

# Vowel Detection and Classification using Support Vector Machines (SVM)

Malihe Gheidi<sup>\*</sup>, Abolghasem Sayadian<sup>\*\*</sup>

*\*Electrical Engineering Department, Amirkabir University of Technology, Tehran, Iran*  
marjan\_gh1@yahoo.com

*\*\*Electrical Engineering Department, Amirkabir University of Technology, Tehran, Iran*  
eeas55@aut.ac.ir

**Abstract:** This paper presents our work on vowel detection and classification as part of a project of Persian continuous speech recognition based on demi-syllable units. Recent work in machine learning has focused on models, such as the support vector machine (SVM), that automatically control the generalization and discrimination as part of overall optimization process. In this research we track all vowel phonemes in speech signal and in order to achieve minimum vowel losses and accurate detection, we focus on taking special care of vowel detector and classifier using support vector machines and acoustic data of speech signal. At first this technique has been tested with Persian isolated words database and good results have been achieved from this algorithm Experiments with this database result in 1.62 % of total detection and classification errors. The insertion error is 1.06 %, the deletion error is 0.56 % and the classification error is 0 %

**Keywords:** Acoustic data, demi-syllable units, Persian, speech recognition, support vector machines.

## INTRODUCTION

To build a speech recognition system, a basic decision that has to be made is the choice between the different speech units that the system has to be based on. While the best choice for a small vocabulary speech recognition system would be the use of word models, when we deal with medium and large vocabulary systems the use of subword units like phoneme, syllable and demi-syllable to model the speech abstractly becomes a necessity. The basic advantages of using such subword units are the following: (a) phonemes are linguistically well defined; therefore pronunciation models based on phonemes for virtually any word can be looked up easily from a dictionary; (b) the number of the phoneme units describing a language is rather small, for example in the Persian language no more than 30 phoneme units are needed. On the other hand, phonemes are context dependent (CD) and their models do not incorporate co-articulation.

Syllable units have good information of speech, but one disadvantage of syllables as subword is the large number of them. There are about 7000 syllable in Persian language.

Because of small number of demi-syllable and good cover of different cases, they are good choice as

subword units. There are about 700 demi-syllables in Persian language.

As the spectral and temporal characteristics of vowels are more stable than other phonemes, they are important in speech recognition. So in this paper as part of a project of Persian continuous speech recognition based on demi-syllable units, vowel detection and classification is presented.

The next important issue in designing a speech recognition system is how to model the speech signal. Hidden Markov models (HMMs) is the most commonly used method. The power of an HMM representation lies in its ability to model the temporal evolution of a signal via an underlying Markov process. Widespread use of HMMs for modeling speech can be attributed to the availability of efficient parameter estimation procedures [1], [2] that involve maximizing the likelihood (ML) of the data given the model. One of the most compelling reasons for the success of ML and HMMs has been the existence of iterative methods to estimate the parameters that guarantee convergence. The expectation-maximization (EM) algorithm provides an iterative framework for ML estimation with good convergence properties, although it does not guarantee finding the global maximum [3].

The HMM are representative models, but

discriminative approaches are a key ingredient for creating robust and more accurate models.

Another important issue in speech recognition is generalization properties of system. HMM-based speech recognition systems perform very well on closed-loop tests but performance degrades significantly on open-loop tests [4]. The performance of systems on speaker-dependent tasks is significantly better than on speaker-independent tasks. This can be attributed to the fact that most systems do not generalize well [5], [6].

The need for discrimination and classifiers with good generalization and convergence properties that can be used for speech recognition has led us to look at a new machine learning paradigm called the support vector machines (SVM).

The remainder of this paper is organized as follows. In Section 2 we describe the theory of support vector machines. We discuss the algorithm description which consists: vowel detection stage and vowel classification stage in Section 3. Section 4 represents the experimental results and Section 5 summarizes the main contributions of the paper.

### 1. Support Vector Machines

The foundations of Support Vector Machines (SVM) are based on Structural Risk Minimization (SRM) principle, which minimizes an upper bound on the generalization error, as opposed to Empirical Risk Minimization (ERM) which minimizes the error on the training data.

