A SPARSE MODELING APPROACH TO SPEECH RECOGNITION BASED ON RELEVANCE VECTOR MACHINES[1]

J. E. Hamaker and J. Picone

Institute for Signal and Information Processing
Department for Electrical and Computer Engineering
Mississippi State University, Mississippi State, MS
{hamaker, picone}@isip.msstate.edu

A. Ganapathiraju

Conversay Computing Corporation
Redmond, Washington
aganapathiraju@conversay.com

ABSTRACT

In this paper, we compare two powerful kernel-based learning machines, support vector machines (SVM) and relevance vector machines (RVM), within the framework of hidden Markov model-based speech recognition. Both machines provide nonlinear discriminative classification ability: the SVM by kernel-based margin maximization and the RVM using a Bayesian probabilistic framework. The hybrid systems are compared on a vowel classification task and on the continuous speech Alphadigits corpus. In both cases, the RVM system achieves better error rates with significantly fewer parameters.

1. INTRODUCTION

The most prominent modeling technique for speech recognition today is the hidden Markov model with Gaussian emission densities. However, they suffer from an inability to learn discriminative information. Artificial neural networks have been proposed as a replacement for the Gaussian emission probabilities under the belief that the ANN models provide better discrimination capabilities. However, the use of ANNs often results in over-parameterized models which are prone to overfitting. Techniques such as cross-validation have been suggested as remedies to the overfitting problem but employing these is wasteful of both resources and computation. Further, cross-validation does not address the issue of model structure and over-parameterization.

Recent work on machine learning has moved toward automatic methods for controlling generalization and parameterization. One model that has gained much popularity is the support vector machine (SVM). SVMs use the principle of structural risk minimization to simultaneously control generalization and performance on the training set. Previously [1], we have employed the SVM in a hybrid framework for speech recognition. While the HMM/SVM hybrid produced a decrease in the error rate, the implementation had some significant shortfalls which we address in this work.

First, SVMs are not probabilistic in nature and, thus, are not able to adequately express the posterior uncertainty in predictions. This is particularly important in speech recogntion because there is significant overlap in the feature space. SVMs also make unnecessarily liberal use of parameters to define the decision region. In this paper, we describe a Bayesian model, termed the relevance vector machine (RVM) [2], which takes the same form as an SVM model, but provides a fully probabilistic alternative to SVMs. Sparseness of the model is automatic using automatic relevance determination methods (ARD) [3]. We demonstrate an initial application of RVMs to speech recognition and compare peformance to a hybrid SVM system.

2. RELEVANCE VECTOR MACHINES

RVMs are an application of the evidence framework defined by MacKay [3] to kernel machines. As with SVMs, the RVMs are formed by defining a vector-to-scalar mapping as a weighted linear combination of basis functions,

$$y(\boldsymbol{o};\boldsymbol{w}) = w_o + \sum_{i=1}^{M} w_i \phi_i(\boldsymbol{o}) = \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{o}), \qquad (1)$$

with $\boldsymbol{w} = \left[w_o, w_1, \ldots, w_M\right]^T$ and $\boldsymbol{\phi} = \left[1, \phi_1(\boldsymbol{o}), \ldots, \phi_M(\boldsymbol{o})\right]^T$. $\boldsymbol{\phi}$ is a set of M basis functions that each form a nonlinear mapping of the observed vector, $\boldsymbol{o}$, to a scalar. The weights, w_i, are the parameters to be tuned to produce an accurate model (under some appropriate measure) of

1. This material is based upon work supported by the National Science Foundation under Grant No. IIS0095940. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

the phenomena we desire to learn. It is important to note the form of the basis functions, ϕ_i. Since SVMs are optimizing a distance measure in the transform space, they require that the basis functions take the form of a so-called Mercer kernel [4] (i.e. a kernel which acts as a dot-product in some space). No such restriction is placed on the basis functions that can be employed by the RVM. However, the power demonstrated by kernel machines gives compelling reason to pursue this special form of the basis function.

