatic speech recognition and understanding. IEEE Signal Processing Society.

Rabiner, L. R., Rosenberg, A. E., Levinson, S. E., 1978. Considerations in dynamic time warping algorithms for discrete word recognition. The Journal of the Acoustical Society of America 63 (S1), S79–S79.

Ram, D., Asaei, A., Bourlard, H., 2016. Subspace detection of DNN posterior probabilities via sparse representation for query by example spoken term detection. In: Seventeenth Annual Conference of the International Speech Communication Association (INTERSPEECH).

Ram, D., Asaei, A., Bourlard, H., 2017. Subspace regularized dynamic time warping for spoken query detection. In: Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS).

Ram, D., Asaei, A., Bourlard, H., June 2018a. Sparse subspace modeling for query by example spoken term detection. IEEE/ACM Transactions on Audio, Speech, and Language Processing 26 (6), 1126–1139.

Ram, D., Asaei, A., Dighe, P., Bourlard, H., 2015. Sparse modeling of posterior exemplars for keyword detection. In: Sixteenth Annual Conference of the International Speech Communication Association (INTERSPEECH).

Ram, D., Miculicich, L., Bourlard, H., 2018b. CNN based query by example spoken term detection. In: Nineteenth Annual

Conference of the International Speech Communication Association (INTERSPEECH).
Rish, I., Grabarnik, G., 2014. Sparse modeling: theory, algorithms, and applications. CRC press.
Rodriguez-Fuentes, L. J., Penagarikano, M., 2013. Mediaeval 2013 spoken web search task: system performance measures. n. TR-2013-1, Department of Electricity and Electronics, University of the Basque Country.
Rodriguez-Fuentes, L. J., Varona, A., Penagarikano, M., Bordel, G., Diez, M., 2014. High-performance query-by-example spoken term detection on the SWS 2013 evaluation. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 7819–7823.
Sahraeian, R., Van Compernolle, D., de Wet, F., 2015. Under-resourced speech recognition based on the speech manifold. In: INTERSPEECH. pp. 1255–1259.
Sainath, T. N., Ramabhadran, B., Picheny, M., Nahamoo, D., Kanevsky, D., 2011. Exemplar-based sparse representation features: From TIMIT to LVCSR. Audio, Speech, and Language Processing, IEEE Transactions on 19 (8), 2598–2613.
Sakoe, H., Chiba, S., 1978. Dynamic programming algorithm optimization for spoken word recognition. IEEE transactions on acoustics, speech, and signal processing 26 (1), 43–49.
Sprechmann, P., Ramirez, I., Sapiro, G., Eldar, Y. C., 2011. C-HiLasso: A collaborative hierarchical sparse modeling framework. Signal Processing, IEEE Transactions on 59 (9), 4183–4198.
Szöke, I., Skácel, M., Burget, L., 2014. BUT QUESST 2014 system description. In: MediaEval.
Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological), 267–288.
Wang, H., Lee, T., Leung, C. C., Ma, B., Li, H., May 2013. Using parallel tokenizers with DTW matrix combination for low-resource spoken term detection. In: IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 8545–8549.
Zhang, Y., Glass, J. R., 2009. Unsupervised spoken keyword spotting via segmental DTW on gaussian posteriorgrams. In: Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on. IEEE, pp. 398–403.
Zhang, Y., Salakhutdinov, R., Chang, H.-A., Glass, J., 2012. Resource configurable spoken query detection using deep boltzmann machines. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 5161–5164.