

[This version is based on our Eurospeech 2003 paper but with updated results on our OBS experiment in Fig.2 which are now the same as the ones shown in our presented poster in the conference. The major difference lies in the fact that in the new experiments, we do not allow unreachable or non-propagating states from happening.]

Pruning Transitions in a Hidden Markov Model with Optimal Brain Surgeon

Brian Mak

Kin-Wah Chan

Hong Kong University of Science & Technology
Department of Computer Science
Clear Water Bay, Hong Kong
mak@cs.ust.hk

Hong Kong University of Science & Technology
Dept. of Electrical & Electronic Engineering
Clear Water Bay, Hong Kong
eeivan@ee.ust.hk

Abstract

This paper concerns about reducing the topology of a hidden Markov model (HMM) for a given task. The purpose is two-fold: (1) to select a good model topology with improved generalization capability; and/or (2) to reduce the model complexity so as to save memory and computation costs. The first goal falls into the active research area of model selection. From the model-theoretic research community, various measures such as Bayesian information criterion, minimum description length, minimum message length have been proposed and used with some success. In this paper, we are considering another approach in which a well-performed HMM, though perhaps oversized, is optimally pruned so that the loss in the model training cost function is minimal. The method is known as Optimal Brain Surgeon (OBS) that has been used in the neural network (NN) community. The application of OBS to NN is a constrained optimization problem; its application to HMM is more involved and it becomes a quadratic programming problem with both equality and inequality constraints. The detailed formulation is presented, and the algorithm is shown effective by an example in which HMM state transitions are pruned. The reduced model also results in better generalization performance on unseen test data.

1. Introduction

In data modeling, one always faces the modeling dilemma: if the model has too many parameters, it runs the risk of over-fitting the training data; but if the model has too few parameters, it may compromise its capability to represent the data distribution. There are at least two approaches for the regularization problem:

- In modeling theory, this is the model selection

problem. Various selection measures such as the Bayesian information criterion (BIC), Akaike information criterion (AIC), minimum description length (MDL), and minimum message length (MML) have been proposed. ([6] is a good review for these measures.) Some successful applications of BIC to select the number of Gaussian mixtures or the model complexity have been reported in speech recognition [3, 2] and handwriting recognition [1].

- A (possibly oversized) model is trained and then pruned to a desirable size according to an optimality criterion. This approach may also be used to reduce the model size for saving memory and computation cost at the expense of small degradation in classification performance.

This paper investigates a method belonging to the second approach, called *Optimal Brain Surgeon* (OBS) [5] to reduce the topology of Hidden Markov model (HMM). HMM has been widely used for representing temporal or spatial data in areas like automatic speech recognition (ASR) or handwriting recognition. However, for example, in ASR, the topology of an HMM such as the number of states and their connectivity is usually pre-set by experience or heuristic. It will be interesting to see if OBS may help decide an optimal HMM topology for a given task.

OBS belongs to a class of pruning methods that make use of second-order derivatives to prune the least “important” weights optimally in a neural network (NN). The method has been shown effective in refining the complex topology of an over-fitted neural network in [7]. When applied to a neural network, OBS is a constrained optimization problem which tries to eliminate a weight

connection in an NN but at the same time re-adjusts all the other weights optimally. When OBS is applied to an HMM, it becomes a quadratic programming problem with equality and inequality constraints.

