KEYWORD SPOTTING USING SUPPORT VECTOR MACHINES

Yassine Ben Ayed[1], Dominique Fohr[1], Jean Paul Haton[1], and Gérard Chollet[2]

[1] LORIA-CNRS/ INRIA Lorraine, BP239, F54506, Vandoeuvre, France
{ybenayed, fohr, jph}@loria.fr
[2] ENST, CNRS-LTCI, 46 rue Barrault, F75634 Paris cedex 13, France
{chollet}@tsi.enst.fr

Abstract. Support Vector Machines is a new and promising technique in statistical learning theory. Recently, this technique produced very interesting results in pattern recognition [1][2][3].
In this paper, one of the first application of Support Vector Machines (SVM) technique for the problem of keyword spotting is presented. It classifies the correct and the incorrect keywords by using linear and Radial Basis Function kernels. This is a first work proposed to use SVM in keyword spotting, in order to improve recognition and rejection accuracy. The obtained results are very promising.

1 Introduction

Automatic speech recognition systems need to be more flexible to accept a wide range of user responses and behaviours. Unaware of technology limitations, users rightfully expect the system to work properly, even if their response includes disfluencies such as hesitation, extraneous speech and false starts. Users may also respond with utterances that do not include any of the recognizer keywords, or that include the expected keywords in a phrase or a sentence.

A recognizer must thus be able, in first time to spot a keyword embedded in speech, in second time to reject speech that does not include any valid keyword. However, word-spotting and word rejection are interrelated such that a good word-spotting capability necessarily implies a good rejection performance.

In this paper, a support vector machine based method is proposed for keyword spotting. The SVM minimizes the structural risk, i.e., the probability of misclassifying patterns for fixed but unknown probability distribution of the data. This is in contrast to traditional pattern recognition techniques of minimizing the empirical risk, i.e., of optimising the performance on the training data. This minimum structural risk principal is equivalent to minimizing an upper bound on the generalisation error [4].

The organisation of this paper is as follows : Section 2 gives a brief description of the basic principles of the SVM. The details concerning database and recognition system are given in section 3. In section 4, we present the way we use SVM for keyword spotting. Experimental results are described in section 5, and conclusions are given in section 6.

2 Support Vector Machines (SVM)

Support vector machines have been recently introduced as a new technique for solving pattern recognition problems [5][6]. They perform pattern recognition between two point classes by finding a decision, determined by certain points of the training set, termed support vectors. This surface, which in some feature space of possibility infinite dimension can be seen as a hyperplane. It is obtained from the solution of the problem of quadratic programming that depends on regularization parameter.

2.1 The linear separable case

Consider the problem of separating the set of training vectors belonging to two separated classes,

$$D = \{(x_1, y_1), ..., (x_m, y_m)\},$$

where $x_i \in R^n$ is a feature vector and $y_i \in \{-1, 1\}$ a class label, with a hyperplane of equation $w.x + b$ =0.

The goal is to find the hyperplane that separates the positive from the negative examples; the one that maximizes the margin would generalise better as compared to other possible separating hyperplanes.

A separating hyperplane in canonical form must satisfy the following conditions :

$$w.x_i + b \geq 1 \quad if \;\; y_i = 1$$
$$w.x_i + b \leq -1 \quad if \;\; y_i = -1$$

These can be combined into one set of inequalities

$$y_i(w.x_i + b) \geq 1 \quad \forall i \in \{1, ..., m\}$$

The distance $d(w, b, x)$ of a point x from the hyperplane (w, b) is,

$$d(w, b, x) = \frac{|w.x + b|}{||w||}$$

The optimal separating hyperplane is given by maximizing the margin M given by the equation :

$$M = \min_{x_i | y_i = -1} d(w, b, x_i) + \max_{x_i | y_i = 1} d(w, b, x_i) = \frac{2}{||w||}$$

To maximize the margin M, one need to minimize :

$$\Phi(w) = \frac{w^2}{2}$$

The solution to the optimisation problem is given by the saddle point of the Lagrange functional (Lagrangian)

$$L(w, b, \alpha) = \frac{1}{2} w.w - \sum_{i=1}^{m} \alpha_i [y_i(w.x + b) - 1]$$

