Soft Margin Estimation of Hidden Markov Model Parameters

Jinyu Li[1], Ming Yuan[2] and Chin-Hui Lee[1]

[1]School of Electrical and Computer Engineering
[2]School of Industrial and Systems Engineering
Georgia Institute of Technology, Atlanta, GA. 30332 USA
{jinyuli, chl}@ece.gatech.edu, myuan@isye.gatech.edu

Abstract

We propose a new discriminative learning framework, called soft margin estimation (SME), for estimating parameters of continuous density hidden Markov models. The proposed method makes direct usage of the successful ideas of soft margin in support vector machines to improve generalization capability, and of decision feedback learning in minimum classification error training to enhance model separation in classifier design. We attempt to incorporate frame selection, utterance selection and discriminative separation in a single unified objective function that can be optimized with the well-known generalized probabilistic descent algorithm. We demonstrate the advantage of SME in theory and practice over other state-of-the-art techniques. Tested on a connected digit recognition task, the proposed SME approach achieves a string accuracy of 99.33%. To our knowledge, this is the best result ever reported on the TIDIGITS database.

1. Introduction

With the prevailing usage of hidden Markov models (HMMs), we have witnessed a rapid progress in automatic speech recognition (ASR) in the last two decades. Usually the HMM parameters are estimated by the traditional maximum likelihood estimation (MLE) method. MLE is known to be optimal for density estimation, but it often does not lead to minimum recognition error which is the goal of ASR. As a remedy, several discriminative training methods have been proposed in recent years to boost the ASR system accuracy. They are maximum mutual information (MMI) [1], minimum classification error (MCE) [2], and minimum word/phone error (MWE/MPE) [3]. MMI training separates different classes by maximizing the posterior probability. On the other hand MCE directly minimizes string errors, while MWE/MPE attempts to optimize the word/phone error rate of a string.

If the training set matches well with the testing set, these discriminative training methods usually achieve very good performance in testing. However, such a good match can not always be expected for most practical pattern recognition problems. The power to deal with possible mismatches between the training and testing conditions can often be measured by the generalization ability of the machine learning algorithms. In particular, large margin classification tools, such as support vector machines (SVMs) [4] have demonstrated superior generalization ability over other conventional classifier learning algorithms. By securing a margin from decision boundaries, correct decision can still be made if the mismatched test samples fall within a tolerance region around the decision boundaries defined by the margin. For example, a combination of SVMs and HMMs was explored in [5] with discrete distributions. Adopting the concept of enhancing margin separation, large margin estimation (LME) [6][7] and its variant, large relative margin estimation (LRME) [8], of HMMs have been recently proposed, and shown to achieve very good results on the TIDIGITS [9] and ISOLET databases. In essence, LME and LRME update the models only with accurately classified samples as if the training set is indeed separable. However, it is well known that misclassified samples are also critical for training classifiers. In SVM learning, for the real inseparable cases, the misclassified samples are used to define a penalty, and a soft margin is found by minimizing a penalized objective function. LME ignores the misclassified samples, and the separation margin it achieved is often hard to be justified as a true margin for good generalization.

In this paper, we integrate the concept of soft margin into HMM parameter estimation. We call this framework soft margin estimation (SME). The proposed SME method defines a unified objective function to integrate frame selection, sample selection and discriminative separation in a flexible framework. The algorithm gets the best result ever reported on the TIDIGITS database. Using 12-state digit models, SME achieves a 99.33% string accuracy using 32-component mixture Gaussian state observation densities. Even with 1-mixture SME models, the achieved string accuracy is better than that obtained with 32-mixture MLE models, although a single Gaussian model can not characterize the state distribution well.

2. Soft Margin Estimation

In the standard binary classification problem, one wants to predict the class labels based on a given vector of features. Let x denote the feature vector. The class label, y, is coded as +1 or -1. The purpose of classification is to construct a classifier f based on a set of training samples $(x_1, y_1),\ldots,(x_n, y_n)$. For a candidate function f, one can check for each sample (x_i, y_i) whether it is correctly classified by f, that is, whether $y_i f(x_i)>0$. SVMs are unique in that they focus more on the generalization of the classification than the number of misclassifications. To formalize it, first consider the so-called separable case where there exists a f such that $y_i f(x_i)>0$ for all samples. In this situation, SVMs solve the following optimization problem

$$\max_{f} \rho(f), \text{subject to } y_i f(x_i) > \rho ,$$

where ρ is often referred to as the margin. With this optimization objective, every mapped sample is at least away from decision boundary with a tolerance distance of ρ . If the mismatch between the training and testing sets only causes a shift less than this margin in the projected space, a correct decision can still be made. This is one way to characterize the generalization property for SVMs.

