



# Distinctive Feature Based Representation of Speech for Query-by-Example Spoken Term Detection

Abhijeet Saxena, B. Yegnanarayana

Speech and Vision Lab, International Institute of Information Technology, Hyderabad, India

abhijeet.saxena@research.iiit.ac.in, yegna@iiit.ac.in

## Abstract

In this paper, we address the problem of searching spoken queries within spoken databases, which is referred to as query-by-example Spoken Term Detection (QbE STD). A knowledge-based posteriorgram representation of speech is proposed. The knowledge of sound pattern of a language can be captured in terms of binary distinctive features (DFs). This idea is tailored for the needs of an STD system. The proposed representation can be used as a front-end of a template-based QbE STD system. Template-based spoken term detection experiments are conducted on TIMIT database. Segmental dynamic time warping (DTW) is used for template matching. The performance of STD system improves from a mean average precision (MAP) score of 68.38% when using multi-layer perceptron (MLP) posteriorgram, to an MAP score of 75.35% when using proposed DF representation.

**Index Terms:** spoken term detection, posteriorgrams, distinctive features, support vector machines, probabilistic hierarchy

## 1. Introduction

Increase in amount of collected data is accompanied by the need of search, indexing and retrieval methods. This was observed for text data on web, and as a result efficient web search engines are available today. A similar need has now arisen for spoken audio data, because of explosion in the amount of such data. The web search engines of the future may take spoken queries as input to search and browse any spoken database like classroom lectures, broadcast news, audio books, and so on. The problem of searching with spoken queries in a spoken database is referred to as Query-by-Example Spoken Term Detection (QbE STD).

Automatic Speech Recognition (ASR) based approaches to the problem of STD have shown good performance for resource-rich languages and datasets. At the same time, there has been a revival of template matching approaches for ASR, which also triggered their use in STD [1, 2, 3, 4, 5]. Template matching approaches have been widely used for STD in low-resource scenarios.

The template-based STD requires matching of templates. The representation of templates should be such that it can absorb the acoustic variability in multiple examples of the same utterance. The source of variability may be speakers, environment noise, phonetic context and so on. The use of phonetic posteriorgrams for STD in [3] demonstrated their robustness to variability due to speakers and noise.

In this paper, a distinctive feature (DF) based phonetic posteriorgram template representation is proposed. This representation makes explicit use of knowledge of sound pattern of language. The objective of this work is to investigate the effect

of using knowledge-based representation for STD. The performance of DF posteriorgram-based STD is compared to other posteriorgram-based STD systems.

Organization of rest of the paper is as follows. In Section 2, relation to prior work is discussed. In Section 3, an overview of template-based QbE STD system is presented. In Section 4, the proposed approach for STD is described. Experiments conducted to test the proposed ideas and their corresponding results are reported in Section 5. In Section 6, we provide a summary, and discuss the implications of this work.

## 2. Relation to Prior Work

In early days of template matching, spectral feature vectors like mel-frequency cepstral coefficients (MFCCs) were most commonly used to represent speech. To overcome the inability of spectral features in adapting to various speakers or environments, supervised phonetic posteriorgram based template representation has been proposed for speech recognition as well as template matching based STD [2, 6, 3, 4]. Phonetic posteriorgrams were obtained by using multi-layer perceptron (MLP) [2, 6, 3, 4]. The use of MLP is motivated by the fact that it performs non-linear transformation on spectral features. It is able to perform discriminative training and model non-linear classification boundaries [2, 6]. Recently, unsupervised Gaussian mixture models (GMM) based posteriorgrams have been proposed as an alternative to phonetic posteriorgrams in situations where resources are scarce, and supervised training is not possible. In this paper, a knowledge based or semi-supervised approach to obtain posteriorgrams is described, which differs from existing supervised and unsupervised approaches.

Distinctive features are used to obtain knowledge-based templates in this work. Theoretical basis of distinctive features can be found in [7, 8]. Feature system followed in this work is adapted from [9]. A model for lexical access using distinctive features was proposed in [10]. More recently, DFs have been used in ASR as well. Sequence of landmarks were obtained as the peaks in firings of distinctive feature detectors [11]. Similarly, distinctive features were used within an event-based ASR system [12]. Support vector machines (SVMs) were employed to perform binary classification corresponding to each distinctive feature [11, 12]. In present work, an SVM-based strategy is employed. In addition, a new scheme for integration of probabilistic outputs of binary classifiers at each frame is proposed, which differs from the approaches presented in [11, 12]. This is done to adapt to the specific needs of template-based STD.

