GMM AND HMM TRAINING BY AGGREGATED EM ALGORITHM WITH INCREASED ENSEMBLE SIZES FOR ROBUST PARAMETER ESTIMATION

Takahiro Shinozaki[*], *Tatsuya Kawahara*

Academic Center for Computing and Media Studies, Kyoto University, Kyoto, Japan

ABSTRACT

In order to compensate for the weaknesses of the expectation maximization (EM) algorithm to over-training and to improve model performance for new data, we have recently proposed aggregated EM (Ag-EM) algorithm that introduces bagging-like approach in the framework of the EM algorithm and have shown that it gives similar improvements as cross-validation EM (CV-EM) over conventional EM. However, a limitation with the experiments was that the number of multiple models used in the aggregation operation or the ensemble size was fixed to a small value. Here, we investigate the relationship between the ensemble size and the performance as well as giving a theoretical discussion with the order of the computational cost. The algorithm is first analyzed using simulated data and then applied to large vocabulary speech recognition on oral presentations. Both of these experiments show that Ag-EM outperforms CV-EM by using larger ensemble sizes.

Index Terms— Expectation maximization algorithm, ensemble training, bagging, sufficient statistics, hidden Markov model

1. INTRODUCTION

Expectation maximization (EM) algorithm is an iterative algorithm consisting of expectation step (E-step) and maximization step (M-step), and is widely used for model training with hidden variables. In the E-step, probabilistic distributions of hidden variables are inferred and expected log likelihood is estimated given an initial model. This process effectively converts the incomplete data to complete data or, more precisely, computes expected sufficient statistics for the incomplete data. Then, the maximum likelihood method is applied using the expected sufficient statistics in the M-step and model parameters are updated.

A restriction with the algorithm is the weakness for over-fitting to the training data. Although it is guaranteed for the EM algorithm that it monotonically increases the training set likelihood for the training iterations, it does not hold for new data. In practice, it is observed that the likelihood for new data begins to decrease after several iterations especially when the amount of training data is small compared to the number of model parameters, because the parameters are specialized too much to the training data and the model loses generality.

Furthermore, depending on the model structures, the EM algorithm can be even unstable. For example, a two-mixture Gaussian distribution gives arbitrarily large likelihood for training data if one of the Gaussians covers a particular data point with very small variance and the other Gaussian spans the rest of the data points. Obviously, such a model is not desirable, as it does not generalize to new data. This type of problem is in fact often observed during GMM training by the EM algorithm.

These problems are originated from optimistic bias in the likelihood estimation in the E-step. Because the model estimated in a M-step is used in the next E-step, and the E and M-steps are alternately applied on the same data, it forms a vicious spiral and the bias is reinforced.

To compensate for this problem, we had proposed cross-validation EM (CV-EM) algorithm and had demonstrated its superiority over conventional EM [1]. The idea of the algorithm was to reduce the bias by efficiently separating data used in the E-step and the M-step. An alternative way of avoiding the bias is to incorporate bagging-like approach in the E-step instead of CV. Based on this idea, we have recently proposed aggregated EM (Ag-EM) algorithm and have shown by speech recognition experiments using broadcast news data that similar improvements as the CV-EM algorithm are obtained [2]. However, a limitation with the experiments was that the ensemble size, which is the number of multiple models used to obtain the aggregated expected sufficient statistics, was fixed to a small value (i.e., three).

In this paper, we investigate the relationship between the ensemble size and the model performance, and show that Ag-EM actually outperforms CV-EM by using increased ensemble sizes. We also give theoretical discussion with the order of the computational cost of the Ag-EM algorithm.

The rest of the paper is organized as follows. Section 2 reviews the Ag-EM algorithm and gives the computational cost. Section 3 shows experiments with GMM training using simulated data. Section 4 applies the algorithm to HMM training for continuous speech recognition on oral presentations. Finally, conclusions are given in Section 5.

[*] Takahiro Shinozaki has now moved to Tokyo Institute of Technology, Tokyo, Japan. Email: shinot@furui.cs.titech.ac.jp

1-4244-1484-9/08/$25.00 ©2008 IEEE

2. AGGREGATED EM (AG-EM) ALGORITHM

In this section, we first review the Ag-EM training procedure. The details of the procedure can be found in [2]. Then, it is newly discussed about the order of the computational cost and the choice of the hyper training parameters.