2. Optimal Brain Surgeon on NN

To use Optimal Brain Surgeon (OBS) [5], a neural network with a set of weights \mathbf{w} is first trained to convergence according to an error function $E(\mathbf{w})$. By using the Taylor's expansion, a change in the error function, δE induced by a change in the weights $\delta\mathbf{w}$ is given by

$$\delta E = \delta\mathbf{w}^T \cdot \frac{\partial E}{\partial \mathbf{w}} + \frac{1}{2} \delta\mathbf{w}^T \cdot \frac{\partial^2 E}{\partial \mathbf{w}^2} \cdot \delta\mathbf{w} + \dots \quad (1)$$

where $\frac{\partial E}{\partial \mathbf{w}}$ is the gradient vector and $\frac{\partial^2 E}{\partial \mathbf{w}^2}$ is the Hessian matrix \mathbf{H} . The first term in Eqn(1) vanishes by the assumption that the network has converged to a local, if not global, minimum of the error function and the gradient is zero. If the third- and all higher-order terms are negligible, then only the second-order derivative term remains and the change in error can be approximated as

$$\delta E = \frac{1}{2} \delta\mathbf{w}^T \cdot \frac{\partial^2 E}{\partial \mathbf{w}^2} \cdot \delta\mathbf{w} . \quad (2)$$

The deletion of a single weight w_j is formulated as the following constraint on Eqn(2)

$$\mathbf{e}_j^T (\delta\mathbf{w} + \mathbf{w}) = 0 \quad (3)$$

where \mathbf{e}_j is a unit vector with the j -th component being 1. The constrained optimization problem can be solved by the standard Lagrangian method.

3. Optimal Brain Surgeon on HMM

The Optimal Brain Surgeon algorithm in the last Section may be modified to prune a state transition in an HMM. As will be shown below, the modification changes a Lagrange optimization problem to a quadratic programming problem.

3.1. Theory

OBS on HMM requires the deletion of a state transition to modify all the remaining transitions optimally so that the *decrease* in log-likelihood of the training data, $\log L$, is minimal *and* all HMM constraints are preserved. Let us arrange all HMM transitions in a vector \mathbf{w} and assume that we would like to delete the j -th transition; that is, $\delta w_j = -w_j$. Based on Eqn(2), OBS on HMM may be formalized as an optimization problem to minimize

$$\delta \log L = \delta\mathbf{w}^T \cdot \frac{\partial^2 \log L}{\partial \mathbf{w}^2} \cdot \delta\mathbf{w} \quad (4)$$

subject to the following constraints:

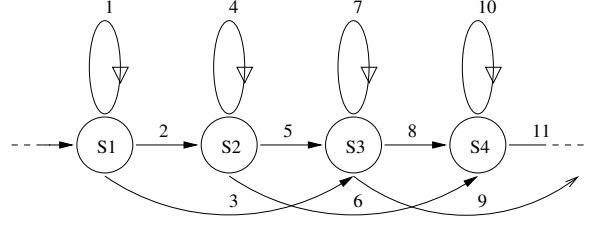


Figure 1: A left-to-right HMM with single-state skips. Only 4 states are shown and all transitions are numbered.

I. Selection Constraint: To prune the j -th state transition, we have

$$\mathbf{e}_j^T (\mathbf{w} + \delta\mathbf{w}) = 0 . \quad (5)$$

II. Sum-of-Transitions Constraint: The sum of probabilities of all out-going transitions from a state must be one. That is,

$$\mathbf{M}^T (\mathbf{w} + \delta\mathbf{w}) = \mathbf{1} \quad \text{or} \quad \mathbf{M}^T \delta\mathbf{w} = 0 \quad (6)$$

where \mathbf{M} is an indicator matrix in which the i -th column vector indicates all valid transitions out of the i -th states by 1 and invalid transitions by 0. For example, if the HMM in Fig.1 has M states and N transitions, and each state has 3 outgoing transitions as shown, then its indicator matrix \mathbf{M} is

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & & & \\ 1 & 0 & 0 & & & \\ 1 & 0 & 0 & & & \\ 0 & 1 & 0 & \cdot & \cdot & \cdot \\ 0 & 1 & 0 & & & \\ 0 & 1 & 0 & & & \\ & & \vdots & & & \end{bmatrix}_{N \times M} . \quad (7)$$

III. Positivity Constraint: All transition probabilities must be positive.

$$\mathbf{w} + \delta\mathbf{w} \geq \mathbf{0} . \quad (8)$$

One may think we need another constraint to make sure all transitions are less than one. However, the positivity constraint together with the sum-of-transitions constraint implicitly imply that already.