The idea of using knowledge of speech production in terms of place of articulation and manner of articulation for representation of speech exists in literature in form of articulatory features (AFs). AF-based ASR system, to compensate for speaker variability using a multi-stream architecture, was proposed in

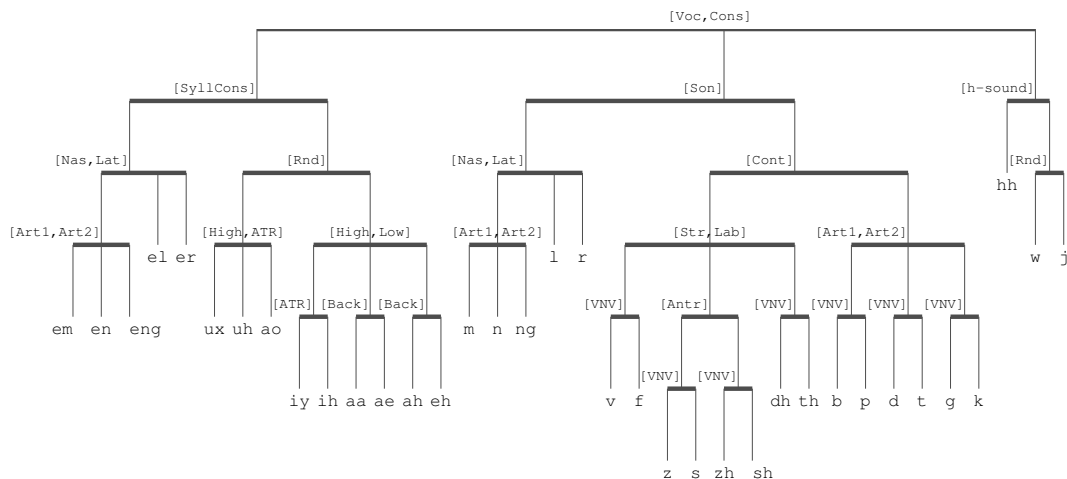


Figure 1: Coding tree illustrating the mapping between phonetic units and their distinctive features (DFs), and also the hierarchical organization of DFs.

[13]. The feature detector for presence or absence of articulatory features was obtained using log-likelihood scores. This differs from the proposed SVM-based binary classifiers. AFs have also been used for STD recently. Broad articulatory phonetic units have been used for indexing audio files in [14]. Articulatory information was used to obtain derivative bottleneck features in [15]. The use of articulatory units for STD was driven by their ability to model speech-specific features across languages. The aim of proposed approach is to capture the sound pattern of a language to obtain a robust representation, which differs from the approaches given in [14, 15].

### 3. Overview of template-based STD system

There are three main stages in all template-based spoken term detection systems [16]. They include query template selection, template representation and template matching. In the initial stage of query selection, best examples of the query term are selected from a choice of multiple examples. In the next stage, representation of both query terms and test utterances in the search database are obtained. This is followed by a final stage of template matching, which assigns a similarity score between two templates.

The overall search algorithm selects examples of query term, obtains template representation of query terms and test utterances in search database and then searches by sliding the query template over the test templates in the search database. A match between query template and a window of test template is performed by using dynamic time warping (DTW) algorithm to obtain a matching score. A list of test utterances, ranked by their scores, is returned as the output for each query term. Clearly, such a sliding window approach to search is time-consuming. It is not applicable to search in large databases that may have hundreds of hours of spoken data. Thus, template based approaches may be used as second pass after some initial fast indexing mechanism, like bag of acoustic words model discussed in [17, 18].

In this work, query selection is not considered. It is assumed that only one example of the query term is available. Furthermore, in template matching stage, segmental dynamic

time warping [5], which is a variant of dynamic time warping (DTW) algorithm [19], is adapted for matching templates of varying lengths. The focus of this work is on the template representation stage. In the following subsection, details of this stage are described briefly.