For the inseparable case, in which there are some misclassified samples, the target function is to get a soft margin ρ to make a tradeoff between maximization of the margin and minimization of the loss of possible misclassified samples, which can be measured as $\sum \varepsilon_i$ in Figure 1, with ε_i defined as a positive slack variable to measure the distance between sample x_i and the class support boundary corresponding to the decision function. If ε_i exceeds the margin ρ , there will be a decision error as shown in Figure 1. The samples with positive ε_i value have a tendency to be misclassified when the mismatch between the training and testing is greater than the margin.

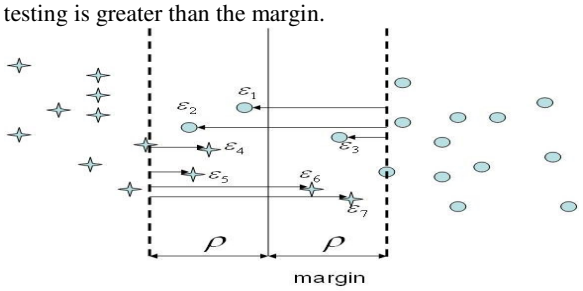

Figure 1: Soft margin classifier

Although SVM has enjoyed a great success in the machine learning community, it can not be easily adopted to ASR modeling. The major reasons are that it does not work directly on temporal sequences and can not handle the hidden states. In this study, we propose SME to combine the advantages of HMMs and SVMs for ASR.

2.1 Model separation measure and frame selection

In order to get good generalization, we try to maximize the separation between different models. For every utterance, we need to define a separation measure, and impose a margin on those separation scores. A common choice is to use log likelihood ratio (LLR) or generalized LLR (GLLR). LLR is defined as:

$$d_i^{LLR}(\Lambda) = \log\left[l\left(X_i \middle| W_{\text{target}}\right) \Big/ l\left(X_i \middle| W_{\text{comp}}\right) \right], \qquad (1)$$

where $\Lambda = (\pi, a, b)$ is the parameter set denoting the state initial probability, transition probability and observation distribution. W_{target} and W_{comp} are the target and the most competitive strings for the *ith* utterance X_i, respectively, while $l(X_i|W_{\text{target}})$ and $l(X_i|W_{\text{comp}})$ are the corresponding likelihood functions. If d_i is greater than 0, the classification is correct, otherwise we get a wrong decision.

In this following, we define a more precise model separation measure rather than using LLR. For every utterance, we select the frames that have different model labels in the target and competitor string. Only those frames can provide discriminative information for models. So we evaluate the frame LLR for those frames and average those frame LLRs as the model separation measure for a given utterance. We use n_i to denote this number of different frames for utterance X_i. Then a separation of the models is defined as:

$$d_i^{SME}(\Lambda) = \frac{1}{n_i} \sum_j \log\left[\frac{l\left(X_{ij} \middle| W_{\text{target}}\right)}{l\left(X_{ij} \middle| W_{\text{comp}}\right)} \right] I\left(X_{ij} \in F_i\right), \qquad (2)$$

where I is an indicator function, F_i is the frame set that the inside frames have different labels in the competing strings and X_{ij} is the jth frame for utterance X_i.

Our separation measure definition is different from LME or MCE, in which the utterance LLR is used. We believe the normalized LLR is more discriminative, because the utterance length and the number of different models in the competing strings affect the overall utterance LLR value. For example, it is not easily justified that an utterance consisting of five different units in the target and competitive strings has more separation than another utterance with only 1 different unit because the former has a larger LLR value.

