Improved Ensemble Training for Hidden Markov Models using Random Relative Node Permutations

Article · March 2003

Source: OAI

2 authors, including:

Brian Carrington Lovell
The University of Queensland
305 PUBLICATIONS   **4,572** CITATIONS

Some of the authors of this publication are also working on these related projects:

Automation of Quantitative Fractography by Advanced Image Processing View project

Digital Immunology: ANA test using HEp-2 cells View project

Improved Ensemble Training for Hidden Markov Models using Random Relative Node Permutations

Richard I. A. Davis and Brian C. Lovell
Intelligent Real-Time Imaging and Sensing Group,
School of Information Technology and Electrical Engineering,
The University of Queensland, Australia, 4072
{riadavis, lovell}@itee.uq.edu.au

Abstract

Hidden Markov Models have many applications in signal processing and pattern recognition, but their convergence-based training algorithms are known to suffer from over-sensitivity to the initial random model choice. This paper focuses upon the use of model averaging, ensemble thresholding, and random relative model permutations for improving average model performance. A method is described which trains by searching for the best relative permutation set for ensemble averaging. This uses the fit to the training set as an indicator. The work provides a simpler alternative to previous permutation-based ensemble averaging methods.

1 Introduction

The work of Davis and Lovell [1] focused upon Hidden Markov Model (HMM) ensemble learning using the well-known Baum-Welch procedure [2, 5, 6] for individual sequences and then averaging the resulting HMM ensemble parameters. It was demonstrated that this is a superior method to the standard method of converging a single model using the multiple sequences simultaneously [4, 2] in performing approximations to sequence distributions, and also for performing classification tasks. Thresholded Winsorization in which the best sub-ensemble is used was also shown to provide further improvements.

This paper investigates the potential for still further improvements on these methods based on searches through random relative permutations of the nodes of the ensemble of models, prior to ensemble parameter averaging. The methods investigated were based on varying the random permutation probability and the number of node transpositions applied to each member of the ensemble.

By relabelling states, it is easy to show that many differ-

ent models can achieve the same probability despite large differences in configuration. The possibility of finding equivalently good models with large structural differences argues against the parameter averaging method to obtain improved models. In other words, it is quite possible that the average of two good models may be a very poor model in the same way that that the point midway between two mountain peaks is quite often a valley. This was the motivation for the method of searching through relative random node permutations.

Levinson, et al. (see [3], appendix B) first proposed that an approximation to bipartite graph matching be used for permutating an ensemble of trained models to achieve a good permutatation match, and good parameter estimates through averaging the permutation-aligned ensemble. This paper demonstrates that simpler randomised methods can be used with good results for node matching when the best method is selected according to the overall fit to the training data set.

2 Permutations of representations of Hidden Markov Models

The focus of this paper is on methods for permuting the alignment of models in an ensemble of models with the same structure.

A hidden Markov model ([2] chapter 6) consists of a set of n nodes or states, each of which is associated with a set of m possible observations (the structure of the model). The parameters of the model include an initial state π which describes the distribution over the initial node set, a transition matrix a_{ij} for the transition probability from node i to node j conditional on node i, and an observation matrix $b_i(O_h)$ for the probability of observing symbol O_h given that the system is in state i. Rabiner uses $\lambda = (A, B, \pi)$ to denote the model parameters.

This paper is based around the method of ensemble aver-
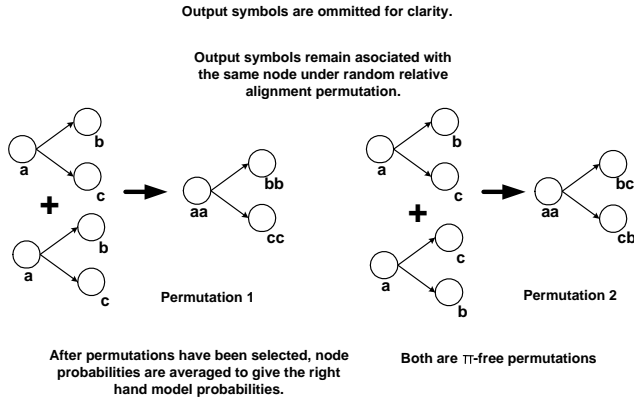
Node alignment in permutation averaging

Output symbols are ommitted for clarity.