It is because of the last inequality constraint that turns the optimization to a quadratic programming problem. In this paper, the quadratic programming problem is solved by the *active set method* [4].

3.2. Algorithm

The whole OBS algorithm is shown in Algorithm 1. It is an iterative procedure that removes one state transition at

a time. The saliency of a transition is defined as the decrease in the log-likelihood of the training data, $\delta \log L$, if the transition is pruned. The algorithm deletes the transition with the maximum saliency in each iteration.

Notice that some transition deletions will render the reduced HMM infeasible — that is, the final state of the HMM cannot be reached. Thus, after each iteration of the OBS algorithm, one has to run a feasibility test and prevents this from happening.

Algorithm 1 Optimal Brain Surgeon algorithm on HMM

- STEP 1.** Train an HMM until it converges.
- STEP 2.** Compute its full Hessian on the training data.
- STEP 3.** Solve the quadratic programming problem of Eqn(4) for deleting each possible transition and record the corresponding saliency $\delta \log L$ and $\delta \mathbf{w}$.
- STEP 4.** Sort the saliencies. If the greatest saliency $\delta \log L$ is less than a threshold \hat{L} , then stop.
- STEP 5.** Delete the transition corresponding to the greatest saliency and update other transition probabilities by the $\delta \mathbf{w}$ result in STEP 3.
- STEP 6.** Repeat STEP 2 – 5 until the maximum number of allowable iterations is exceeded.
-

4. Hessian Calculation

The Hessian matrix, $\mathbf{H} = \frac{\partial^2 \log L}{\partial \mathbf{w}^2}$ is derived from the first principle by differentiating the forward probability in the well-known Forward-Backward algorithm with respect to each state transition probability a_{ij} . Given an observation sequence $O = \{o_1, o_2, \dots, o_T\}$ and an HMM with N states and model parameters λ , we will denote the likelihood of the observation sequence $P(O | \lambda)$ by L , and the log-likelihood by $\log L$. The likelihood may be computed efficiently by the iterative Forward Procedure as summarized in the following formulas:

$$\alpha_s(1) = \pi_s b_s(o_1) \quad (9)$$

$$\alpha_s(t) = \left[\sum_{r=1}^N \alpha_r(t-1) a_{rs} \right] b_s(o_t) \quad (10)$$

$$L = P(O|\lambda) = \sum_{r=1}^N \alpha_r(T) \quad (11)$$

where $\alpha_r(t)$ is the probability of observing the sub-sequence $\{o_1, o_2, \dots, o_t\}$ by the model, ending up in state r at time t ; π_s is the initial probability of state s ; a_{rs} is the probability of transiting from state r to state s ; and $b_s(o_t)$ is the probability of observing o_t at state s .

4.1. Gradient and Hessian in log Domain

A general term in the gradient of the log-likelihood is given by

$$\frac{\partial \log L}{\partial a_{im}} = \frac{1}{L} \cdot \frac{\partial L}{\partial a_{im}}. \quad (12)$$

Similarly, a general term in the Hessian of the log-likelihood is given by

$$\frac{\partial^2 \log L}{\partial a_{im} \partial a_{jn}} = \frac{1}{L} \cdot \frac{\partial^2 L}{\partial a_{im} \partial a_{jn}} - \frac{1}{L^2} \cdot \frac{\partial L}{\partial a_{im}} \cdot \frac{\partial L}{\partial a_{jn}}. \quad (13)$$