2.2 SME objective and sample selection

We define the overall objective function for SME as:

$$L^{SME}(\Lambda) = \frac{\lambda}{\rho} + \frac{1}{N} \sum_{i=1}^{N} \ell_i\left(X_i, \Lambda\right), \qquad (3)$$

where $\ell_i\left(X_i, \Lambda\right)$ is a loss function for utterance X_i and ρ is the soft margin. N is the total number of training utterances and λ is a coefficient to balance soft margin maximization and loss minimization of possible misclassified samples. A smaller λ corresponds to allocating a higher penalty to the potential recognition errors. Similar to SVMs, the loss happens when the utterance separation measure is less than soft margin. We use a function $(\ )_+$ to define the loss function as:

$$(\rho - x)_+ = \begin{cases} \rho - x, \text{if } \rho - x > 0 \\ 0, \text{ otherwise} \end{cases}.$$

Then Eq. (3) becomes:

$$\begin{aligned} L^{SME}(\Lambda) &= \frac{\lambda}{\rho} + \frac{1}{N} \sum_{i=1}^{N} \left(\rho - d_i^{SME}(\Lambda)\right)_+ \\ &= \frac{\lambda}{\rho} + \frac{1}{N} \sum_{i=1}^{N} \left(\rho - d_i^{SME}(\Lambda)\right) I\left(X_i \in U\right) \\ &= \frac{\lambda}{\rho} + \frac{1}{N} \sum_{i=1}^{N} \left(\rho - \frac{1}{n_i} \sum_j \log\left[\frac{l\left(X_{ij} \middle| W_{\text{target}}\right)}{l\left(X_{ij} \middle| W_{\text{comp}}\right)} \right] I\left(X_{ij} \in F_i\right) \right) I\left(X_i \in U\right), \end{aligned} \qquad (4)$$

where U is the set of utterances that have the separation measure less than the soft margin. With epoch based optimization we attempt to find the soft margin ρ and a set of parameter Λ to minimize Eq. (4). For a fixed ρ , it is clear that only utterances that have smaller separation scores than the soft margin contribute to the optimization process, i.e. SME focuses on difficult samples, which often have a tendency to be misclassified.

As shown in the formula in Eq. (4), we integrate frame selection (by $I\left(X_{ij} \in F_i\right)$), utterance selection (by $I\left(X_i \in U\right)$) and discriminative separation in a single unified objective function. This quantity provides a flexible framework for future studies. For example, for frame selection, we can

define F_i as a subset with frames more critical for discriminating models instead of equally choosing distinct frames in the current study. As in [10], we can also approximate the indicator function with a smooth embedding for direct optimization of Eq. (4).

2.3 Solution to SME

Unlike SVMs, in which the solution to the soft margin optimization problem can be well formulated as a constrained quadratic programming problem, the direct optimization of L^{SME} in Eq. (4) is hard to solve.

In this study, we search for a sub-optimal solution. First, we choose a margin ρ heuristically. Because of a fixed ρ, we only need to consider the samples with separation smaller than the margin. Assume that there are a total of N_C utterances satisfying this condition, we can minimize the following function with the constraint $d_i^{SME}(\Lambda) < \rho$:

$$L^{sub}(\Lambda) = \sum_{i=1}^{N_c} \left(\rho - d_i^{SME}(\Lambda)\right) .$$

Now, this problem can be solved by a generalized probabilistic descent (GPD) algorithm as in [10], with α_t as a step size: $\Lambda_{t+1} = \Lambda_t - \alpha_t \nabla L^{sub}(\Lambda)|_{\Lambda=\Lambda_t}$.

3. Comparison with LME and MCE

In the following, we compare SME with LME and MCE to illustrate SME's advantages in better usage of training frames and samples, and discriminative objective for generalization.

3.1 Difference between SME and LME

In the class of discriminative training methods, LME and its variant LRME also try to handle the model generalization issue. These two methods are very similar, and LME was reported with better performance. Here, we first distinct SME from LME. LME uses utterance LLR as the model separation measure and maximizes the distance between the decision boundary and the nearest correctly classified samples, as:

$$\max_{\Lambda} \rho(\Lambda), \text{subject to } d_i^{LLR}(\Lambda) > \rho \ .$$

However, this objective function neglects the misclassified samples, e.g., samples 1, 2, 3 and 4 in Figure 2. In this case, the margin obtained by LME is hard to be justified as a real margin for generalization. Consequently, LME often needs a very good preliminary estimate from the training set to make the influence of ignoring misclassified samples small. So it usually uses MCE models as the initial model.

