Hybrid models for automatic speech recognition: a comparison of classical ANN and kernel based methods

Ana I. García-Moral, Rubén Solera-Ureña, Carmen Peláez-Moreno and Fernando Díaz-de-María

Department of Signal Theory and Communicationos
University Carlos III Madrid, Leganés (Madrid), Spain
rsolera@tsc.uc3m.es

Abstract

Support Vector Machines (SVM) are state-of-the-art methods for machine learning but share with more classical Artificial Neural Networks (ANN) the difficulty of their application to temporally variable input patterns. This is the case in Automatic Speech Recognition (ASR). In this paper we have recalled the solutions provided in the past for ANN and applied them to SVMs performing a comparison between them. Preliminary results show a similar behaviour which results encouraging if we take into account the novelty of the SVM systems in comparison with classical ANNs. The envisioned ways of improvement are outlined in the paper.

1. Introduction

Hidden Markov Models (HMMs) have become the most employed core technique for Automatic Speech Recognition (ASR). After several decades of intense research work in the field, it seems that the HMM ASR systems are very close to reach their limit of performance. Some alternative approaches, most of them based on Artificial Neural Networks (ANNs), were proposed during the late eighties and early nineties. Among them, it is worth to draw out attention to hybrid HMM/ANN systems (see [1] for an overview), since the reported results were comparable or even slightly superior to those achieved by HMMs.

On the other hand, during the last decade, a new tool appeared in the field of machine learning that has proved to be able to cope with hard classification problems in several fields of application: the Support Vector Machines (SVMs) [2]. The SVMs are effective discriminative classifiers with several outstanding characteristics, namely: their solution is that with maximum margin; they are capable to deal with samples of a very high dimensionality; and their convergence to the minimum of the associated cost function is guaranteed.

Nevertheless, it seems clear that the application of these kernel-based machines to the ASR problem is not straightforward. In our opinion, the main difficulties to be overcome are three: 1) SVMs are originally static classifiers and have to be adapted to deal with the variability of duration of speech utterances; 2) the SVMs were originally formulated as a binary classifier while the ASR problem is multiclass; and 3) current SVM training algorithms are not able to manage the huge databases typically used in ASR. In order to cope with these difficulties, some researchers have suggested hybrid SVM/HMM systems [3, 4], that notably resemble the previous hybrid ANN/HMM systems ([5]). In this paper we comparatively describe both types of hybrid systems (SVM/ and ANN/HMM), highlighting both their common fundamentals and their special characteristics with the aim of also conducting an experimental performance comparison for both clean and noisy speech recognition tasks.

2. Hybrid systems for ASR

As a result of the difficulties found in the application of ANN to speech recognition, mostly motivated by the temporal variability of the speech instances corresponding to the same class, a variety of different architectures and novel training algorithms that combined both HMM with ANNs were proposed in the late 80's and 90's. For a comprehensive survey of these techniques see [1]. In this paper, we have focused on those that employ ANNs (and SVMs) to estimate the HMM state posterior probabilities proposed by Bourlard and Morgan ([5, 6]).

The starting point for this approach is the well-know property of using feed-forward networks such as multi-layer perceptrons (MLPs) of estimating a posteriori probabilities given two conditions:

1. There must be enough number of parameters to train a good approximation between the input and output layers and
2. A global error minimum criterion must be used to train the network (for example, mean square error or relative entropy).

The fundamental advantage of this approach is that it introduces a discriminative technique (ANN) into HMM (generative systems) while retaining their ability to handle the temporal variability.

However, this original formulation had to be modified to estimate the true emission (likelihood) probabilities by applying Bayes' rule. Therefore, the a posteriori probabilities output should be normalized by the class priors to obtain what is called *scaled likelihoods*. This fact was further reinforced by posterior theoretical developments in the search of a global ANN optimization procedure (see [7]).