2.1. Training procedure

Figure 3 shows the procedure of the Ag-EM algorithm. The procedure is similar to the parallel EM training [3] shown in Figure 1, which is used to shorten the turn-around time of the training, in that it partitions the training set and computes sufficient statistics for each subset. However, for Ag-EM, the partitioning is not just for the parallelization but has more essential role. It utilizes sufficient statistics both as a target of the aggregation and as a means for efficient processing.

Specifically, the first E-step is identical to the parallel EM algorithm, and K sufficient statistics files are computed for the subsets. Then, instead of making a single model by accumulating all the sufficient statistics, N different models are generated in the M-step using $K' < K$ of the subsets chosen without replacement. In the next E-step, the same subset is repeatedly processed by the N models and the resulting N sufficient statistics are averaged to make more robust estimation of the expected sufficient statistics. We refer to N as the ensemble size of this algorithm. The process is repeated as EM and the final model is obtained by merging all the sufficient statistics.

Compared to the CV-EM training procedure shown in Figure 2, Ag-EM allows overlap in data between the E-step and the M-step. Instead, it avoids the over-fitting by aggregating the expected sufficient statistics. In addition, it is expected that it can find better local optima as multiple models are used to estimate the expected sufficient statistics.

2.2. Computational cost

If the subset-wise sufficient statistics are not used and the N models are trained independently, the computational cost for the E-step is $O\left(\frac{K'}{K}TN^2\right)$, where T is the training set size. This is because each model is estimated from $\frac{K'}{K}T$ of the training data using N different models and the estimation is repeated for N models at each training iteration. However, because there are overlaps between the training sets of the N models, the computation is redundant. Ag-EM removes the redundancy by first estimating the sufficient statistics for the exclusive subsets and then generating the models by accumulating them. In this way, the computational cost of Ag-EM is $O(TN)$, which is linear in N. If $K' = K$ and $N = 1$, then Ag-EM reduces to the parallel EM training and hence the computational cost becomes the same as (parallel) EM and CV-EM.

The storage requirement is mostly determined by the collection of sufficient statistics files and is linear in K. Because Ag-EM incorporates the aggregation into the iterative parameter estimation and the output is a single model, it does not increase the decoding cost as opposed to the bagging training.

2.3. On parameter settings

Because the improvement by Ag-EM is obtained through aggregating multiple models, these models need to be moderately different each other. On the other hand, if these models are too different, then the averaging operation on the sufficient statistics does not make sense since the correspondences between the hidden variables over the multiple models are lost. The degree of the similarity between the multiple models are determined by $\frac{K'}{K}$. In the following experiments, $\frac{K'}{K}$ was set to 0.6 based on a preliminary experiment.

Similar to the bagging training, the performance of Ag-EM depends on the number of the multiple models or the ensemble size N. Basically, the larger the ensemble size, the better the performance. We experimentally show the relationship in the following sections.

3. EXPERIMENTS WITH GAUSSIAN MIXTURE MODELS

3.1. Experimental setups

Experiments were performed using 4-dimensional 8-mixture Gaussian distributions as random population distributions whose component diagonal Gaussians and weights were randomly defined. The training and the test data were independently sampled from the GMM. GMMs with 8 mixture components were trained by first initializing their parameters using a global mean and variance and then applying the EM, CV-EM, or Ag-EM algorithm. The performance of the models was evaluated by likelihood calculated for the test set with 1000 samples. To eliminate the randomness from the results, the experiments were repeated 10 times for each training condition using data sampled from the different random population distributions and their likelihood was averaged. A common variance floor was used (10^{-5}) in all of the training methods.

3.2. Experimental results

Figure 4 shows the test set likelihood of the GMMs trained by using 20 and 80 training samples. The horizontal axis is the number of training iterations. The zeroth iteration means the likelihood was evaluated using the initial model. The number of subsets K was 20 for both CV-EM and Ag-EM. For Ag-EM, the ensemble size N was 8 and the number of subset selection K' was chosen to 12 so that $\frac{K'}{K} = 0.6$.

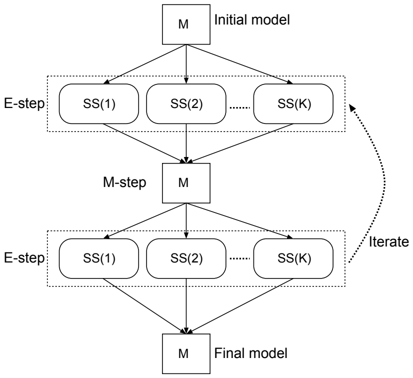

Fig. 1. Parallel EM training. SS(i) denotes the sufficient statistics for the i-th data subset.