#### 3.1. Template Representation

The proposed template representation belongs to a class of posteriorgram-based representations. Specifically, it is a phonetic posteriorgram. Phonetic posteriorgram is defined as a sequence of posterior feature vectors. Each element of the vector represents the probability of a phonetic class, given acoustic observations. It can be expressed as follows:

Consider a speech utterance represented by a sequence of acoustic observation vectors  $O = o_1, o_2, \dots, o_T$ , where  $T$  is the number of spectral frames in the utterance. The posteriorgram representation  $R$ , corresponding to  $O$ , is given by a  $M \times T$  matrix, where  $M$  is the total number of phonetic classes and each element of  $R$  is a probability term given by the following expression.

$$R_{ij}(O) = Pr(C_i | o_j), \forall i = 1, 2, \dots, M; \forall j = 1, 2, \dots, T \quad (1)$$

where  $C_i$  is  $i^{th}$  phonetic class, and  $o_j$  is  $j^{th}$  spectral frame.

In next section, we present a knowledge-based approach to obtain template representation in terms of phonetic posteriorgrams.

### 4. Distinctive feature based phonetic posteriorgram

There are three key ideas that act as the basis of this work. They include representation of speech in terms of distinctive features, internal organization of these features as a hierarchy, and acoustic correlates corresponding to each feature. These ideas are explored to obtain a posteriorgram representation of speech.

Speech is viewed as a sequence of phones, and each phone is described as a bundle of binary-valued distinctive features. The mapping between a phone and features is such that changing the binary value of even one feature can potentially result in

realizing a different phone. For example, bilabial stop sounds are identified as either /b/ or /p/ based on presence or absence of a single binary feature [voicing]. High vowels /i/ and /u/ are discriminated on the basis of presence or absence of feature [back]. Thus, each phonetic class has a unique description in terms of DFs. The inventory of DFs and description of phonetic units in terms of them is adapted from [9]. A few exceptions are as follows. A single feature [VNV] is used for voiced versus non-voiced classification. A pair of features, [Art1,Art2], is used to describe presence of lips, tongue blade or tongue body. [SyllCons] is used to classify between syllabic consonants and vowels and lastly, [h-sound] separates the sound /h/ from glides /w/, /j/.

The posterior probability for phonetic class  $C_i$  given spectral feature vector  $o_t$ , as on right hand side of equation (1), can be written as:

$$Pr(C_i|o_t) = Pr(f_1 f_2 \dots f_K | o_t) \quad (2)$$

i.e., the posterior probability of a phonetic class is expressed in terms of the posterior probability of a joint event  $f_1 f_2 \dots f_K$ , where the terms  $f_k$  denote events corresponding to DFs. By application of chain rule, the probability for this joint event can be broken into product of probabilities for conditional events. In the following paragraph, the procedure to identify conditional events corresponding to each joint event is described.

Distinctive features are related to each other in a hierarchical manner [12]. Coding of phonetic units in terms of binary features and the hierarchical organization of these features are illustrated in Figure 1. Nodes of this hierarchical structure correspond to features, and leaf nodes are the phonetic classes. The probability of a phonetic unit is equal to the probability of the joint event of DFs, as expressed in equation (2), which is equal to the product of probabilities of conditional events,  $f_{ck}$ , that lie on the path of that phonetic unit in the tree.

The hierarchical structure is characterized by two types of nodes. They are: (a) Single feature, two classes (for example: [anterior] for /s/, /z/ versus /sh/, /zh/) and, (b) Two features, three classes (for example: [vocalic, consonantal] for syllabics versus consonants versus glides). Nodes of type (b) exists because there is no hierarchical relationship between the two features at these nodes. Thus, such features have to exist at the same hierarchical level.