In contrast, SME works on all the training data, both the correctly classified and misclassified samples, as in Eq. (4). We believe those misclassified samples are important for classifier learning because they carry the information to discriminate models.

In SME, model separation measure is carefully treated by a normalized LLR measure over only the set of different frames. With such normalization, the utterance separation scores can be more closely compared with a fixed margin ρ than an un-normalized LLR, without being affected by different numbers of distinct units and length of the utterances. The soft margin can now be defined with a value comparable with the average frame LLR.

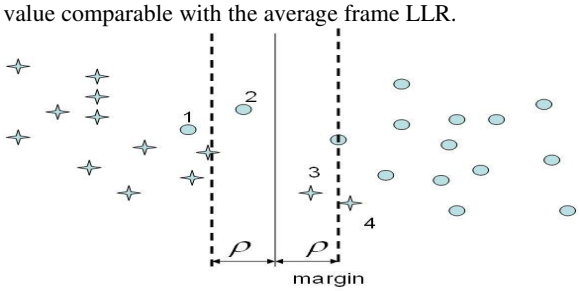

Figure 2: LME classifiers

3.2 Difference between SME and MCE

The misclassification measure in MCE is also a separation measure, defined as the negative of LLR or GLLR. Usually MCE transforms it with a sigmoid function to approximate the string error count. Because sigmoid function is monotonic, minimizing the approximate string error can be considered equivalent to minimizing the original misclassification measure. As a result, MCE is also good at enlarging model separation.

Despite of this similarity, SME has some advantages over MCE and results in a better performance than MCE which will be demonstrated in the experiments in Section 4 later. MCE uses LLR (or GLLR) as a separation measure while SME normalizes LLR with the number of frames with different target and competing models. The normalization factor makes the comparison of utterance separations more discriminate.

In practice, there is another difference between SME and MCE. Because of the sigmoid function's first order derivative is near 0 for the samples that are far away from the decision boundary, MCE doesn't update parameters with those far away misclassified samples. Ignoring these difficult samples may restrict the learning methods from finding optimal model parameters. Consequently, in theory, SME is more flexible than MCE, because the latter does not effectively take into account the information embedded in difficult samples.

4. Experiment

We evaluated our proposed framework on the TIDIGITS database. To our knowledge, the best result on this database was reported in [7] by using LME. We used the same configuration as that in [7]. There are 11 whole-digit HMMs, one for each of the 11 English digits, including the word "oh". Each HMM has 12 states and each state observation density is represented by a mixture Gaussian density. The input features are 12MFCCs + energy, and their first and second order time derivatives.

We first use HTK to build the baseline HMMs with MLE. We also trained MCE models for comparison. It gave slightly better results than the MCE results reported in [7]. Our SME models were initiated with the MLE models. This is a clear contrast with the LME models, built on top of the well-performed MCE models [7].

Table 1 compares different training methods with various number of mixture components. Only string accuracies are listed in Table 1. We believe at this high

level of performance in TIDIGITS, the string accuracy is a strong indicator of model effectiveness. Clearly SME significantly outperforms MLE and MCE, and is consistently better than LME. For 1-mixture SME models, the string accuracy is 98.64%, which is better than that of the 32-mixture MLE models. The goal of our design is to separate the models as far as possible, instead of modeling the observation distributions. With SME, even 1-mixture models can achieve satisfactory model separation.

We believe a string accuracy of 99.33% listed in the bottom row of Table 1 represents the best result ever reported on the TIDIGITS task. The excellent SME performance is attributed to the well defined model separation measure, good objective function for generalization and better handling of difficult training samples than conventional MCE.

Figure 3 plots the histograms of the separation measure of the testing utterances for the 32-mixture MLE, MCE and SME models, respectively. Usually the larger the separation measure, the better the models are. The separation used here is defined in Eq. (2) with the normalized LLR. As the right most curve in Figure 3, SME gets significant better separation than MLE and MCE, because of direct model separation maximization.