Support Vector Machines (SVM) is a statistical algorithm with a great potential to generalize, that can successfully be used in pattern recognition and information retrieval tasks. The main idea in training a SVM system is finding a hyperplane as a decision boundary between two classes [7].

#### 1.1. Linear Support Vector Machines

Consider the problem of separating the set of  $N$  training vectors  $\{x_1, x_2, \dots, x_i, \dots, x_N\}$  belonging to two different classes  $y_i \in \{+1, -1\}$ . The goal is to find the linear decision function  $f(x)$  and the separating hyperplane  $H$ .

$$H : w \cdot x + b = 0$$

$$f(x) = \text{sgn}(w \cdot x + b)$$

Where  $b$  is the distance of the hyperplane from the origin and  $w$  is the normal to the decision region. Let the “margin” of the SVM be defined as the distance from the separating hyperplane to the closest two classes. The SVM training paradigm finds the separating hyperplane which gives the maximum margin. The margin is equal to  $2/\|w\|$ . Once the hyperplane is obtained, all the training examples satisfy the following

inequalities.

$$x_i \cdot w + b \geq +1 \quad \text{for } y_i = +1$$

$$x_i \cdot w + b \leq -1 \quad \text{for } y_i = -1$$

We can summarize the above procedure to the following:

$$\text{Minimize } \frac{1}{2} w \cdot w$$

$$\text{Subject to } y_i (w \cdot x_i + b) \geq +1, \quad i = 1, 2, \dots, N$$

This is a quadratic optimization problem [6].

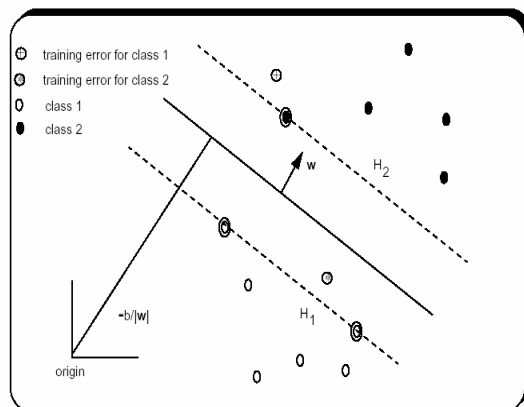


Figure 1: an example of SVM classifier

#### 1.2. Non-linear case

Real-world classification problems typically involve data that can only be separated using a nonlinear decision surface. Optimization on the input data in this case involves the use of a kernel-based transformation [8]:

$$k(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$$

Kernels allow a dot product to be computed in a higher dimensional space without explicitly mapping the data into these spaces.

Two commonly used kernels are:

$$k(x, y) = \exp\{-\gamma|x - y|^2\} \quad \text{Radial Basis Function (RBF)}$$

$$k(x, y) = (x \cdot y + 1)^d \quad \text{Polynomial}$$

#### 1.3. Multi-class classifiers

So far we have only discussed using SVMs to solve two-class problems. However, if we are interested in conducting vowel classification experiments, we will need to choose between multiple classes. The best method of extending the two-class classifiers to multi-class problem is not clear. Previous work has generally constructed a “one vs. all” classifier for each class, or constructed a “one vs. one” classifier for each pair of classes. The “one vs. all” approach works by constructing for each class a

classifier which separates that class from the remainder of data. The “one vs. one” approach simply constructs for each pair of classes a classifier which separates those classes. A test example is then classified by all of the classifiers, and is said to belong to the class with the largest number of positive outputs from these sub-classifiers.

## 2. Algorithm Description

The algorithm consists of two basic stages: the vowel detection stage and the vowel classification stage.

### 2.1. Vowel Detection Stage

In this stage the location and duration of vowels are obtained. The procedure in this stage can be presented by the following three steps:

*Step 1: Band-Pass energy contour estimation.* In this step for the primary detection of vowels, the acoustic data of speech signal has been used. The band pass energy function is calculated. The energy function is then smoothed using five-point Hanning window.

*Step 2: Primary vowel candidate location.* From the smoothed energy function, the extremes (peaks and dips) are located by applying a usual peak-peaking procedure. The frames with energy value below a heuristic threshold are rejected. the reminder of frames candidate as a vowel segment.