Where α are the Lagrange multipliers. The Lagrangian has to be minimized with respect to w, b and with $\alpha \geq 0$. This problem can easily be transformed into the dual problem, and hence the solution is given by :

$$\alpha^0 = argmax \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j (x_i.x_j)$$

with constraints,

$$\alpha_i \geq 0, \forall i \in 1, ..., m$$

$$\sum_{i=1}^{m} \alpha_i^0 y_i = 0$$

2.2 The non-linear separable case

In this case, the set of training vectors of two classes are non-linearly separable. To solve this problem, Cortex and Vapnik [7] introduce non-negative variables, $\xi_i \geq 0$, which measure the miss-classification errors. The optimisation problem is now treated as a minimization of the classification error [8]. The separating hyperplane must satisfy the following inequality :

$$(w.x_i) + b \geq +1 - \xi_i, \quad if \quad y_i = +1$$

$$(w.x_i) + b \leq -1 + \xi_i, \quad if \quad y_i = -1$$

The generalised optimal separating hyperplane is determined by the vector w, that minimizes the functional,

$$\phi(w, \xi) = \frac{w^2}{2} + C \sum_{i=1}^{m} \xi_i$$

Where $\xi = (\xi_1, ..., \xi_m)$ and C are constants.
The dual problem corresponding to this case is slightly different from the linear separable case, the goal now is to maximize :

$$\sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j (x_i.x_j)$$

subject to

$$\sum_{i=1}^{m} \alpha_i^0 y_i = 0$$

$$0 \leq \alpha_i \leq C \quad \forall i \in \{1, ..., m\}$$

2.3 Kernel support vector machines

In the case where a linear boundary is inappropriate, the SVM can map the input vector into a high dimensional space through function $\phi(x)$, where the SVM constructs a linear hyperplane in the high dimensional space.

Since finding the SVM solutions involve the dot products of the sample vectors $x_i.x_j$, kernel functions play a very important role in avoiding explicit producing the mappings, and avoiding the curse of dimensionality, so that $\phi(x_i).\phi(x_j) = K(x_i.x_j)$, i.e., the dot product in that high dimensional space is equivalent to a kernel function of the current space.

In the linear and non-linear cases, the optimal separating hyperplane defined by w^0 and b^0 is determined as follows :

$$H : w^0.\phi(x) + b^0$$

where,

$$w^0 = \sum_{SV} \alpha_i \phi(x_i) y_i$$

and

$$b^0 = 1 - w^0.x_i \quad for \quad x_i \quad with \quad y_i = 1$$

The classification function is :

$$class(x) = Sign(w^0.\phi(x) + b^0)$$

$$class(x) = Sign[\sum_{SV} \alpha_i^0 y_i \phi(x_i).\phi(x) + b^0]$$

$$class(x) = Sign[\sum_{SV} \alpha_i^0 y_i K(x_i.x) + b^0]$$

Some widely used kernels are :
Linear : $K(x, y) = x.y$
Polynomial : $K(x, y) = (x.y + 1)^d$
Radial Basis Function (RBF): $K(x, y) = exp[-\frac{|x-y|^2}{2\sigma^2}]$

3 Database and recognition system

3.1 Database

For training, we use 5300 sentences of the French BREF80 database, pronounced by 80 speakers. These sentences are recorded at 16 KHz with 16 bits. This is a general purpose database, and the sentences have no relationship with our application.

The test database contains one hour of recording speech of radio braodcast news at 16 kHz. It is segmented into fragments of duration is 20s. This recorded speech have been pronounced by several speakers (different from the speakers of the training database). In our application we choose 10 different keywords.

3.2 Recognition system

The recognizer used in this work is a speaker independent HMM system. The modeled unit is the phone, each phone is represented by 3-state, strictly left-to-right, continuous density HMM. A word is represented by the concatenation of phone models. The number of probability density function (pdf) per state is determined during the training phase.