We reformulate (1) as

$$y(\boldsymbol{o};\boldsymbol{w}) = w_o + \sum_{i=1}^{M} w_i K(\boldsymbol{o};\boldsymbol{o}_i)\,, \qquad (2)$$

where there is one weight, w_i, associated with each training vector and $K(\boldsymbol{o};\boldsymbol{o}_i)$ defines a kernel function (not necessarily a Mercer kernel). Due to the large number of parameters in this model (one per observation) we must guard against overfitting of the model to the training data. SVMs use the control parameter, C, to implicitly balance the trade-off between training error and generalization. RVMs take a Bayesian approach and explicitly define an ARD prior distribution over the weights

$$p(\boldsymbol{w}|\alpha) = \prod_{i=0}^{N} \boldsymbol{N}\left(w_i|0,\frac{1}{\alpha_i}\right) = \frac{1}{\sqrt{(2\pi)^{N+1}|\boldsymbol{A}^{-1}|}} e^{-\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{w}} \qquad (3)$$

where we have defined $\boldsymbol{A} = diag(\alpha_o, \alpha_1, \ldots, \alpha_N)$. This prior acts to force weak components of the model toward a weight of zero, thus finding the inputs that are relevant to modeling.

Each weight in the RVM model has an individual hyperparameter, α_i, that is iteratively reestimated as part of the optimization process. As α_i grows larger, the prior on w_i becomes infinitely peaked around zero, forcing w_i to go to zero and, thus, contributing nothing to the summation in (2). This process automatically embodies the principle of Occam's Razor because it explicitly seeks the simplest model that satisfies the data constraints. In practice, the majority of the weights are pruned, resulting in an exceedingly sparse model with generalization abilities on par with SVMs [2]. To complete the Bayesian specification of the model, we have to specify a prior probability over the α_i. In practice we use a non-informative (flat) prior to indicate a lack of preference [2].

With SVMs the form of (2) arises from the need to optimize the classification margin in a high-dimensional space. With RVMs, however, the goal is to directly model the posterior probability distribution. The posterior is formed by generalizing the linear model to a probability distribution with a sigmoid link function,

$$\sigma(y) = \frac{1}{1+e^{-y}}\,, \qquad (4)$$

and adopting the two-class Bernoulli distribution for $P(\mathrm{t}|\boldsymbol{o})$ to give

$$\mathrm{P}(\mathrm{t}_i|\boldsymbol{w},\boldsymbol{o}_i) = [\sigma\{\mathrm{y}(\boldsymbol{o}_i;\mathbf{w})\}]^{\mathrm{t}_i}\ [1-\sigma\{\mathrm{y}(\boldsymbol{o}_i;\mathbf{w})\}]^{1-t_i} \qquad (5)$$

where $t_i \in \{0,1\}$. Under the assumption that each data sample is drawn independently, the likelihood of the training data set can be written as

$$P(\boldsymbol{t}|\boldsymbol{w},\boldsymbol{O}) = \prod_{n=1}^{N} \sigma_n^{t_n}(1-\sigma_n)^{1-t_n} \qquad (6)$$

where $\sigma_n = \sigma\{y(\boldsymbol{o}_n;\boldsymbol{w})\}$.

The objective of training is to find a parameter set which yields a model that is well-matched to the training data. In mathematical terms we want to find

$$(\hat{\boldsymbol{w}},\hat{\alpha}) = \underset{\boldsymbol{w},\alpha}{\mathrm{argmax}}\ \ p(\boldsymbol{w},\alpha|\boldsymbol{t},\boldsymbol{O})\,. \qquad (7)$$

A closed form solution to this maximization is not possible so we use the iterative approximation due to MacKay [3]. For a fixed α, find the locally most probable weights $\hat{\boldsymbol{w}}$. This process typically involves the use of a gradient descent optimization over the parameters. The Hessian with respect to the weights is then negated and inverted to give an approximation to the covariance, Σ, of a Gaussian posterior over the weights, centered about $\hat{\boldsymbol{w}}$. Using Σ and $\hat{\boldsymbol{w}}$ as the covariance and mean, respectively, of the Gaussian approximation, we can follow MacKay's approach [3] to update the $\{\alpha_i\}$ by

$$\alpha_i = \frac{\gamma_i}{\hat{w}_i^2}, \qquad \gamma_i = 1-\alpha_i\Sigma_{ii}\,. \qquad (8)$$