Thus, systems of this type keep being locally discriminant given that the ANN was trained to estimate a posteriori probabilities. However, it can also be shown that, in theory, HMMs can be trained using local posterior probabilities as emission probabilities, resulting in models that are both locally and globally discriminant but the problem is that there are generally mismatches between the prior class probabilities implicit to the training data and the priors that are implicit to the lexical and syntactic models that are used in recognition. In spite of this,

some results imply that for certain cases the division by the priors is not necessary [7].

Among the advantages of using hybrid approaches we can cite the following (from [7]):

- Model accuracy: both MLP and SVM have more flexibility to provide more accurate acoustic models including the possibility of including different combinations of features as well as different sizes of context.
- Local discrimination ability (at a frame level).
- Parsimonious use of parameters: all the classes share the same ANN parameters.
- Complementarity: since the combination of results from standard HMM systems have been proved to provide better results

3. Experimental Setup

3.1. Database

We have used the well-known SpeechDat Spanish database [8] for the fixed telephone network. This database comprises recordings from 4000 Spanish speakers recorded at 8 kHz over the fixed PSTN using an E-1 interface, in a noiseless office environment.

In our experiments we have used a large vocabulary (more than 24000 words) continuous speech recognition database. The training set contains approximately 100 hours of voice from 3496 speakers (71000 utterances). The callers spoke 40 items whose contents are varied, comprising isolated and connected digits, natural numbers, spellings, city and company names, common applications words, phonetically rich sentences, etc. Most items are read, some are spontaneously spoken. The test set, corresponding to a connected digits task, contains approximately 2122 utterances and 19855 digits (3 hours) from 315 different speakers.

3.2. Parameterization

In our preliminary experiments we have used the classical parameterization based on 12 MFCCs (Mel-Frequency Cepstral Coefficients) plus energy, and the first and second derivatives. These MFCCs are computed every 10 ms using a temporal windows of 25 ms. Thus, the resulting feature vectors have 39 components. In this work, we have considered two different kinds of normalization for the features.

The first normalization considered was a per utterance normalization, that is, every parameter is normalized in mean and variance according to the following expression:

$$\hat{x}_i[n] = \frac{x_i[n] - \mu_f}{\sigma_f + \theta}, \quad (1)$$

where $x_i[n]$ represents the i^{th} component of the feature vector corresponding to frame n, μ_f is the estimated mean from the whole utterance, σ_f is the estimated standard deviation, and θ is a constant just to avoid numerical problems (for our experiments, we have chosen $\theta = 10$).

Thus, per utterance normalization will be more appropriate in the case of noisy environments where test and training conditions do not match. Nevertheless, when we work in a noiseless environment, the second normalization we consider provides better performance like we explain in following sections. This normalization consist of a global normalization, that is, we compute the global mean and variance for all the parameterization utterances in the training set, and finally each parameter normalized in mean and variance according to the next expression:

$$\hat{x}_i[n] = \frac{x_i[n] - \mu}{\sigma}, \quad (2)$$

where $x_i[n]$ represents the i^{th} component of the feature vector corresponding to frame n, μ is the estimated mean from all the utterances in the training set and σ is the estimated global standard deviation.

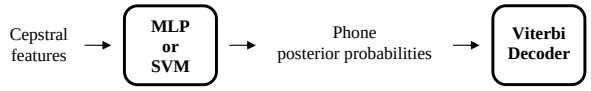

Figure 1: *The whole hybrid recognition system. First, initial phone evidences are estimated using MLPs or SVMs, then these evidences are integrated as local scores for decoding.*

3.3. Baseline experiment with HMMs

Our reference result is the recognition rate achieved by an left-to-right HMM-based recognition system. We use 18 context-dependent phones with 3 states per phone. Emission probabilities for each state were modelled by a mixture of 16 Gaussians, as described in [8].

For this paper, we have partitioned every phone into three segments and obtained a segmentation of the database by performing a forced alignment with this HMM baseline experiment considering each segment delimited by the state transitions of this system (see [4]).

3.4. Experiments with Hybrid Recognition Systems

In this work we consider two different hybrid recognition systems, an ANN/HMM system and a SVM/HMM one. Both of them use a Viterbi decoder using posterior probabilities as local scores as discussed in 2.