The parameterization is based on MFCC (Mel-Frequency Cepstral Coefficients) parameters. The user can modify this parameterization : size of the analyzing window, shift, number of triangular filters, lower and upper frequency cut-off of the filter bank, and number of the cepstral coefficients. Finally the delta (the first derivation) and acceleration coefficients (the second derivation) are added. In the following experiments, the acoustic feature vectors are built as follows : 32ms frames with a frame shift of 10ms, each frame is passed through a set of 24 triangular band-pass filter resulting in a vector of 35 features, namely 11 static mel-cepstral coefficients (C_0 is removed), 12 delta and 12 delta delta coefficients.

In the recognition phase, we adjust parameters in order to have no deletion keywords (as consequence we obtained a great number of insertion keywords).

4 SVM for keyword spotting

In this section, we describe the way we have utilised the SVM for the keyword detection. After the recognition phase, the goal is to classify a sequence of detected keywords into correct and incorrect keywords.

For each keyword, we compute the frames assigned to each phone state and extract the acoustic feature. In this work, 13 different features have been observed to provide information about the correctness of a word hypothesis were utilised. These features, as computed for each word, are:

1. The total number of frames
2. The number of frames of the first phone
3. The number of frames of the end phone
4. The minimum number of frames in word phones
5. The maximum number of frames in word phones
6. The number of phones
7. The average number of frames per phone
8. The log phone posterior probability of the first phone
9. The log phone posterior probability of the end phone
10. The minimum log phone posterior probability
11. The maximum log phone posterior probability
12. The average per-frame log phone posterior probability
13. The average duration-normalised log phone posterior probability

For each keyword (insertion keyword, recognized correct keyword) we compute a feature vector that is used as input for the SVM.
In our work the insertion keyword belongs to the class labelled -1 and the correct keyword is assigned to the class labelled +1. Thus, we classify the correct and the incorrect keywords.

The SvmFu package was used for these experiments. It is available as freeware on : http://www.kernel-machines.org/

5 Experimental results

The database used in the second phase of our experiments (after the recognition phase) is composed of 600 keywords for the training data, and 560 keywords for the test data. In this work, we use linear and Radial Basis Function (RBF) kernels.

To evalute the performances of our recognizers, we use two evaluation rates :

The False Acceptance Rate also called False Alarm Rate (FAR). It is defined by the equation :

$$FAR = \frac{Total\ \ False\ \ Acceptance}{Total\ \ False\ \ Attempts}$$

The False Rejection Rate (FRR). FRR is defined by the equation :

$$FRR = \frac{Total\ \ False\ \ Rejection}{Total\ \ True\ \ Attempts}$$

Plotting a graph of FRR versus FAR gives a Receiver Operating Characteristics (ROC) graph.

The resulting ROC curves, using linear SVM by varying the value of the parameter C, $C \in \{1, 10, 100\}$ are presented in figure (Fig. 1). They show that better performances are obtained in case $C = 1$.

The figure (Fig. 2) presents ROC curves corresponding to the performance obtained using RBF kernel by varying the value of the parameter $sigma$, with $C = 1$. In our experiments, we test several values of $sigma$ and in this article, in order to alleviate the figure, we choose to present only the interesting curves for $sigma \in \{5, 10, 70\}$. The best results are achieved for $sigma = 70$.

The figure (Fig. 3) demonstrates that the obtained results for RBF kernel are better than results obtained for linear SVM.