Table 1: String accuracy comparison with different methods

	MLE	MCE	LME [7]	SME
1-mix	95.20	96.94	96.94	98.64
2-mix	96.90	97.40	98.51	98.90
4-mix	97.80	98.24	98.80	99.10
8-mix	98.03	98.66	99.14	99.23
16-mix	98.36	98.87	99.20	99.24
32-mix	98.51	98.98	99.28	99.33

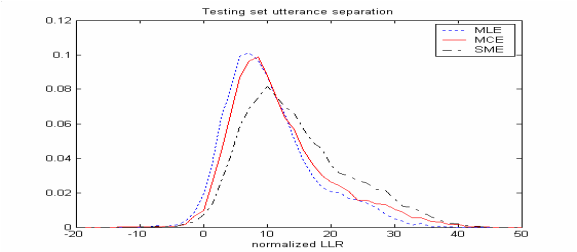

Figure 3: Testing utterance separation

5. Conclusion

We have proposed a novel discriminative training method, called SME, to achieve both high accuracy and good model generalization. By combining the advantages in SVM and MCE it directly maximizes the separation of competing models to enhance the testing samples to approach a correct decision if the deviation from training models is within a safe margin. Frame and utterance selections are integrated into a unified framework to select the training utterances and frames critical for discriminating competing models. We have compared SME with LME, a recently proposed discriminative training method, both in theory and in experiment to show the effectives of our proposed method. Tested on the TIDIGTS database, even 1-mixture model can well separate different words and produce better string accuracy than that with 32-mixture MLE models. SME's performance is consistently better than that of LME, and significantly better than those of MLE and MCE.

This paper only presents our initial study, we are now working on many related research issues to further complete the theory of SME. The first is to design a good optimization method. Current solution is to choose the soft margin in advance, which is too heuristic and suboptimal. If we can optimize the soft margin and HMM parameters simultaneously, better performance is expected. Second, we only select the most competitive string to define the separation measure. *N*-best list and lattice can also be incorporated to enrich the competing alternatives. Finally, we will extend our current work to large vocabulary ASR. We have already done some experiments on the TIMIT database for phone recognition and our preliminary results have shown SME's advantage over MCE. Further results will be reported later.

6. Acknowledgements

We would like to thank Dr. Hui Jiang of York University for valuable discussions about LME. This work was partially supported by the NSF grant, IIS-04-27113, and the GALE program of the Defense Advanced Research Projects Agency, Contract No. HR0011-06-C-0022.

7. References

[1] Normandin, Y., "Maximum Mutual Information Estimation of Hidden Markov Models," In: Lee, C.-H., Soong, F.K. and Paliwal, K.K., (Eds.), Automatic Speech and Speaker Recognition. Kluwer Academic Publishers, 1996.

[2] Juang, B.-H., Chou, W., and Lee, C.-H., "Minimum Classification Error Rate Methods for Speech Recognition," *IEEE Trans. on Speech and Audio Proc.*, vol. 5, no. 3, pp. 257-265, 1997.

[3] Povey, D., and Woodland, P.C., "Minimum Phone Error and I-smoothing for Improved Discriminative Training," *Proc. ICASSP*, vol. 1, pp. 105-108, 2002.

[4] Burges, C., "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery,* vol. 2, no. 2, 1998.

[5] Altun, Y., Tsochantaridis, I., and Hofmann, T., "Hidden Markov Support Vector Machines," *Proc. ICML*, 2003.

[6] Li, X., Jiang, H., and Liu, C., "Large Margin HMMs for Speech Recognition," *Proc. ICASSP*, 2005.

[7] Li, X., "Large Margin Hidden Markov Models for Speech Recognition," *M.S. thesis,* Department of Computer Science and Engineering, York University, Canada, 2005.

[8] Liu, C., Jiang, H., and Li, X., "Discriminative Training of CDHMMS for Maximum Relative Separation Margin," *Proc. ICASSP*, 2005.

[9] Leonard, R.G., "A Database for Speaker-Independent Digit Recognition," *Proc. ICASSP*, 1984.

[10] Katagiri, S., Juang, B.-H., and Lee, C.-H., "Pattern Recognition Using a Family of Design Algorithms Based upon the Generalized Probabilistic Descent Method," *Proc. IEEE*, vol. 86, no. 11, pp. 2345-2373, 1998.