4.2. Gradient and Hessian in Linear Domain

The Hessian of the log-likelihood in Eqn(13) requires the calculation of the gradient and Hessian of the likelihood (in linear domain). The latter may be computed using the iterative Forward Procedure by differentiating the forward probabilities given by Eqn(10) as follows:

$$\frac{\partial \alpha_s(t)}{\partial a_{im}} = \sum_{r=1}^N \left[\frac{\partial \alpha_r(t-1)}{\partial a_{im}} a_{rs} + \alpha_r(t-1) \frac{\partial a_{rs}}{\partial a_{im}} \right] b_s(o_t) \quad (14)$$

$$\begin{aligned} \frac{\partial^2 \alpha_s(t)}{\partial a_{im} \partial a_{jn}} = & \sum_{r=1}^N \left[\frac{\partial^2 \alpha_r(t-1)}{\partial a_{im} \partial a_{jn}} a_{rs} + \frac{\partial \alpha_r(t-1)}{\partial a_{im}} \frac{\partial a_{rs}}{\partial a_{jn}} \right. \\ & \left. + \frac{\partial \alpha_r(t-1)}{\partial a_{jn}} \frac{\partial a_{rs}}{\partial a_{im}} \right] b_s(o_t). \quad (15) \end{aligned}$$

Finally, elements of the Hessian matrix in Eqn(13) are given by

$$\frac{\partial L}{\partial a_{im}} = \sum_{r=1}^N \frac{\partial \alpha_r(T)}{\partial a_{im}} \quad (16)$$

$$\text{and } \frac{\partial^2 L}{\partial a_{im} \partial a_{jn}} = \sum_{r=1}^N \frac{\partial^2 \alpha_r(T)}{\partial a_{im} \partial a_{jn}}. \quad (17)$$

5. Experimental Evaluation

The adult data set of TIDIGITS is used for evaluation. It consists of 8623 training utterances and 8700 testing utterances. Whole digit HMMs are trained. Each digit HMM has 16 states with 16 Gaussian components per state. The HMM topology is shown in Fig.1. Compared with the commonly used HMM topology which is strictly left-to-right and each state only has two outgoing transitions — from state i to state i and $i + 1$, our topology is more complex with an additional skip-transition for each state — from state i to state $i + 2$. The transition arcs are numbered as follows:

- self-transitions are numbered as 1, 4, 7, ..., 46.
- next-transitions are numbered as 2, 5, 8, ..., 47.
- skip-transitions are numbered as 3, 6, 9, ..., 48.

The acoustic vector is the conventional 39-dimensional cepstral vector containing 12 MFCCs and normalized energy plus their first- and second-order derivatives.

5.1. Saliency Evaluation

Eleven digit models, a noise model, and a short pause model are trained by the EM algorithm until convergence. Then OBS in Algorithm 1 is used to compute the saliency, $\delta \log L$ of each transition for each digit HMM and the transition with the greatest saliency is deleted from each HMM. The greatest five saliencies and their corresponding transition arcs for the HMM of digit “zero” are shown in Table 1.

Table 1: Choice of transition deletion determined by OBS and manual pruning for the HMM of digit “zero”.

Rank	OBS		Manual Pruning	
	Arc#	Saliency	Arc#	Saliency
1st	12	-3.53e-36	12	-9.11e-02
2nd	18	-3.89e+00	18	-1.98e+01
3rd	39	-1.04e+01	3	-1.10e+02
4th	3	-1.19e+01	39	-1.13e+02
5th	33	-1.73e+01	33	-1.23e+02

To gauge our results, we also try to delete each transition *manually* from each HMM, re-normalize the transition probabilities of the affected state, and compute the corresponding saliency. The results are also shown in Table 1. Both OBS and manual pruning method suggest to delete the transition arc numbered 12, which is the skip transition from the 4th state to the 6th state. The two methods also agree well in their top 5 recommendations. However, one should remind that the manual pruning method is not optimal because the remaining transitions are not optimally re-adjusted as in OBS.

