Support Vector Machines as Acoustic Models in Speech Recognition

Sven E. Krüger, Martin Schafföner, Marcel Katz, Edin Andelic, Andreas Wendemuth
Otto-von-Guericke Universität Magdeburg, IESK, Kognitive Systeme, PF 4120, 39016 Magdeburg, Germany
sven.krueger@ovgu.de

Abstract

Speech recognition is usually based on Hidden Markov Models (HMMs), which represent the temporal dynamics of speech very efficiently, and Gaussian mixture models, which do non-optimally the classification (acoustic modeling) of speech into single speech units (phonemes). In this paper we present an overview about the use of Support Vector Machines (SVMs) for the classification task by integrating SVMs in a HMM-based speech recognition system. SVMs are very appealing due to their association with statistical learning theory and have already shown good results in pattern recognition. In our hybrid SVM/HMM system we use SVMs as acoustic models in a HMM-based decoder. We train and test the hybrid system on the DARPA Resource Management (RM1) corpus, showing better performance than HMM-based decoder using Gaussian mixtures. To reduce the effort for training of the SVMs, we also use mixtures of SVMs which scales nearly linearly with respect to the number of training vectors making it easier to deal with the large amount of speech data.

Introduction

Support Vector Machines (SVMs) [1, 2] have become quite popular for many applications of pattern classification, mainly due to their great ability to generalize, often resulting in better performance than traditional techniques, such as artificial neural networks.

SVMs have also been successfully integrated [3] into continuous speech recognition which is based on Hidden Markov Models (HMMs). In HMM speech recognizers the classification of single speech units (phonemes) is usually done with Gaussian mixture models (GMMs), which do not discriminate well. In our hybrid SVM/HMM system these Gaussian mixtures, i.e., the acoustic models on the phoneme level, are replaced with SVMs. In such a hybrid SVM/HMM system the problem of quadratic computer time for SVM training becomes especially severe because of the large amount of training data in speech recognition.

In order to overcome this drawback, we also integrate parallel mixtures of SVMs (as introduced in [4]) into speech recognition. We do this by using the SVM mixtures as acoustic models in a HMM framework. Since the mixtures of SVMs need only nearly linear training time [4] we could use a one-vs-rest approach for the multi-class classification, whereas using single SVMs we use one-vs-one.

The hybrid speech recognition system

In Automatic Speech Recognition (ASR) [5] Hidden Markov Models (HMMs) with transition probabilities and emission probabilities are used. We have to find the HMM M^* which maximizes the posterior probability $p(M|X)$ of the hypothesized HMM M given a sequence X of feature vectors. Since this probability cannot be computed directly, it is usually split using Bayes' rule into the acoustic model (likelihood) $p(X|M)$ and a prior $p(M)$ representing the language model: $p(M|X) \propto p(X|M)p(M)$.
In the usual approach the emission probabilities (acoustic models on phoneme level) are modelled with Gaussian mixture models (GMMs). In our hybrid approach the acoustic models are estimated with Support Vector Machines (SVMs).

Estimation of the acoustic models

Support Vector Machines

Support Vector Machines (SVMs) [1, 2] developed from the theory of *structural risk minimization* are linear classifiers (i.e. the classes are separated by hyperplanes), but they can be used for non-linear classification by the so-called *kernel trick*. Using this, the dot-product in the feature space (which is nonlinearly related to the input space $\mathcal{R}^n$) is equal to the so-called *kernel function* $k(\mathbf{x}, \mathbf{y})$. We use the Gaussian kernel $k(\mathbf{x}, \mathbf{y}) = \exp(-\gamma||\mathbf{x} - \mathbf{y}||^2)$.

Given a training set $\{\mathbf{x}_i, y_i\}$ of N training vectors $\mathbf{x}_i \in \mathcal{R}^n$ and corresponding labels $y_i \in \{-1, +1\}$ and a kernel function $k(\mathbf{x}, \mathbf{y})$ the SVM finds a optimal separating hyperplane in the feature space by solving a quadratic programming problem resulting in some α_i and in b. The output of the SVM for a pattern $\mathbf{x}$ is

$$s(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b, \qquad (1)$$

where $s(\mathbf{x}) > 0$ means that $\mathbf{x}$ is classified to class $+1$. We use the software library *Torch* [6] for the training of the SVMs.