Output symbols remain asociated with the same node under random relative alignment permutation.

Permutation 1

Permutation 2

After permutations have been selected, node probabilities are averaged to give the right hand model probabilities.

Both are π-free permutations

Figure 1. Illustration of the permutation averaging method for two different permutations.

aging suggested by Levinson et al. [3] and tested in [1]. The method is as follows. Consider an ensemble of N HMMs $\{\lambda_k = (A, B, \pi)_k; k = 1 \ldots N\}$ which are each trained using the Baum-Welch method to a specific observation sequence in a set of training sequences. Select an alignment of nodes across all N models, and then form a single model λ by averaging the matrix elements of the ensemble, using the selected node alignment (see figure 1).

In this paper we focus upon permutations which only affect those nodes with zero entries in the starting matrix π. These permutations will be termed π-*free permutations*.

Theorem. Random π-free permutations $S = \{S_k\}$ of HMM ensembles $\Lambda = \{\lambda_k\}$ trained using multiple Baum-Welch convergence have the same parameter-averaged model performance as unpermuted ensembles.

Proof. For every model λ_k, partition the set N of nodes into nodes Q with non-zero π values and a subset R with zero π-values. π-free permutations only act on R. Since initialisation of the BW procedure randomly selects π-free nodes, then the Baum-Welch process produces models for which the nodes in R are randomly permuted. Therefore, a permutation of these nodes after convergence will not change the result of the method. ∎

This result means that the insertion of relative random permutations at any stage of the process does not affect the model performance. This paper concentrates on the potential for improvement in the model performance when a good permutation (as measured by its P_{all} score on the training set) rather than a random permutation is applied.

In cases where a permutation might involve a node with a very small starting probability, it is not possible to use the above proof. Permutations involving Q as well as R are less

easy to analyze and will not be tackled here. However many models of interest have a well-defined, small starting node set.

3 Classes of permutations in HMM ensemble averaging

All methods investigated involve the use of the *joint emission probability P_{all}* to quantify the model quality, and also to select the permutation which maximises this function. P_{all} is defined as the product (over all sequences in an ensemble) of the probability that the model generated those sequences individually. This is used both in the training ensemble and in the test ensemble for final evaluation of the method.

The algorithms are all based upon the following set of main design components:

- Winsorization threshold level search based on P_{all}.

- Permutation fraction: the fraction of models in the ensemble being permuted which actually receive a permutation prior to averaging

- Number of transposition: the number of nodes being transposed in a given permutation in each model prior to averaging

- Type of permutation: excluding certain types of permutations from the set being considered, such as permutations involving the starting nodes in π or permutations which would change the transition structure of the resulting average model (from left-right to ergodic, for example)

There are many different ways in which these features may be combined, so we restrict ourselves to those which seem to represent the most important aspects of the ensemble permutation idea.

The following different permutation methods were evaluated:

- **VariableProbPerm** - this method scans through a probability scale from 0 to 1 representing the probability that a given model will be permuted in the ensemble.

- **NumTrans** - this method scans through the number of random transpositions applied to each model. This is limited by the size of the model, as applying too many relative interchange operations per model in the search has no extra benefit.

The ensemble used in both cases is a Windsorised ensemble - with the fraction of models used being determined before these trials were run. Code for training HMMs using these methods is available [7].

4 VariableProbPerm trial

Davis and Lovell [1] gave an empirical study of the following idea (initially suggested by Levinson et al. [3]): In HMM ensemble averaging, because the individual Baum-Welch convergence runs are all initialized to a set of random model parameters (random seeds) then applying another set of random permutations to the models, either before or after training to the observation sequence, will have no effect on the final model formed using parameter averaging.