5.2. Recognition Evaluation

Although the major objective of OBS is to reduce the complexity of an HMM optimally, the recognition performance is also our concern. In this experiment, the OBS algorithm is iterated and during each iteration, one transition is deleted from *each* HMM. The word accuracies of the reduced HMMs after each iteration are plotted in Fig.2. We observe that OBS not only reduces the topology of the digit HMMs but also improves their generalization performance on unseen test data even after 15 iterations. The generalization peaks at the 11-th iteration and the recognition accuracy improves from the original 99.2% to 99.4% — an error reduction of 25% which is statistically significant at the 0.05 confidence level. Again, OBS gives better performance than manual pruning. Another interesting observation is that although we may expect the skip-transitions to be the least important, not all deleted transitions are skip-transitions. In fact, only 70% of deleted transitions are skip-transitions after 16 iterations of OBS.

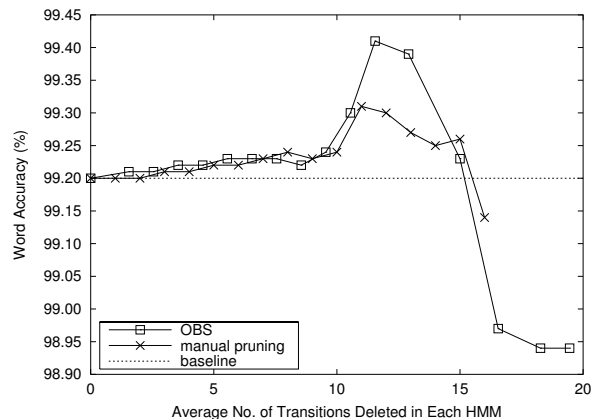


Figure 2: Recognition performance of HMMs after transition pruning by OBS.

6. Conclusions

In this paper, we have adopted the theory of OBS in neural network to prune state transitions in an HMM successfully. Experimentally we also find that the generalization performance of pruned models is also improved.

In the future, we would like to extend the current work in two directions. Firstly, although we delete only one transition at a time in this paper, it can be easily extended to prune several transitions simultaneously. As a result, even a whole HMM state may be pruned optimally under the OBS framework. Secondly, the evaluation experiment in this paper uses a relatively simple HMM. It will be interesting to apply the OBS algorithm to more complex HMMs, such as product HMMs that are commonly used in multi-band ASR and audio-visual ASR.

7. Acknowledgements

This work is supported by the Hong Kong Research Grants Council under the grant number DAG00/01.EG09.

8. References

- [1] A. Biem, J. Y. Ha, and J. Subrahmonia. A Bayesian Model Selection Criterion for HMM Topology Optimization. In *Proc. of ICASSP*, vol. I, pp. 989–992, 2002.
- [2] Y. C. Chan, M. Siu, and B. Mak. Pruning of State-Tying Tree using Bayesian Information Criterion with Multiple Mixtures. In *Proc. of ICSLP*, vol. IV, pp. 294–297, 2000.
- [3] S. S. Chen and P. S. Gopalakrishnan. Clustering via the Bayesian Information Criterion with Applications in Speech Recognition. In *Proc. of ICASSP*, pp. 645–648, 1998.
- [4] R. Fletcher. *Practical Methods of Optimization*, chapter 12, pp. 277–330. The Art of Computer Programming. John Wiley & Sons, Reading, Massachusetts, 2nd ed., March 1990.
- [5] Babak Hassibi and David G. Stork. Second order derivatives for network pruning: Optimal brain surgeon. In Stephen José Hanson, Jack D. Cowan, and C. Lee Giles,

editors, *Advances in Neural Info. Proc. Sys.*, vol. 5, pp. 164–171. Morgan Kaufmann, San Mateo, CA, 1993.

- [6] A. D. Lanterman. Schwarz, Wallace, and Rissanen: Intertwining themes in theories of model selection. *International Statistical Review*, August 2001.
- [7] Thomas Ragg, Heinrich Braun, and Heiko Landsberg. A comparative study of neural network optimization techniques. In *Proc. of ICANNGA*. Springer-Verlag, 1997.