Model-Based Margin Estimation for Hidden Markov Model Learning and Generalization

Sabato Marco Siniscalchi[a,*], Jinyu Li[b], Chin-Hui Lee[c]

[a] *Faculty of Engineering and Architecture, Kore University of Enna, Cittadella Universitaria, Enna, Sicily, Italy*
[b] *Microsoft Corporation, Redmond, WA, 98052 USA*
[c] *School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA*

Abstract

Recently, speech scientists have been motivated by the great success of building margin-based classifiers, and have thus proposed novel methods to estimate continuous-density hidden Markov model (HMM) for automatic speech recognition (ASR) according to the notion that the decision boundaries determined by the estimated HMMs attain the maximum classification margin as in learning support vector machines (SVMs). Although a good performance has been observed, the margin used in the ASR community is often specified as a parameter that has no explicit relationship with the HMM parameters. The issues of how the margin is related to the HMM parameters and how it directly characterizes the generalization capability of HMM-based classifiers have not been addressed so far in the community. In this paper we attempt to formulate the margin used in the soft margin estimation (SME) framework as a function of the HMM parameters. The key idea is to relate the standard distance-based margin with the concept of divergence among competing HMM state Gaussian mixture model densities. Experimental results show that the proposed model-based margin function is a good indication about the quality of HMMs on a given ASR task without the conventional needs of running experiments extensively using a separate set of test samples.

*Corresponding author
Email address: E-mail: marco.siniscalchi@unikore.it, Telephone:+39 3472913375, Fax:+39 0935536623 (Sabato Marco Siniscalchi)

Keywords: Soft margin estimation, hidden Markov model, generalization, divergence, discriminative classifier learning

1. Introduction

For classifier learning based on a set of training samples, one key design issue is the ability for the classifiers to generalize to unseen test data, some of them can come from mismatch conditions different from those observed in training. In particular, large margin learning frameworks, such as support vector machines (SVMs) [2], have demonstrated superior generalization capabilities over other conventional classifiers. By securing a margin from the decision boundary to the nearest training sample, a correct decision can still be made if the mismatched test sample falls within a tolerance region around the original training samples defined by the margin. Inspired by the past success of margin-based classifiers, there is a trend to incorporate the margin concept into hidden Markov models (HMMs) for automatic speech recognition (ASR). Recent attempts based on margin maximization [7, 19, 14, 17, 13, 15] have shown some advantages over conventional discriminative training methods (e.g., [1, 9, 8]) for some ASR tasks.

The margin for ASR applications is defined in terms of the distance of log likelihood values between the true model and its competing models. In large margin estimation of HMMs [7] the goal is to adjust decision boundaries, which are implicitly determined by all HMM models, through optimizing HMM parameters to make all support tokens (namely, a subset of the spoken sentences) as far from the decision boundaries as possible. These support token were selected such that the distance between true model and competing model fall between zero and to a preset positive number ρ. The large margin HMM technique proposed in [17] constraints the Mahalanobis score of each target sequence to exceed that of each competing sequence by an amount equal to or greater

than the Hamming distance between these two sequences. The work in [19] incrementally adjusts the margin value to achieve best performance. Soft margin estimation (SME) [14, 13, 15] presets the margin (i.e., the distance between competing HMM models) and only optimizes the HMM parameters. Typically the value of the margin varies according to the number of the parameters and the complexity of the model. Therefore, constraining the margin using expert knowledge, as done so far in margin- based HMM training for ASR, often limits the number of models to be experimented and the ability to achieve a good performance. Furthermore, the issues of how the margin is related to the HMM parameters and how it directly characterizes the generalization ability of HMM based classifiers have not been addressed so far in the above methods.

The divergence is known as a good measure to compare two probability densities [4], and a new system divergence measure for HMM as an average divergence between top competing models in the complex set of HMMs is here proposed. The margin is expressed as a function of the system divergence, which can then be plugged into the SME objective function [15] for optimization. When compared with ASR results obtained in conventional SME [15] using empirically specified margin constants, the proposed margin function gives very similar performance, yet with potentially more theoretical impacts and flexibility than the original SME algorithms. Furthermore, it will be shown how the proposed model-based margin can be employed as a figure of the quality of the system design. This figure aims at predicting the performance of HMM-based recognition systems without the need of actually running recognition experiments. The idea of predicting the run-time performance of a HMM based recognition system without running any recognition experiments using a separate set of test samples is highly desirable both in theory and in practice, and it was formerly explored in [6]. Specifically, the authors showed that the error

rate can be accurately estimated from the probabilistic distance between the assumed parametric models. The model-based margin introduced in this paper can be used to compare HMM-based ASR systems and predict which one would perform better on the unseen test samples.