*Step 3: Final vowel candidate location.* In this step the SVM classifier is trained as two class vowel vs. non-vowel problem and then is applied to the frames of vowel segments that candidate from the pervious step. Finally the frames belong to the vowel class is accepted as final vowel segment.

The insertion and deletion errors are calculated in this step.

### 2.2. Vowel Classification Stage

In this stage the detected vowel segments from pervious stage, are classified to seven different types of vowel.

In this case the SVM classifiers are trained as multi-class problem. A fundamental issue in classifier design is whether the classifiers should be one-versus-one classifiers, which learn to discriminate one class from another class, or one-versus-all, which learn to discriminate one class from all other classes. One-versus-one classifiers are typically smaller and less complex and can be estimated using fewer resources than one-versus-all classifiers. When the number of classes is  $N$ , we need to estimate  $N(N-1)/2$  one-versus-one classifiers as compared to one-versus-all classifiers. On several standard classification tasks, it has been proven that one-versus-one classifiers are marginally more accurate than one-versus-all classifiers [10], [11]. Nevertheless, for computational efficiency, we chose to use one-versus-all classifiers in the experiments reported here.

We use SVM with RBF kernel. The classification error is calculated in this stage. The total error from two stages is calculated as following:

$$\text{Total error} = \text{Insertion error} + \text{Deletion error} + \text{Classification error}$$

## 3. Experimental Results

In these experiments the window length is 20 ms and the frame shift is 8 ms. 12 MFCC coefficients have been used as feature vector.

We use 80 % of database as a training set and remaining 20 % of database as testing set. We repeat this work five times as all of data in database has been used for testing one time. Table 1 represents the average of the errors in five experiments.

All of data in this database have been manually labeled.

Table 1: The average of the errors in five repeated experiments

Persian vowel	Deletion error (%)	Insertion error (%)	Classification error (%)	Total error (%)
a	0.26	0.51	0	0.77
@	0.77	1.03	0	1.80
e	0.48	1.19	0	1.67
i	1.60	1.60	0	3.20
o	0	0.43	0	0.43
u	0.80	2.40	0	3.60
w	0	0.26	0	0.26
Mean error	0.56	1.06	0	1.62

## 4. Conclusion and Future Works

In this paper we present our work on vowel detection and classification using support vector machines and acoustic data of speech signal. The good results have been achieved from this algorithm.

We can use this algorithm for continuous speech recognition with extension to the work.

## ACKNOWLEDGMENT

The authors would like to thank Iran Telecommunication Research Center for financial supporting of this project.

## REFERENCES

- [1] F. Jelinek, *Continuous speech recognition by statistical*

- methods, Proc. IEEE*, vol. 64, pp. 532–537, 1976.
- [2] L. R. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
- [3] A. P. Dempster, N. M. Laird, and D. B. Rubin, *Maximum likelihood estimation from incomplete data*, *J. R. Statist. Soc.*, vol. 39, no. 1, pp. 1–38, 1977.
- [4] J. Fritsch, *Hierarchical Connectionist Acoustic Modeling for Domain-Adaptive Large Vocabulary Speech Recognition*, Ph. D. dissertation, University of Karlsruhe, Germany, 2000.
- [5] A. Ganapathiraju, J. Hamaker, and Picone, *Applications of support vector machines to speech recognition*, *Signal Processing*, IEEE Transaction, 2004.
- [6] A. Ganapathiraju, *Support Vector Machines For Speech Recognition*, Ph.D. dissertation, Mississippi State University, Mississippi State, MS, 2002.
- [7] C.J.C. Burges, *A tutorial on support vector machines for pattern recognition*, *Knowledge Discovery Data Mining*, vol. 2, no. 2, pp. 121–167, 1998.
- [8] V. N. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.
- [9] I. Gavati, G. Costache, and Iancu, *Robust Speech Recognizer using Multi-class SVM*, IEEE 2004.
- [10] E. Allwein, R. E. Schapire, and Y. Singer, *Reducing multiclass to binary: A unifying approach for margin classifiers*, *J. Machine Learning Res.*, vol. 1, pp. 113–141, Dec. 2000.
- [11] J. Weston and C. Watkins, *Support vector machines for multi-class pattern recognition*, in *Proc. Seventh Eur. Symp. Artificial Neural Networks*, Bruges, Belgium, 1999, pp. 219–224.