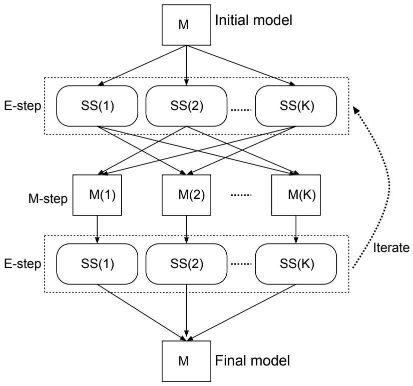

Fig. 2. CV-EM training. M(i) denotes the i-th CV model estimated without using the i-th data subset.

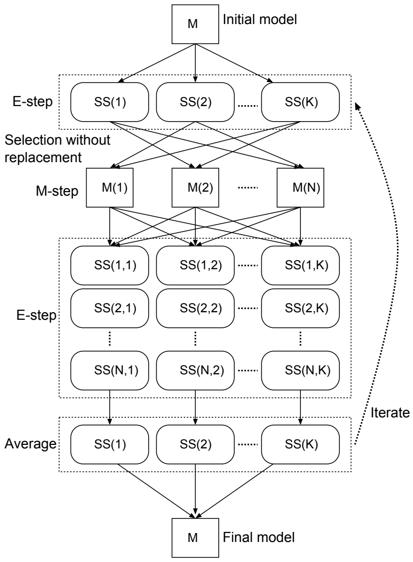

Fig. 3. Aggregated EM-algorithm. SS(i,j) denotes the sufficient statistics for j-th data subset by the i-th model.

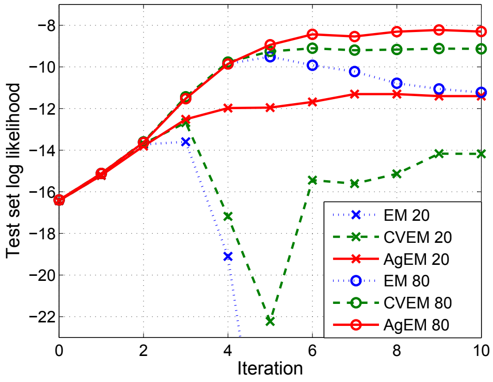

Fig. 4. Test set likelihood of GMMs trained by EM, CV-EM and Ag-EM with 20 and 80 training samples.

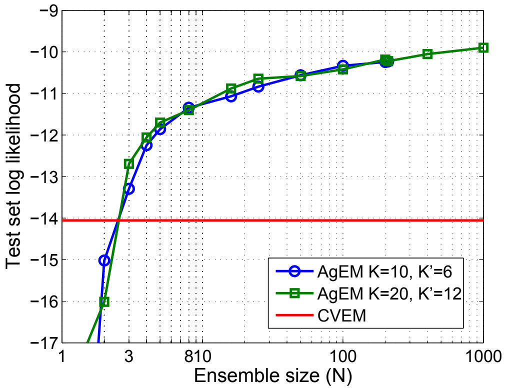

Fig. 5. Ensemble size and model performance.

As can be seen in the figure, the test set likelihood by the EM algorithm is not monotonic with the number of iterations. It increases in the beginning, but falls after several iterations. This is because the model parameters are over-fitted to the training data. The drop is especially large when the training data is small relative to the number of parameters. Ag-EM is much more robust to the over-fitting than EM and CV-EM, and gives higher likelihood than these methods. In the other words, Ag-EM has a potential to accurately train more complex and precise model than EM and CV-EM given the same amount of training data.

Figure 5 shows the model performance for the ensemble size N. The number of training samples was 20 and the number of training iterations was 10. The number of subsets K was set to 10 and 20. The results in Figure 4 at 10th iteration correspond to the likelihood at $N = 8$ in this figure. For the purpose of comparison, the likelihood by CV-EM, which is independent of N, is also shown in the figure.

As can be seen, the performance of Ag-EM increases for the ensemble size N. While Ag-EM gave poor performance when N was small, it outperformed CV-EM when N was larger than three[1]. It can also be seen that Ag-EM gave similar performance for different K as far as $\frac{K'}{K}$ was the same. Although, if K is too small, the choice of N is restricted as the maximum value is determined by the number of combinations of choosing K' out of K.

[1] This is consistent with our previous results in [2] that CV-EM and Ag-EM gave similar performance since N was fixed to three in the experiments.