The rest of the paper is organized as follows. In Section 2, the starting point for this work, i.e., the general concept of margin for two classes, is briefly described. Section 3.1 gives details about how to generalize the notion of a separation margin for Gaussian mixture models using a symmetric divergence. The extension for multiway classification and HMMs is also given in Section 3.1. The soft margin estimation procedure via model divergence is presented in Section 4. The experimental evaluation is carried out in Section 5. Finally, Section 6 concludes this work.

2. SVMs for Two Classes

The purpose of SVM design is to construct a projection w and an offset b based on a set of training samples $(x_1,\ y_1),\ldots,\ (x_n,\ y_n)$, where y_i is the class label. In the separable case, where there exists a pair $(w,\ b)$ such that $y_i(\rho x_i + b) > 0$ for all samples, SVMs solve the following optimization problem: $max_w \frac{1}{\|w\|}\ (s.t.\ y_i(x_i w_i + b) - 1 > 0)$ where $\|\rho\|$ is the Euclidean norm of w, and $1/\|w\|$ is referred as the margin. With this optimization objective, every mapped sample is at least away from decision boundary with a tolerance distance of $1/\|w\|$. The margin, $1/\|w\|$, can be considered as a measure to characterize the generalization property for the SVMs, and the margin (times 2) can be considered as a separation distance of these two competing models. This view can be extended to defining the optimization target of non-separable SVMs as a combination of **empirical risk minimization** and **model separation (or margin) maximization**.

3. System Generalization via Model Distance

The strong relationship between the model distance and the margin outlined in Section 2 can also be extended to the Gaussian mixture model (GMM) and HMM cases, which have a much large number of parameters and with nonlinear decision boundaries. In the following, a system divergence to measure the model distance for GMMs and HMMs is first defined, and the link between the model distance and the margin is introduced.

3.1. GMMs

If the two classes are each modeled by GMMs, a symmetric divergence can be used to measure the model distance [10]:

$$D = E\left\{-ln\frac{p_1(x)}{p_2(x)} \mid w_2\right\} - E\left\{-ln\frac{p_1(x)}{p_2(x)} \mid w_1\right\}, \tag{1}$$

where $p_1(x)$ and $p_2(x)$ are the probability density functions of the two competing models, w_1 and w_2.

For two Gaussian densities, k, and l, a closed form exist for Eq. (1) [4]:

$$D_G(k,l) = \frac{1}{2}tr\left\{\left(\Sigma_k^{-1} + \Sigma_l^{-1}\right)\left(\mu_k - \mu_l\right)\left(\mu_k - \mu_l\right)^T\right\} + \\ + \frac{1}{2}tr\left(\Sigma_k^{-1}\Sigma_l + \Sigma_l^{-1}\Sigma_k - 2I\right), \tag{2}$$

where (μ_k, Σ_k) and (μ_l, Σ_l) are the means and covariance matrices of the Gaussian densities, k and l, respectively. The matrix I is the identity matrix. For GMMs, the following approximation is made for the divergence of the ith and jth GMMs:

$$D_{GMM}(i,j) = \sum_{k=1}^{m_i}\sum_{l=1}^{m_j} c_{ik}c_{jl}D_G(ik,jl), \tag{3}$$

where ik and jl indicate the kth and lth Gaussians of the ith and j GMMs. This approximation weights all the pair wise Gaussian components from GMMs with the corresponding mixture weights (c_{ik} and c_{jl}) and sums them together.

3.2. GMMs and HMMs for multiway classification

The system divergence as the model distance for multiple GMMs is now defined as:

$$D(\Lambda) = \frac{1}{NG} \sum_{i} D_{GMM}(i, nearest(i)), \tag{4}$$

where Λ denotes all GMM parameters in the system and NG is the total number of GMMs in the system. For the ith GMM, only its nearest GMM (*nearest(i)*) is considered to significantly contribute the value of the system divergence. In our opinion the divergence of two GMMs far apart from each other provides little information to quantify the system confusion, or system generalization. For HMM system, every state is modeled by a GMM. Hence, the system divergence of HMMs can also be defined in a similar way to Eq. (3) by considering all the state GMMs. The only difference is that the GMMs belong to the same speech unit cannot be included in defining *nearest(i)*. It is well known that the transition probability of the HMMs is not critical for speech recognition. Therefore, most discriminative training methods only adjust the Gaussian parameters in practice. As a consequence, we do not include the transition probability into the divergence distance.