Mixture of SVMs

The idea of mixture models is to use a set of experts (here SVMs) and a gating network to weight the outputs of the single experts for the final output. Here we use parallel mixtures of SVMs as introduced in [4]. The output of the mixture for an input vector $\mathbf{x}$ is

$$f(\mathbf{x}) = \sum_{m=1}^{M} w_m(\mathbf{x}) s_m(\mathbf{x}) \qquad (2)$$

where M is the number of experts (here single SVMs), and $s_m(\mathbf{x})$ is the output of the m^{th} SVM (i.e., expert) and $w_m(\mathbf{x})$ is the weight for the m^{th} SVM. The $w_m(\mathbf{x})$ form the gating network and are defined by a Multi Layer Perceptron (MLP), and trained using a backpropagation algorithm. During training, the single SVMs are trained separately over one subset of the order of N/M. If N/M remains constant, then the total training time scales nearly *linearly* with N.

Estimation of posterior probabilities from the SVM output

There is no clear relationship between the output $s(\mathbf{x})$ of the SVM (and $f(\mathbf{x})$ of the SVM mixture, respectively) and the posterior class probability $p(y=+1|\mathbf{x})$ that the pattern $\mathbf{x}$ belongs to the class $y=+1$. We estimate the posterior class probability with the method of Platt [7].

Multi-class classification

Since the (mixtures of) SVMs are *binary* classifiers we have to use a certain method for more than two classes. For the SVM mixtures we use the one-versus-rest approach, where a mixture of SVMs learns to discriminate one class from all other classes. Having K classes means we need to train K mixtures of SVMs.

For the single SVMs, we use a one-vs-one approach, where a SVM learns to discriminate one class from one other class.

Experiments

Experimental setup

Our implementation of a hybrid HMM system using acoustic models other than Gaussian mixture models is based on the Hidden Markov Toolkit (HTK). We use the DARPA Resource Management (RM1) corpus. Preprocessing is performed in the usual way (short-time FFT, cepstral analysis, etc.) resulting in feature vectors containing altogether 39 components. We use the standard 72-speaker training set, and our systems were evaluated using the speaker independent Feb'89 test set and the standard RM word-pair grammar.

First we compute a time alignment using a standard Gaussian mixture HMM decoder to get the state (label) for each feature vector. Using monophone models with three HMM states each and an optional pause model `sp` (to model the pauses between words) we have $3 \times 48 + 1 = 145$ different states altogether. As baseline of our experiments (using a GMM with 8 mixtures) we have a word accuracy of 91.96%.

Results

The mixtures of SVMs are trained with the full training set (about $N = 988,000$ feature vectors) in a one-vs-rest approach. Having 145 classes (HMM models) we have to train 145 mixtures of SVMs. For each mixture we use 30 SVMs (these are the experts, each to be trained with $N/30$ training vectors on average). The gating network ($w_m(\mathbf{x})$ in Eq. (2)) is a MLP with 90 hidden neurons. Using the one-vs-one approach, $145(145-1)/2 = 10440$ single SVMs, i.e., one for each pair, are trained with on average $2N/145 \approx 13600$ training vectors. Using a skipping approach, we skip certain pairwise classifications of two classes, if it appears not meaningful to discriminate them *from each other* (see [8] for more details). Our results as given in Tab. 1 are better than the baseline results with Gaussian mixtures.

classifier (in a one-state-HMM)	Word accuracy
Baseline (GMM with 8 mixtures)	91.96 %
Mixture of SVMs (one-vs-rest)	92.23 %
single SVMs (one-vs-one)	94.10 %

Table 1: Word accuracy of the SVM approach compared with the baseline.

Conclusion

We have created a SVM/HMM hybrid system for continuous speech recognition, using SVMs (instead of GMMs) to estimate the acoustic models on the phoneme level. Our system was tested on the DARPA RM1 corpus (speaker independent) and we got a relative word error rate reductions of up to 26% compare to Gaussian mixtures.

References

[1] V. Vapnik. *The Nature of Statistical Learning Theory.* Springer Verlag, 1995.

[2] B. Schölkopf and A. J. Smola. *Learning with Kernels.* MIT Press, 2002.

[3] J. Stadermann and G. Rigoll. A hybrid SVM/HMM acoustic modeling approach to automatic speech recognition. In *INTERSPEECH-2004 ICSLP*, pages 661–664, 2004.

[4] R. Collobert, S. Bengio, and Y. Bengio. Parallel mixture of svms for very large scale problems. *Neural Computation*, 14:1105–1114, 2002.

[5] Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of Speech Recognition.* Prentice Hall, Englewood Cliffs, 1993.

[6] Ronan Collobert and Samy Bengio. SVMTorch: Support vector machines for large-scale regression problems. *The Journal of Machine Learning Research*, 1:143–160, 2001.

[7] J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In B. Schölkopf C. J. C. Burges and A. J. Smola, editors, *Advances in Large Margin Classifiers.* MIT Press, 1999.

[8] Sven E. Krüger, Martin Schafföner, Marcel Katz, Edin Andelic, and Andreas Wendemuth. Speech recognition with support vector machines in a hybrid system. In *Proc. INTERSPEECH-2005*, pages 993–996, 2005.