The whole hybrid recognition system is composed of two stages shown in Figure 1. The first stage estimates initial evidences for phones in the form of posterior probabilities using an MLP or an SVM. The second stage is a classical Viterbi decoder where we replace the likelihoods estimates provided by the reference HMM-based recognition system by the posteriors estimates obtained in the first stage.

While the reference HMM-based recognition system uses the whole training data set (71000 utterances), the hybrid systems (SVM- and ANN-based recognition systems) only use a small portion of the available training data, due to a practical limitation respect to the number of training samples that the SVM software can consider. Therefore, we have considered useful to evaluate the evolution of the accuracy of each system performing incremental tests using balanced subsets of the available training data (equal number of frames per phone, randomly selected from the whole training set), between 250 and 20000 frames per phone.

3.4.1. Experiments with SVMs

In this case, a multiclass SVM (using the *1-vs-1* approach) is used to estimate posterior probabilities for each frame using

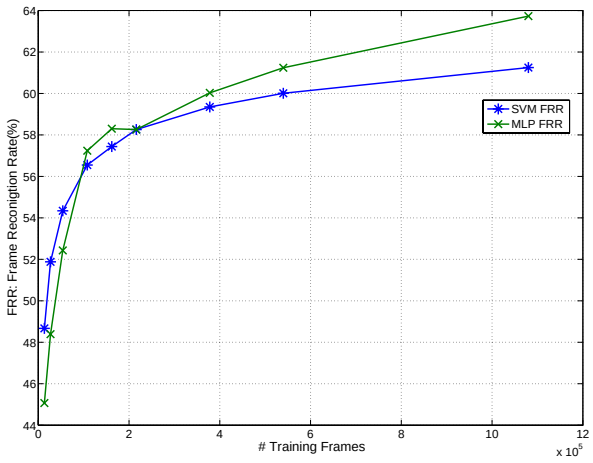

Figure 2: *Frame recognition rate of SVMs and ANNs.*

Platt's approximation ([9]). The SVM uses a RBF (Radial Basis Function) kernel whose parameter, the standard deviation σ, must be tuned by means of a cross-validation process, as well as a parameter C, which establishes a compromise between error minimization and generalization capability in the SVM. The values we have used in our experiments are $C = 256$ and $\sigma = 0.007812$ [4].

3.4.2. Experiments with ANNs

Posterior probabilities used by the Viterbi decoder are now obtained using a MLP trained on a smaller version of the training set, as we mentioned before. The MLP has one hidden layer with 1800 units. MFCC features jointly with energy, delta and acceleration features are used as inputs. There are 54 output units, each of them corresponding to a different part of phone, as we described in section 3.4. The MLP is trained using the relative entropy criterion and the back propagation factor μ was experimentally fixed at 0.14.

4. Preliminary Results and Discussion

This section is devoted to the presentation and discussion of the results obtained by the systems described in the previous section.

Preliminary experiments show a similar behaviour of both SVMs and ANNs at a frame classification level. For the first data normalization method presented in section 3, we observe little differences between SVMs and ANNs. Also, we can see in figure 2 that better results are achieved when more samples are added to the training database, up to a final recognition rate around 61% obtained for the maximum number of input samples our SVM-based system can handle (1080000, 20000 frames per phone).

We have noticed that this first normalization method presents a problem: delta and acceleration coefficients do not have unitary variance. This is due to the constant θ added to the standard deviation in (1). The value used in the experiments (10) is not comparable with the standard deviation of the data and it results in a excessive normalization. This is a problem for the SVM and ANN-based systems. For the first case, the SVM employs a RBF kernel with the same variance for all dimensions, while the training data present different variances for each component (or, at least, for the static, delta and acceleration coefficients). For the latter, this may cause to start in a point far from the solution and, as a consequence, to slow down the convergence of the algorithm. This has led us to apply a second normalization stage to the database, in order to get a unitary variance for all the components of the training data. Some experiments show an important improvement of the previous results (around 4.5%).