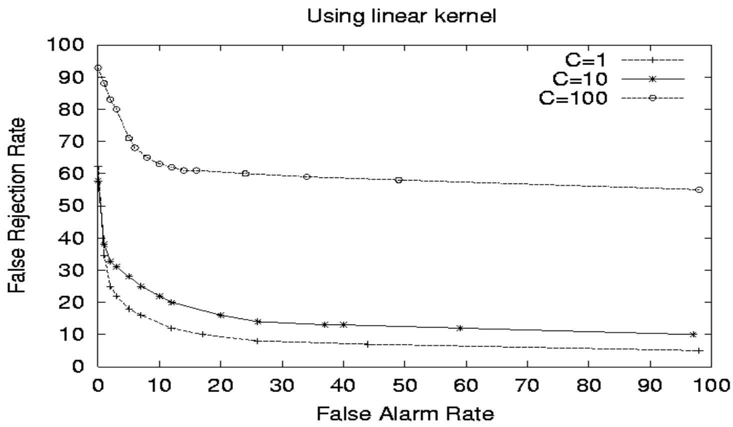

Fig. 1. ROC curves on test data using a linear kernel by varying the value of C

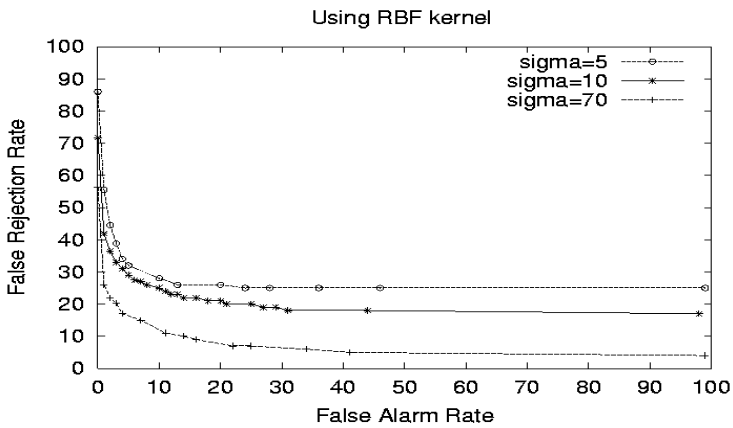

Fig. 2. ROC curves on test data using a RBF kernel by varying the value of $sigma$

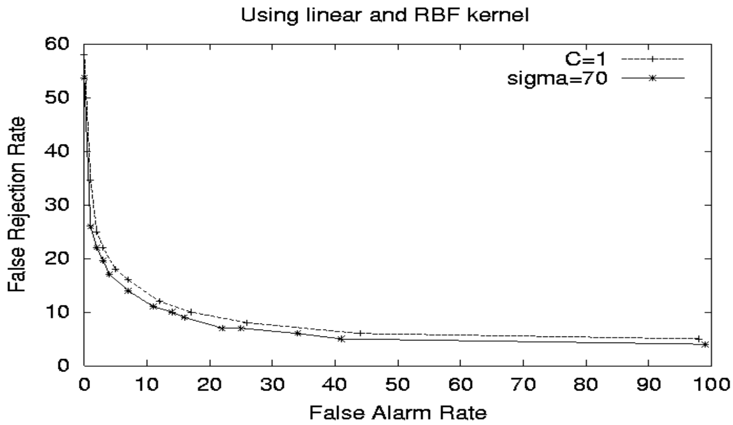

Fig. 3. ROC curves comparing the performance of the linear and the RBF kernels

6 Conclusion

This paper presents the results achieved by SVM techniques for the keyword spotting problem using linear and RBF kernels. Taking into account that is a first approach using support vector machine in keyword spotting, the results obtained seem to be very promising. In the near future, feature vector will be adjusted for each keyword in order to have more information about it. Other different kernel types and parameters will be experimented.

References

1. Kharroubi, J., Petrovska, D., Chollet, G.: Combining GMM's with Suport Vector Machines for Text-independent Speaker Verification. EuroSpeech (2001) 1761-1764
2. Ousna, E., Freund, R., Girosi, F.: Support Vector machines : Training and Applications. A.I .Memo 1602, MIT (1997)
3. Ganapathiraju, A.: Support Vector Machines for Speech Recognition. Thesis Mississipi State University, January (2002)
4. Gunn, S.: Support Vector Machines for Classification and Regression. Technical Report ISIS-1 (1998)
5. Vapnick, V.: The Nature of Satistical Learning Theory. Springer-Verlag (1995)
6. Vapnick, V.: Satistical Learning Theory. Johon Wiley and Sons (1998)
7. Cortes, C., Vapnick, V.: Support Vector Networks. Machine Learning, Vol. 20. (1995) 273-297
8. Burges, C.: A Tutorial on Support Vector Machines for Pattern Recognition Data Mining and Knowledge Discovery, 2(2) (1998)