A comparison of String Kernels and discrete Hidden Markov Models on a Spanish Digit Recognition Task

J. Goddard[1], A. E. Martínez[1], F. M. Martínez[1], H. L. Rufiner[2]
[1]Department of Electrical Engineering, Universidad Autonoma Metropolitana, Iztapalapa, Mexico*
[2]Cybernetics Laboratory, Engineering Faculty, National University Entre Rios, Argentina

Abstract - **String kernels have been introduced recently in an attempt to apply support vector machine (svm) classifiers to variable-length sequential data from a discrete alphabet. They have been used in the areas of text classification and bioinformatics, where notable results have been obtained. In the present paper string kernels are applied to a Spanish digit recognition task and their performance is compared to that of discrete hidden markov models (dhmm). It is found that string kernels produce comparable results and may offer an alternative discriminative approach for certain speech recognition tasks.**

Keywords - **discrete hidden Markov models, digit recognition, string kernels, support vector machines**

I. Introduction

SVM classifiers were introduced in the early 1990's by Vapnik [1] and since then they have been applied to a wide variety of classification problems with excellent results, usually outperforming other techniques. Their success has to do principally with their generalization ability, nevertheless they also provide an attractive discriminative approach to classification problems through the use of kernels.

The basic idea behind svms for a two class classification problem is to map the data in each class into linearly separable sets in a higher dimensional inner product vector space, called the feature space. A separating hyperplane is then found in feature space which maximizes the minimal distance, known as the margin, between the hyperplane and the closest points of the classes to it. New patterns are then classified according to the side of the hyperplane they are mapped to. The kernel function k is related to the transformation ϕ through the inner product by:

$$k(x,y) = <\phi(x),\phi(y)> \qquad (1)$$

The kernels most frequently employed are radial basis functions and polynomials, in which case the transformation is not explicitly defined. These kernels have usually been applied to classification problems involving non-sequential data in Euclidean space.

In the case of automatic speech recognition, biosequence classification, and text categorization however, the data is naturally variable-length and sequential in nature and one would like to use an svm classifier which directly incorporates this additional information. Recently kernels have been extended to discrete spaces [2,3] and to problems involving sequential data [4,5,6]. Two classes of kernels, the Fisher kernel and string kernels, have been defined on the relevant sequence spaces using explicit transformations to feature space (c.f. section II for more details). This allows two sequences to be compared for similarity; for example, if the sequences represent vectors derived from the mel frequency cepstral coefficients (mfcc) corresponding to the frames of two speech waveforms, the Fisher kernel provides a way to compare the similarity between them.

Both classes of kernels have been applied successfully to classification tasks related to biosequences and text documents. Fisher kernels, in particular, have also been used for automatic speech recognition [7] and speaker verification and identification [8]. These results show that svm classifiers with specially defined kernels may provide a viable and interesting alternative to other classifiers, particularly those using hmms, for problems involving variable-length sequential data.

In the present paper some of these ideas are explored further by applying svm classifiers with certain string kernels to a Spanish digit recognition task. The sequence space for the speech signals is derived from a codebook of vectors obtained from the k-means clustering algorithm. Different types of string kernels are then defined on this sequence space and the corresponding svm classifier is trained on the sequences. The results obtained are compared to those of dhmms.

The paper is organized as follows: in the next section the string kernels considered here are described in detail as well as their integration into an svm classifier. Section III describes the speech data used for the experiments and presents the results obtained by applying string kernels and

The first three authors are grateful for financial support from CONACYT under contract No. 400200-5-31929A

dhmms. Finally some conclusions related to the paper are given.

II. Methods

The methods used in the paper are svm classifiers with string kernels and dhmms. While dhmms are well-known in the speech community [9], svm classifiers, and in particular string kernels, are more recent introductions and their definitions are given here.

A. Kernels

Let X be a set, then a (positive-definite) kernel k on XxX is a real-valued function:

$$k{:}X\mathrm{x}X\rightarrow\Re$$

such that
(i) k is symmetric: $k(x,y) = k(y,x) \quad \forall\ x,y \in X$, and
(ii) k is positive definite: $\forall\ n \geq 1$

$$\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geq 0$$

$\forall\ c_1,\ldots,c_n \in \Re$ and $x_1,\ldots,x_n \in X$

It can be shown [10] that given a kernel k, there exists a (Reproducing Kernel) Hilbert space H and a transformation ϕ: X→ H such that (1) holds. This allows elements of X to be compared for similarity, and in fact a simple calculation shows that a pseudo-metric d can be defined on XxX through the relation:

$$d(x,y) = ||\phi(x)-\phi(y)|| = \sqrt{k(x,x) - 2k(x,y) + k(y,y)}$$

This introduces the concept of a distance into X.

The most commonly used kernels are the polynomial and radial base function kernels defined on $\Re^m \mathrm{x} \Re^m$ by:

$$k(x,y) = (<x,y> + 1)^d$$

$$k(x,y) = \exp(-||x-y||^2/2\sigma^2)$$

respectively, where d = 1,2,… and $\sigma \in \Re$. For these kernels the transformation in (1) is not defined explicitly, and the kernels are applied directly in the original data space. This is known as the 'kernel trick'.