Preliminary experiments at word and sentence levels show results are comparable with respect to those of the standard HMM-based speech recognition system used as a baseline. These results are specially promising due to the fact that SVM and ANN-based systems are trained using a maximum of only 3.04% of the available data samples, whereas HMMs are trained using the entire database. This limit is imposed by the SVM software used in the experiments [10], which requieres to mantain the kernel matrix in memory.

In addition, as we have stated in section 2, both SVMs and ANNs provide posteriors to the Viterbi decoder, whereas what we really need and HMMs compute are likelihoods. We think that the hybrid methods might benefit from the use of likelihoods instead of posteriors [5], just by dividing them by the *a priori* probabilities.

Finally, one of the major drawbacks of current HMM-based automatic speech recognition systems is its poor robustness against noisy conditions. During the last years, several techniques aimed at increasing the performance of these systems have been presented, most of them consisting in some preprocessing of the voice signal or modifications of the parameterization stage. From previous experiments ([11]) we suspect that SVM-based systems could provide inherent robust models. Besides, as discussed in section 2, hybrid systems are more amenable for its use with different types of parameterizations that do not comply with the restrictions of independence imposed by HMM. This could result advantageous in the search of robustness.

5. Conclusions

In this paper we have performed a comparison of the accuracy of MLPs and SVMs at a frame level showing a similar performance. However, we still think there is room for improvement of the latter, specially in noisy environment conditions. The maximum margin principle used for its training can make an important difference under those conditions. There are also several issues that should be address as the possibility to incorporate more training samples, the addition of a wider context in the feature vectors, the selection of appropriate feature sets and the computation of further results at a word level.

6. References

[1] E. Trentin and M. Gori, "A survey of hybrid ANN/HMM models for automatic speech recognition," *Neurocomputing*, vol. 37, pp. 91–126, 2001.

[2] B. E. Boser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifiers," in *Computational Learning Theory*, 1992, pp. 144–152. [Online]. Available: citeseer.ist.psu.edu/boser92training.html

[3] A. Ganapathiraju, J. Hamaker, and J. Picone, "Hybrid SVM/HMM architectures for speech recognition," in *Proceedings of the 2000 Speech Transcription Workshop*, vol. 4, Maryland (USA), May 2000, pp. 504–507.

[4] J. Padrell-Sendra, D. Martín-Iglesias, and F. Díaz-de-

María, "Support vector machines for continuous speech recognition," in *Proc. of the 14th European Signal Processing Conference*, 2006.

[5] H. Bourlard and N. Morgan, *Connectionist speech recognition: a hybrid approach*. Norwell, MA (USA): Boston: Kluwer Academic, 1994.

[6] N. Morgan and H. Bourlard, "Continuous speech recognition: An introduction to the hybrid hmm/connectionist approach," *IEEE Signal Processing Magazine*, pp. pp. 25–42, 1995.

[7] H. Bourlard, N. Morgan, C. L. Giles, and M. Gori, *Adaptive Processing of Sequences and Data Structures. International Summer School on Neural Networks 'E.R. Caianiello'. Tutorial Lectures*. Germany; Berlin: Springer-Verlag, 1998, ch. Hybrid HMM/ANN systems for speech recognition: overview and new research directions, pp. 389–417.

[8] A. Moreno, "SpeechDat Spanish database for fixed telephone network," Technical University of Catalonia, Tech. Rep., 1997.

[9] J. C. Platt, *Advances in kernel methods: support vector learning*. Cambridge, MA (USA): MIT Press, 1999, ch. Fast Training of Support Vector Machines Using Sequential Minimal Optimization, pp. 185–208.

[10] C. C. Chang and C. J. Lin, *LIBSVM: a library for support vector machines*, 2001, software available at http://www.csie.ntu.edu.tw/cjlin/∼libsvm.

[11] R. Solera-Ureña, D. Martín-Iglesias, A. Gallardo-Antolín, C. Peláez-Moreno, and F. Díaz-de-María, "Robust asr using support vector machines," *Speech Communication (In press)*, 2007.