With two binary-valued features, four possible classes can be defined. All nodes belonging to type (b) possess a consistent property. At these nodes, one of the four possible classes is non-existent. Thus, there are only three classes and one null class  $\phi$ , as shown in Table 1. Probability of each class is defined as the product of corresponding probabilities of distinctive features, denoted by  $p_a$  and  $p_b$  in Table 1. Probabilities of observing +1 class for these two features are denoted by  $p$  and  $q$ , respectively. The fourth class ( $\phi$ ) has a non-zero probability, and thus the sum of probabilities for three classes is not unity. In order to realize posterior vector as a probability distribution, the non-zero probability of  $\phi$  is distributed among the three classes by adding a bias term to obtain modified probability, denoted by  $p_m$  in Table 1. The variables  $w_1, w_2, w_3$  control the fraction of probability to be distributed to each class. The complete idea of integration of posterior probabilities of conditional events to get posterior probabilities of phonetic units is expressed as follows:

$$Pr(C_i|o_t) = \prod_{k=\gamma_i} Pr(f_{ck}|o_t), \quad (3)$$

where,  $\gamma_i$  is the path corresponding to class  $C_i$  in the hierarchical tree in Figure 1.

Table 1: Scheme for probability redistribution at type (b) nodes of coding tree.

Class ( $C_i$ )	$p_a$	$p_b$	$p_i = p_a p_b$	bias ( $b_i$ )	$p_m = p_i + b_i$
$C_1$	p	1-q	$p_1$	$w_1 p_4$	$p_1 + w_1 p_4$
$C_2$	1-p	q	$p_2$	$w_2 p_4$	$p_2 + w_2 p_4$
$C_3$	1-p	1-q	$p_3$	$w_3 p_4$	$p_3 + w_3 p_4$
$C_4(\phi)$	p	q	$p_4$	$-p_4$	<b>0</b>

The terms corresponding to posterior probabilities of conditional events  $f_{ck}$  in equation (3) are computed using support vector machines (SVMs). The phones belonging to +1 / -1 class of each classifier are identified using the hierarchical structure of Figure 1. In training a SVM, only examples of phones relevant to that SVM are used.

## 5. Experiments and Results

Various experiments are conducted to test the ideas proposed in previous section. A template-based spoken term detection system is implemented using DF posteriorgram templates. Other representations including raw spectral features (MFCCs), Gaussian posteriorgrams [5], MLP phonetic posteriorgrams [2, 4], are also used for comparison with proposed representation. Experiments are described in the following subsections.

### 5.1. SVM training

Support Vector Machines are employed to estimate the probabilities of distinctive features at each node of the coding tree illustrated in Figure 1. Positive and negative examples for each SVM are the phones that belong to its corresponding class in the coding tree. For example, at node [Antr], positive and negative examples of SVM are sounds belonging to /s/, /z/ and /sh/, /zh/, respectively. At node [High, Low], the SVM corresponding to feature [High] classify between /iy/, /ih/ and /aa/, /ae/, /ah/, /eh/, while the SVM corresponding to [Low] classify between /aa/, /ae/ and /iy/, /ih/, /ah/, /eh/. This strategy is based on SVM training presented in [12]. In this manner, a total of 35 SVMs are identified. The 61 labels in TIMIT database were merged into 36 distinct phonetic classes ( $M = 36$ ). The sounds labeled as diphthongs and affricates in TIMIT were not used for training.

The SVMs are trained on TIMIT training set (*sx* and *si* sentences only). It consists of 3696 files, each of duration between 2 to 4 seconds, thus adding up to roughly 3 hours of data. The same database is used to train a multi-layer perceptron with back propagation learning. The number of neurons in the hidden layer is set at 1000.

Speech signals are processed at frame rate of 10 ms and frame size of 20 ms. 13-dimensional Mel frequency cepstral coefficients (MFCCs) are extracted for each frame. Furthermore, for each frame, vectors of current frame, 4 prefix frames and 4 suffix frames are concatenated to obtain 117-dimensional vectors. These raw spectral vectors are used to train SVM and MLP classifiers. The library LIBSVM [20] is used to implement binary classifiers. Radial basis function (RBF) kernels are used in training of all the SVMs. The optimal parameters  $C$  and  $\gamma$  for each SVM are obtained using grid search.

### 5.2. Template generation: Integration of SVM outputs

The probability redistribution weights,  $w_1, w_2, w_3$ , as shown in Table 1, are the crucial parameters for integration of SVM

Table 2: Performance evaluation of QbE STD system based on various template representations for TIMIT database.