The next step is to investigate the effect of other, more refined random permutation schemes. A similar strategy to the Winsorization approach will be studied in which a small set of relative permutation sets is compared in terms of the performance of the ensemble-permuted average model on the training data, as measured by P_{all}.

Methodology. In this trial of the VariableProbPerm method, a set of 20 test and 20 training sequences was generated from an initial generating model. The models were randomly generated left-right models with 5 nodes and 4 observation symbols. P_{all} values for the training data were used to compare permutation sets. The best permutation averaged model from those considered was selected. This selected model was then evaluated on the test data. The trial was repeated for 100 initial generating models. Permutations were ensured to be $\pi - free$ as they did not involve the left-right model.

Results. The performance of the variable probability of the small permutation method VariableProbPerm presented in the previous section was investigated and the results are displayed in figure 2. It shows the performance on 20 test sequences of the average of ensembles in which each member of the ensemble is permuted. The method shows steady improvement up to 90% permuted. Clearly there are significant improvements to be obtained by varying the probability of permutation of models in the ensemble and selecting the best fraction.

5 Trial of NumTrans

This trial of the NumTrans method was designed to investigate the best scale of permutations. The major issues of interest were:

1. Is it worth probing the entire range of relative permutations, and

2. Are there any gains to be made by looking at large numbers of transpositions?

Methodology. In this trial of the NumTrans method, a set of 20 test and 20 training sequences was generated from an initial generating model. P_{all} values for training data were used to compare permutation sets. Permutations of
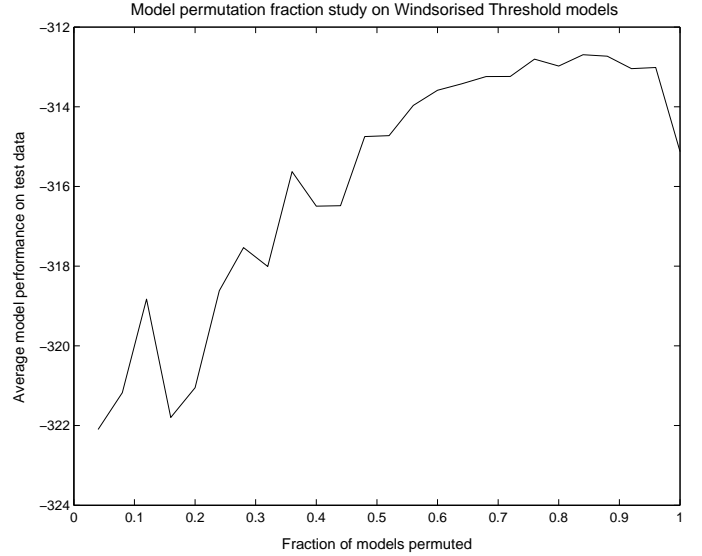

Figure 2. Small permutations are applied to a variable fraction of models in the ensemble.

the ensemble were constructed using a VariableProbPerm method with a 50% probability of permuting a given model. The parameter under investigation was the number of transpositions applied per model, when that model was selected for permutation. The number of transpositions ranged from 1 to 15. It was anticipated that large numbers of transpositions would not show any benefit over small numbers because of the existence of permutation inverses (so a large number of interchange operations is equivalent to its inverse operation, which can consist of a small number of interchange operations).

The best permutation averaged model from those considered was selected. This selected model was then evaluated on the test data. The trial was repeated for 100 initial generating models.