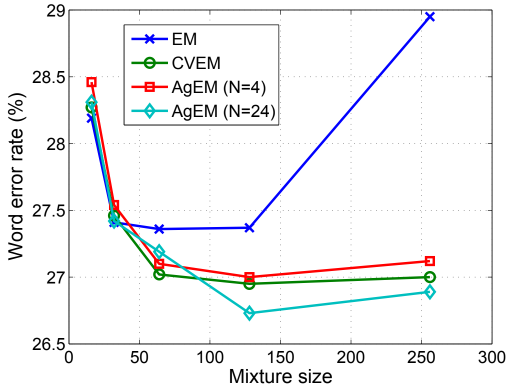

Fig. 6. Word error rates of oral presentation speech recognition with 30 hours of CSJ training data.

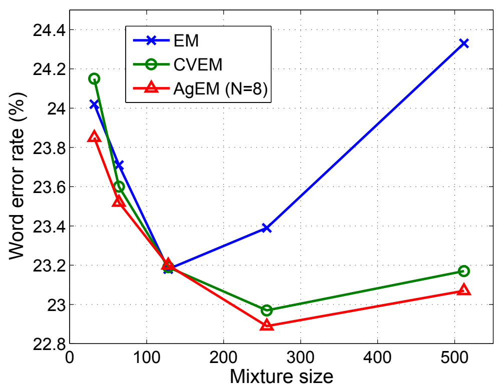

Fig. 7. Word error rates of oral presentation speech recognition with 254 hours of CSJ training data.

4. SPEECH RECOGNITION EXPERIMENTS

4.1. Experimental setups

Tied-state Gaussian mixture triphone HMMs were trained on the academic oral presentations from the Corpus of Spontaneous Japanese (CSJ) [4]. The total amount of the presentations was 254 hours. Feature vectors had 39 elements comprising of 12 MFCC and log energy, their delta, and delta delta. The HTK toolkit [3] was used for the EM training. In order to support the operations on sufficient statistics, a modified version of HTK was used for CV-EM and Ag-EM. The language model was a trigram model trained from 6.8M words of academic and extemporaneous presentations from the CSJ. Test set was the CSJ evaluation set that consisted of 10 academic presentations given by male speakers. Speech recognition was performed using the Julius decoder [5]. The number of the CV folds K was 30 for CV-EM. The number of subsets K was 10 and the number of subset selection K' was 6 for Ag-EM.

4.2. Experimental results

Figure 6 shows word error rates when the HMMs were trained using 30 hour random subset of the CSJ data. The number of tied-states of the HMMs was 1000. The lowest word error rates by EM and CV-EM were 27.4% and 27.0%, respectively. Ag-EM gave slightly lower performance than CV-EM when the ensemble size N was 4 but it gave higher performance when the ensemble size N was increased to 24. The lowest error rate by Ag-EM was 26.7% with $N = 24$.

Figure 7 shows word error rates when the HMMs were trained using all the CSJ data. The number of tied-states of the HMMs was 3000. The ensemble size N was 8 for Ag-EM. Similar to Figure 6, both CV-EM and Ag-EM were more robust to larger model sizes than EM, and the lowest word error rate was obtained by Ag-EM.

5. CONCLUSION

We have explained the training procedure of the aggregated EM (Ag-EM) algorithm comparing it to parallel EM and cross-validation EM, and have discussed about the order of the computational cost. We have experimentally investigated the relationship between the ensemble size of the Ag-EM algorithm and its performance. It has been shown that both CV-EM and Ag-EM improves model performance compared to conventional EM and that Ag-EM outperforms CV-EM by using larger ensemble sizes.

6. ACKNOWLEDGMENTS

This study was supported by the Special Coordination Funds for Promoting Science and Technology, Ministry of Education, Culture, Sports, Science and Technology, Japan.

7. REFERENCES

[1] T. Shinozaki and M. Ostendorf, "Cross-validation EM training for robust parameter estimation," in *ICASSP*, Hawaii, 2007, vol. IV, pp. 437–440.

[2] T. Shinozaki and M. Ostendorf, "Cross-validation and aggregated EM training for robust parameter estimation," *Computer speech and language*, accepted.

[3] S. Young *et al.*, *The HTK Book*, Cambridge University Engineering Department, 2005.

[4] T. Kawahara, H. Nanjo, T. Shinozaki, and S. Furui, "Benchmark test for speech recognition using the Corpus of Spontaneous Japanese," in *Proc. SSPR2003*, 2003, pp. 135–138.

[5] A. Lee, T. Kawahara, and S. Doshita, "An efficient two-pass search algorithm using word trellis index," in *Proc. ICSLP*, 1998, pp. 1831–1834.