Template	P@N (%)	MAP (%)	EER (%)
MFCC	40.41	43.28	18.95
GMM	41.99	44.52	16.56
MLP	62.60	68.38	7.94
SVM-random	69.30	74.91	5.60
SVM-equiprobable	69.76	75.29	5.34
SVM-priors	69.88	75.36	5.32

outputs. Experiments are conducted for three different choices of these weight parameters. In the first case, weights are not used at all. Instead, posterior probability vectors are normalized to sum to unity. This can be interpreted as random redistribution of probability. In the second case, classes are assumed as equiprobable and weights are set as 0.33. In the third case, prior probability of each class is used to obtain weight parameters. Prior probabilities are precomputed by counting relative frequency of data vectors for each class.

### 5.3. SVM template based QbE STD

The posteriorgram templates, obtained by operations discussed above, are used to perform template-based spoken term detection. The skip parameter of segmental DTW algorithm is set at 6 frames, as in [5]. Search complexity of segmental DTW algorithm is  $O(MN^2)$ , where  $M$  and  $N$  are number of frames in test and query templates, respectively. The posterior vectors can be interpreted as the probability distribution of phone classes. This interpretation allows us to apply Kullback-Leibler distance as a measure of local distance between two posterior vectors within DTW matching.

Experiments are performed on the test set of TIMIT database (*sx* and *si* sentences only). This entire set consisting of 1344 utterances (approx. 1.2 hours) is used as the search database. The size of search database does not affect the outcome of experiments as long as the same database is used for comparison across various templates. A set of 50 queries is constructed for these experiments. These query terms are excised from the TIMIT test set. Queries of varied syllable length (2 to 5) are selected. A quick term frequency inverse document frequency (tf-idf) analysis on sentences in TIMIT test set shows that terms other than stop words like “the”, “and”, “from” have almost same frequency. Query terms occurring more than twice are rare (eg. : “between”). The frequency of occurrence of selected query terms in the search database vary from 7 to 24. Examples of each query term carry information about speakers, but variability due to phonetic context is rarely present. One of the objectives of these experiments is to project the significance of proposed posteriorgram representation for robustness to variability due to speakers.

The metrics used to evaluate performance of STD systems are precision at top N hits (P@N) (where N is the total number of occurrences of query terms in search database), mean average precision (MAP) and equal error rate (EER). Results are listed in Table 2. MFCC and GMM representations do not rely on either supervised training or use of knowledge. As a result, they show relatively poor performance. In posteriorgram based QbE STD, MLP phone posteriorgram have reported the best performance. Thus, supervised MLP posteriorgram is the baseline representation for our experiments. All the three knowledge-based SVM posteriorgram perform better than the

baseline. Among the three representations, an improvement across all metrics is observed. This implies that the scheme for integration of SVM outputs is crucial to the performance of the system. The scheme that takes into account the prior distribution of classes for probability redistribution gives the best results. The worst performing queries are common for MLP and SVM-priors representations. They include *living*, *without*, *wardrobe*. But even for these terms, SVM-priors show better performance than MLP representation. Thus we can conclude that use of knowledge for representation of speech can lead to more accurate detection of queries.

## 6. Summary and Conclusions

In this paper, distinctive feature based posteriorgram are used as templates for the task of QbE STD. Distinctive features can be viewed as an intermediate layer between phonetic units and acoustic observations. SVMs are employed to perform binary classification corresponding to each feature given acoustic observation. A new scheme for integration of SVM outputs to obtain posterior vectors of phonetic classes is proposed. The major contribution of this work lies in conception of a mechanism for integration of classifier outputs. The proposed approach is flexible in the sense that binary classification corresponding to any feature is independent of all other features. Various schemes for redistribution of extra probability were tested. The binary nature of features and their hierarchical organization leads to a representation where knowledge based features act as an intermediate layer between acoustic observations from the speech signal and discrete units of representation in a language. This is in contrast to existing posteriorgram representations where acoustic observations are directly mapped on to sub-word units. The performance of proposed representation was compared with that of existing posteriorgram and spectral representations in the context of template of STD. DF posteriorgram representation performs better than the baseline MLP posteriorgram representation in the context of STD. Experiments on probability redistribution parameters highlight the importance of prior class distribution in integration.