Results. The performance of the variable probability of the small permutation method NumTrans presented in the previous section was investigated and the results are displayed in figure 2. It shows the performance on 20 test sequences of the average of ensembles in which each member of the ensemble is permuted. As can be seen from the results of figure 3 transposition searches with more transpositions can be superior to smaller searches. Importantly, the best performance was found when the number of random transpositions was set to the number of states in the model. There was not a very clear trend in this instance. However we can deduce that applying large numbers of relative transpositions is not particularly helpful. This is due to the following:

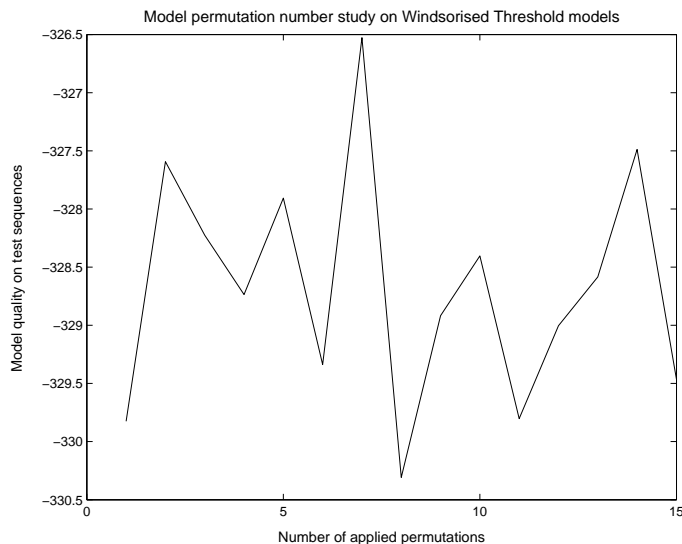1. the models already have highly random relative per-

Figure 3. A variable number of permutations is applied to each member of the ensemble.

mutations due to the random initial seed choice

2. it is counterproductive to apply large numbers of transpositions to all models as it is the relative permutation which matters (so smaller numbers will give superior performance)

3. As the number of transpositions approaches the size of the model, performance will be similar to that in the case of small numbers of transpositions.

That said, searches using more permutations do have a slightly superior search ability. However, unless a high level of computing power is available, it seems best to only use permutations of one or two transpositions, in combination with the VariableProbPerm search method.

6   Conclusions

We have demonstrated that substantial gains in averaging can be made using either of two very simple random permutation alignment schemes, VariableProbPerm and NumTrans. These are simpler alternatives to the more complex scheme proposed by Levinson et al. in [3].

There are clear benefits to be found by applying random permutations of the ensemble, using both methods. A search is necessary to locate the best permutation size. In general, the gains obtainable by varying the probability of permutation are greater than the gains obtainable by varying the number of permutations.

The use of the joint emission probability P_{all} for the training set is a useful indicator in selecting the best permutations.

It may be possible to construct more advanced schemes based upon the findings of this paper. For example, a gradient-descent method in which good permutations are retained as the process continues may be a faster way of locating the best ensemble permutation set.

A modification of this technique to include permutations involving non-zero π vector elements may be possible, but is likely to be more complex. From the success of this method however, it seems a promising avenue for further investigation.

7   Acknowledgements

References

[1] R. I. A. Davis and B. C. Lovell, "Improved Estimation of Hidden Markov Model Parameters from Multiple Observation Sequences", *International Congress on Pattern Recognition*, Quebec City (2002)

[2] L. R. Rabiner, and B. H. Juang, *Fundamentals of Speech Recognition* New Jersey Prentice Hall, 1993.

[3] S. E. Levinson, L. R. Rabiner, and M. M. Sondhi, *An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition*, The Bell System Technical Journal 1035-1074, Vol. 62, No. 4, April 1983.

[4] Xiaolin Li, Marc Parizeau, Réjean Plamondon, "Training Hidden Markov Models with Multiple Observations - A Combinatorial Method". *IEEE Transactions on PAMI*, vol. PAMI-22, no. 4, pp 371-377, April 2000.

[5] D. J. C. Mackay, "Ensemble Learning for Hidden Markov Models", *Technical report*, Cavendish Laboratory, University of Cambridge, 1997.

[6] A. Stolcke and S. Omohundro. "Hidden Markov Model induction by Bayesian model merging." In *NIPS 5*, pages 11-18. 1993.

[7] C Walder, R.I.A. Davis, "IRIS source for estimating Hidden Markov Models".

Available at:

http://www.itee.uq.edu.au/ iris/CVsource/source.html