This iterative procedure is repeated until suitable

convergence criteria are met. Central to this iterative method is a second-order Newton maximization of $P(\boldsymbol{t}|\boldsymbol{w},\boldsymbol{O})p(\boldsymbol{w}|\alpha)$ requiring an inversion operation with complexity $O(N^3)$. As the quantity of training data increases, this becomes prohibitive. SVMs have a similar problem with scalability to large problems that has been addressed through iterative refinement of the training set [5]. Current research is focusing on similar methods for RVMs.

3. HYBRID ASR SYSTEM

The hybrid recognition architecture used in this work and shown in Figure 2 is a parallel of the SVM hybrid presented in [1]. Each phone-level classifier is trained as a one-vs-all classifier. The classifiers are used to predict the probability of an acoustic segment. For the SVM hybrid, a sigmoid posterior fit is used to map the SVM distance to a probability. The RVM output is naturally probabilistic so no link function is needed.

The HMM system is used to generate alignments at the phone level. Each phone instance is treated as one segment. Since each segment could span a variable duration, we divide the segment into three regions in a set ratio and construct a composite vector from the mean vectors of the three regions. In our experiments empirical evidence showed that a 3-4-3 proportion generally gave optimal performance. Figure 1 shows an example for constructing a composite vector for a phone segment. The classifiers in our hybrid systems operate on composite vectors.

For decoding, the segmentation information is obtained from a baseline HMM system — a cross-word triphone system with 8 Gaussian mixtures per state. Composite vectors are generated for each of the segments and posterior probabilities are hypothesized that are used to find the best word sequence using the Viterbi decoder. The HMM system also outputs a set of N-best hypotheses. The posterior probabilities for each hypothesis are determined and the most likely entry of the N-best list is produced.

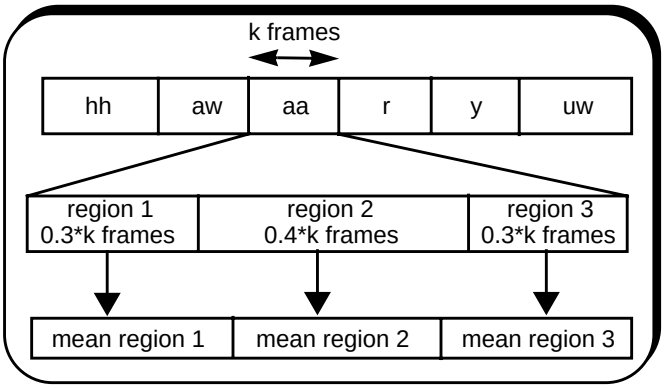

Figure 1: Example of a composite vector construction using a 3-4-3 proportion

4. RESULTS

An initial comparison of the RVM and SVM classifier was carried out on a static vowel classification task, the Deterding vowel data [6]. In this evaluation, the speech data was collected at a 10 kHz sampling rate and low pass filtered at 4.7 kHz. The signal was then transformed to 10 log-area parameters, giving a 10-dimensional input space. A window duration of 50 msec was used for generating the features. The training set consisted of 528 observations from eight speakers, while the test set consisted of 462 observations from a different set of seven speakers. The speech data consisted of 11 vowels uttered by each speaker in a h*d context. The small training set and significant confusion in the vowel data make this data set a very challenging task.

Table 1 compares the RVM and SVM performance on the vowel classification task. Importantly, the RVM classifiers achieve superior performance to the SVM classifiers while utilizing nearly an order of magnitude fewer parameters. While we do not expect the superior error performance to be typical (on pure classification tasks) we do expect the superior sparseness to be typical. This sparseness property will be particularly