Based on the results presented above for TIMIT dataset, it can be established that use of knowledge of sound pattern of a language can aid in improving the performance of a STD system. The next logical step in evolution of these ideas is to address the challenges of scalability and adaptability encountered while dealing with either large datasets for resource-rich languages or low resource languages. For low-resource setting, this can be achieved by evolving an unsupervised method, that would replace the supervised training of proposed method, to obtain knowledge-based DF representation of speech.

## 7. References

- [1] Petr Fousek and Hynek Hermansky, “Towards ASR based on hierarchical posterior-based keyword recognition,” in *proc. ICASSP*, 2006, pp. 433–436.
- [2] Guillermo Aradilla, Jithendra Vepa, and Hervé Bourlard, “Using posterior-based features in template matching for speech recognition.,” in *proc. INTERSPEECH*, 2006.
- [3] Timothy J Hazen, Wade Shen, and Christopher White, “Query-by-example spoken term detection using phonetic posteriorgram templates,” in *proc. of the IEEE Automatic Speech Recognition & Understanding*, 2009, pp. 421–426.

- [4] Vikram Gupta, Jitendra Ajmera, Arun Kumar, and Ashish Verma, "A language independent approach to audio search," in *proc. INTERSPEECH*, 2011, pp. 1125–1128.
- [5] Yaodong Zhang and James R Glass, "Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams," in *proc. of the IEEE Automatic Speech Recognition & Understanding*, 2009, pp. 398–403.
- [6] Guillermo Aradilla, Hervé Bourlard, et al., "Posterior features applied to speech recognition tasks with user-defined vocabulary," in *proc. ICASSP*, 2009, pp. 3809–3812.
- [7] Roman Jakobson, C Gunnar M Fant, and Morris Halle, *Preliminaries to speech analysis*, MIT Press, 1951.
- [8] Noam Chomsky and Morris Halle, *The sound pattern of English.*, Harper and Row Publishers, 1968.
- [9] Kenneth N. Stevens, *Acoustic Phonetics*, MIT Press, Cambridge, MA, 1998.
- [10] Kenneth N Stevens, "Toward a model for lexical access based on acoustic landmarks and distinctive features," *The Journal of the Acoustical Society of America*, vol. 111, no. 4, pp. 1872–1891, 2002.
- [11] Aren Jansen and Partha Niyogi, "Modeling the temporal dynamics of distinctive feature landmark detectors for speech recognition," *The Journal of the Acoustical Society of America*, vol. 124, no. 3, pp. 1739–1758, 2008.
- [12] Amit Juneja and Carol Espy-Wilson, "A probabilistic framework for landmark detection based on phonetic features for automatic speech recognition," *The Journal of the Acoustical Society of America*, vol. 123, no. 2, pp. 1154–1168, 2008.
- [13] Florian Metze and Alex Waibel, "A flexible stream architecture for ASR using articulatory features," in *Proceedings of the International Conference on Spoken Language Processing*, Denver, CO, USA, 2002.
- [14] Gautam Varma Mantena, Bajibabu Bollepalli, and Kishore Prahallad, "SWS task: Articulatory phonetic units and sliding DTW," in *proc. MediaEval*, 2011.
- [15] Gautam Mantena and Kishore Prahallad, "Use of articulatory bottle-neck features for query-by-example spoken term detection in low resource scenarios," in *proc. ICASSP*, 2014, pp. 7128–7132.
- [16] Anupam Mandal, K. R. Prasanna Kumar, and Pabitra Mitra, "Recent developments in spoken term detection: A survey," *Int. J. Speech Technol.*, vol. 17, no. 2, pp. 183–198, June 2014.
- [17] Basil George, Abhijeet Saxena, Gautam Mantena, Kishore Prahallad, and B Yegnanarayana, "Unsupervised query-by-example spoken term detection using bag of acoustic words and non-segmental dynamic time warping," in *proc. INTERSPEECH*, 2014.
- [18] Basil George and B Yegnanarayana, "Unsupervised query-by-example spoken term detection using segment-based bag of acoustic words," in *proc. ICASSP*, 2014, pp. 7133–7137.
- [19] Hiroaki Sakoe and Seibi Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 1, pp. 43–49, 1978.
- [20] Chih-Chung Chang and Chih-Jen Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, pp. 27, 2011.