B. String kernels

Let A be a finite alphabet of size N and X the set of all sequences of elements from A. For $n \geq 1, \Re^{N^n}$ can be considered as the usual euclidean inner product space indexed by all sequences from A of length n. The transformation ϕ:

$$\phi{:}\ X\rightarrow \Re^{N^n} \qquad (2)$$

can be defined for $x \in X$ by forming the vector in $\Re^{N^n}$ whose value for any index $\alpha \in A^n$ is the number of occurrences of α in x; more concisely, if

$$\phi(x) = \{r_\alpha\}_{\alpha\in An}$$

then

$$r_\alpha = \text{number of occurrences of } \alpha \text{ in x} \qquad (3)$$

To illustrate this transformation by means of a concrete example, let A = {1,2,3}, n = 2, and x = 1,2,2,2,3,1. Feature space is $\Re^9$ which is indexed by all pairs $(a,b) \in A\mathrm{x}A$. The vector x is then transformed into $(0,1,0,0,2,1,1,0,0) \in \Re^9$.

Here only contiguous sequences are considered although in other papers, such as [5], non-contiguous sequences have been used.

The n-gram string kernel is then defined as in (1) using the transformation from (2). Two sequences are similar if they have similar n-tuples.

This string kernel is also known as the spectrum kernel in the bioinformatic literature (c.f. [6]).

In fact it turns out that a variant of the n-gram kernel, which is termed the binary n-gram kernel here, proves to give better classification results on the digit recognition task. The binary n-gram kernel is defined by modifying (3) to the following:

$$r_\alpha = 1 \text{ if } \alpha \text{ occurs in x, and 0 if not}$$

It is interesting to note that the transformation corresponding to the binary n-gram kernel maps the elements of X to the vertices of the hypercube in $\Re^{N^n}$. If n = 2 and N = 32, then there are more than 10^{270} vertices on this hypercube, although, in general, the number of non-zero coordinates of $\phi(x)$ is bounded above by length(x)-n+1.

Finally, in the present paper, both the n-gram and binary n-gram kernels k are normalized in order to take into account sequences of different lengths. This means that the kernel k' ultimately used here is defined in both cases by:

$$k'(x,y) = \frac{k(x,y)}{\sqrt{k(x,x)k(y,y)}}$$

For the purpose of this paper, A can be considered to be {1, 2,…, 32}, an enumeration of the prototypes in a suitably defined vector codebook. The sequence $a_1, a_2,\ldots, a_r$ in X associated with a given speech signal is the same as that which would be associated with a dhmm; that is, the a_i's are the indices of the prototypes which are closest to the vectors derived from the mfccs of the frames in the signal.

C. SVM Classifiers

A detailed introduction to svm classifiers can be found in [11], and a good tutorial in [12]. For the present paper it suffices to formulate an svm classifier, for a two class problem, in the following way: Let D = $\{x_1,x_2,\ldots,x_m\}$ be a data set in X where each pattern x_i belongs to one of two classes $y_i \in \{-1, +1\}$, and k a kernel on XxX. It can be shown that by maximizing the following objective function:

$$W(\alpha_1,\ldots,\alpha_m) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \qquad (4)$$

subject to the constraints

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

a suitable hyperplane can be found in feature space which is used to perform the classification. The constraint C is introduced in case the data is not linearly separable in feature space, perhaps due to the presence of noise or outliers, and is known as a soft margin. It represents a trade-off between maximizing the margin, as explained in the introduction, and minimizing the classification error on the training set. For an optimal solution $\{\alpha^0_i\}$ to (4), it is found that for the points which lie on or within the margin, or are incorrectly classified, $\alpha^0_i > 0$. These points are called support vectors. The rest of the points have $\alpha^0_i = 0$. The support vectors are therefore the most informative points in the data set for the classifier, and solving (4), using only these points, would produce exactly the same classifier.

Classification of a test pattern x is given by designating its class to be the sign of:

$$\sum_{i=1}^{m} y_i \alpha_i^0 k(x, x_i) + b \qquad (5)$$

where b is a constant whose value can be found from the optimal solution to (4) (c.f. [11]). The relation of (5) to the hyperplane mentioned above is that if the transformation ϕ were explicitly introduced as in (1), the equation for the hyperplane would be:

$$\sum_{i=1}^{m} y_i \alpha_i^0 \langle z, \phi(x_i) \rangle + b = 0$$

where z is in feature space. Equation (5) then signifies that classification of test points takes place according to the side of the hyperplane the point lies on. In the formulation of (4) the explicit use of the transformation ϕ is not required, although it is vital in order to gain a geometrical understanding of the process. Further, it is required for string kernels by their very definition.

D. Multiclass Classification

The svm classifiers of the previous subsection are defined for a two class classification problem. In order to extend this to a multiclass classification problem several schemes have been proposed and there is, as yet, no definitive method. In the present paper a '1 vs. rest' basis is adopted; that is, an svm is trained for each class by separating that class from the rest of the training data. In this way, a different function in (5) is obtained for each class. A test pattern is then classified as belonging to the class with the maximum function value.