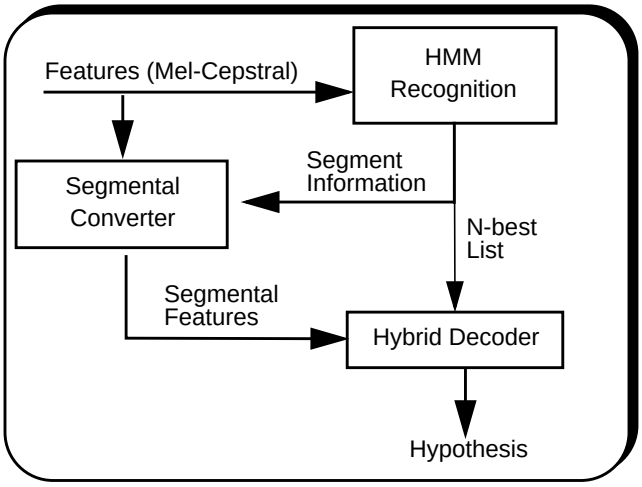

Figure 2: Hybrid system architecture

Approach	Error Rate	Parameter Count
SVM	35%	83 SVs
RVM	30%	13 RVs

Table 1: Comparison of the SVM and RVM systems on a vowel classification task. Both systems outperform other nonlinear classifier approaches. The RVM is able to achieve a lower error rate with much fewer parameters.

important when attempting to build systems which are practical to train and test.

The hybrid SVM and RVM systems have been benchmarked on the OGI alphadigit corpus with a vocabulary of 36 words [7]. A total of 29 phone models, one classifier per model, were used to cover the pronunciations. Each classifier was trained using the segmental features derived from 39-dimensional frame-level feature vectors comprised of 12 cepstral coefficients, energy, delta and acceleration coefficients. The full training set has as many as 30k training examples per classifier. However, the training routines employed for the RVM models are unable to utilize such a large set. The training set was, thus, reduced to 2000 training examples per classifier (1000 in-class and 1000 out-of-class). The test set was an open-loop speaker independent set with 3329 sentences. The composite vectors are also normalized to the range (-1,1) to assist in convergence of the SVM classifiers. Again, we see in Table 2 that the RVM model outperforms (slightly) the SVM system while using much fewer parameters. The training time remains an issue in producing a viable RVM system.

5. SUMMARY

In this work we compared SVM and RVM classifiers in a hybrid speech recognition framework. RVMs have been shown to yield comparable performance with far fewer parameters. However, the size of tasks computable by the RVM is limited due to the complexity of the training process. We are in the process of defining iterative methods which can be used to train the RVM on the very large datasets that are common to speech recognition. Further research is also being directed toward a fully integrated RVM

Approach	Error Rate (%)	No. of weights	Training Time (hrs)	Testing Time (mins)
SVM	16.4	**257 SVs**	0.5	30
RVM	16.2	**12 RVs**	720	1

Table 2: Comparison of the SVM and RVM systems on a 2000-example per phone subset of the OGI Alphadigits data. While the resultant RVM model is extremely sparse, the training time is prohibitive for larger sets.

speech recognition system which does not depend on a Gaussian-based HMM system for generating segmental data or N-best lists.

6. REFERENCES

[1] A. Ganapathiraju, *Support Vector Machines for Speech Recognition*, Ph.D. Dissertation, Mississippi State University, Mississippi State, Mississippi, USA, 2002.

[2] M. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," Journal of Machine Learning, vol. 1, pp. 211-244, June 2001.

[3] D. J. C. MacKay, “Probable networks and plausible predictions --- a review of practical Bayesian methods for supervised neural networks,” Network: Computation in Neural Systems, 6, pp. 469-505, 1995.

[4] V.N. Vapnik, Statistical Learning Theory, John Wiley, New York, NY, USA, 1998.

[5] E. Osuna, R. Freund, and F. Girosi, “Support Vector Machines: Training and Applications,” MIT AI Memo 1602, March 1997.

[6] D. Deterding, M. Niranjan and A. J. Robinson, "Vowel Recognition (Deterding data)," Available at http://www.ics.uci.edu/pub/machine-learning-databases/undocumented/connectionist-bench/vowel, 2000.

[7] R. Cole, "Alphadigit Corpus v1.0". http://www.cse.ogi.edu/CSLU/corpora/alphadigit, Center for Spoken Language Understanding, Oregon Graduate Institute, USA, 1998.