III. Results of a Spanish Digit Recognition Task

The Spanish digits were obtained by recording several repetitions from 6 young Mexican adults, 5 male and 1 female. The recordings were made in a normal office environment which meant that the recording quality varied amongst the utterances. A sampling frequency of 16 kHz was used and the waveforms were converted to vectors with 13 coefficients, 12 mfccs and log energy. The digits were recorded using WASP [13] and the vectors were found with Voicebox [14].

Roughly 66% of the data was used for training and the rest for testing. This meant that the training set consisted of 19,718 vectors corresponding to 396 digits, and the test set of 9,896 vectors representing 197 digits. A k-means clustering algorithm was applied to the training set, using [15], to obtain 32 prototypes. The digits were then represented by variable-length sequences of integers from {1,…,32}.

Various experiments were conducted using dhmms, n-gram kernels and binary n-gram kernels. In the case of the dhmms, a standard left-to-right architecture was used to model each individual digit, all with the same number of

Table 1
Classification results with different dhmms

Number of states	% correct on Test set
1	71.57
2	79.70
3	87.82
4	87.31
5	88.83
6,7,8,9	90.35

states. For the svms a value of 10 was chosen for C. Although a validation set can be used to find suitable values for C, in the present paper no special tuning was conducted.

Tables 1 and 2 show the classification results obtained for different numbers of states and values of n. It can be seen from these results that svm classifiers with binary string kernels and dhmms performed significantly better than the 2-gram kernel, and in turn produced similar results for their best values.

IV. Conclusions

In this paper svm classifiers with certain string kernels were described and their classification results on a Spanish digit recognition task were compared to those obtained using dhmms. The results show that svms with binary n-gram kernels can provide comparable classification accuracy for this task. This is encouraging as it provides a conceptually simple and discriminative alternative to dhmms, which works directly on the variable-length sequential data.

HMMs are the method of choice in automatic speech recognition, although deficiencies have been noted and various hybrid schemes have been proposed. The present paper is an initial attempt to provide a feasible alternative to dhmms for certain automatic speech recognition tasks. Further experimentation is required as the area is new, however results obtained for biosequences and text categorization re-enforce the findings above. Two interesting related directions of research are the connection between the n-gram kernel and the Fisher kernel [16], and the use of a 'mismatch' kernel introduced in [17].

Table 2
Classification results with svms

string kernel	% correct on Test set
binary 2-gram	89.34
binary 3-gram	91.37
binary 4-gram	88.83
2-gram	66.50

References

[1] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer, 1995.

[2] D. Haussler, "Convolution kernels on discrete structures," Technical Report UCSC-CRL-99-10, University of California in Santa Cruz, Computer Science Department, July 1999.

[3] C. Watkins, "Kernels from matching operations," Technical Report CSD-TR-98-07, Royal Holloway, University of London, Computer Science Department, July 1999.

[4] T. S. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," in M S. Kearns, S. A. Solla, and D. A. Cohn, editors, Advances in Neural Information Processing Systems, 11. MIT Press, 1998.

[5] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, "Text Classification Using String Kernels," *Journal of Machine Learning Research*, 2:419-444, Feb 2002.

[6] C. Leslie, E. Eskin and W. Stafford Noble, "The spectrum kernel: A string kernel for SVM protein classification" *Proceedings of the Pacific Symposium on Biocomputing (PSB-2002)*, Hawaii: January 2-7, 2002, pp. 564-575.

[7] N. D. Smith and M. J. F. Gales, "Using SVMs and discriminative models for speech recognition," International Conference on Acoustics, Speech, and Signal Processing, 2002.

[8] S. Fine, J. Navrátil and R. A. Gopinath, "Enhancing GMM scores using SVM "hints"," Proceedings of Eurospeech 2001, Scandanavia, 2001, pp. 1757-1761.

[9] L. Rabiner and B-H. Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, 1993.

[10] B. Schölkopf, "The Kernel Trick for Distances," Microsoft Research Technical Report MSR-TR- 2000-51, 2000.

[11] N. Cristianini and J. Shawe-Taylor, *Support Vector Machines*, Cambridge University Press, 2000.

[12] C. Campbell, "An Introduction to Kernel Methods,", in *Radial Basis Function Networks: Design and Applications*. R. J. Howlett and L.C Jain (eds). Berlin: Springer Verlag, 2000.

[13] WASP: Speech recording system available from http://www.phon.ucl.ac.uk/resource/sfs/

[14] Voicebox: Speech Processing Toolbox available from http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html

[15] Clustering Toolbox available from http://www.cs.ucl.ac.uk/staff/D.Corney/ClusteringMatlab.html

[16] C. Saunders, J. Shawe-Taylor, and A. Vinokourov. String Kernels, Fisher Kernels and Finite State Automata. In S. Becker, S. Thrun, and A. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, 2003.

[17] C. Leslie, E. Eskin, J. Weston and W. Stafford Noble. ``Mismatch String Kernels for SVM Protein Classification." . In S. Becker, S. Thrun, and A. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, 2003.