3.3. Mapping between Model Distance and Margin

The margin is used to keep the class samples away from the decision boundary. A large model distance implies there is enough space between models. Similarly a larger margin often results in better system generalization. Because

of this strong relationship, the margin is defined as a monotonic function of the model distance:

$$\rho(\Lambda) = f(D(\Lambda)), \tag{5}$$

where D is defined as in Eq. (4).

For the separable SVM case, $\Lambda = (w, b)$ and $1/\|w\|$ is referred as the margin. The model distance is $2/\|w\|$, as shown in Section 2. Hence, $\rho(\Lambda) = D(\Lambda)/2$. For the systems with multiple GMMs or HMMs, the mapping relationship between system model distance D and margin ρ is complex because it is hard to get the decision boundary for the multiple classes as a function of model parameters. Different systems may have different mapping. In the next sections, it will be shown how the mapping function is built in the context of SME for ASR tasks.

4. Soft Margin Estimation via Divergence

In this section, a link between the divergence-based model distance and the margin within the SME framework is made. First the original SME formulation is introduced. Then, its potential weaknesses are discussed. As an improvement, the margin is expressed as a function of the system divergence so that it can be plugged into the objective function of SME, so that the margin and the HMM (GMM) parameters can be optimized simultaneously.

4.1. Original SME Formulation

Here, SME is briefly introduced. A detailed discussion can be found in [13, 15]. SME has two targets for optimization. The first is to minimize an empirical risk (i.e., the risk on the training set). The other is to maximize the margin, which is related to classifier generalization. These two quantities are combined into one objective function for minimization:

$$L^{SME}(\Lambda) = \frac{\lambda}{\rho} + R_{emp}(\Lambda) = \frac{\lambda}{\rho} + \frac{1}{N}\sum_{i=1}^{N} l(O_i, \Lambda), \tag{6}$$

where Λ denotes the HMM parameters, $l(O_i, \Lambda)$ is a loss function for the spoken utterance O_i, N is the number of training utterances, ρ is the soft margin that is set in advance using expert knowledge, and λ is a coefficient to balance soft margin maximization and empirical risk ($R_{emp}(\Lambda)$) minimization.

Before defining the loss function, it should be remarked that a continuous ASR system can generate either a single sentence (the best decoded sentences) or a list of competing sentences (usually organized in the form of either a n-best lists or a confusion network) at its output. The loss function can now be defined as:

$$\begin{aligned} l(O_i, \Lambda) &= (\rho - d(O_i, \Lambda))_+ \\ &= \begin{cases} \rho - d(O_i, \Lambda) & \text{if } (\rho - d(O_i, \Lambda)) > 0 \\ 0 & \text{otherwise} \end{cases}, \end{aligned} \tag{7}$$

with the separation measure d defined as:

$$d(O_i, \Lambda) = \frac{1}{n_i}\sum_{j} log\left[\frac{p_\Lambda(O_{ij}|S_i)}{p_\Lambda(O_{ij}|\hat{S}_i)}\right] I(O_{ij} \in F_i), \tag{8}$$

where F_i is the frame set in which the speech samples (commonly referred to as frames) have different labels in the competing decoded sentences. $I(.)$ is an indicator function, O_{ij} is the jth frame of the spoken utterance O_i. The number of frames that have different labels in the target and competing strings for O_i is indicated with n_i, and $p_\Lambda(O_{ij}|S_i)$ and $p_\Lambda(O_{ij}|\hat{S}_i)$ are the likelihood scores for the target string S_i and the most competitive string $\hat{S}_i$, respectively.

4.2. Potential Weaknesses of the Original SME Formulation

There are two potential shortfalls of the original SME formulation [13, 15]. The first is that the solution to minimizing the objective in Eq. (6) is sub-optimal because it presets the margin value ρ, and optimizes the HMM parameters only. The solution assumes that the optimal margin is known in advance. Since there is no direct way to know the optimal value of ρ, it is empirically determined if it delivers the best ASR result. The second problem is common to all margin-based HMM training algorithms so far proposed in the ASR community. The margin ρ in Eq. (6) is purely a numerical value, without any direct relationship with HMM parameters. In contrast, as discussed in Section 2, the margin of SVMs can be considered as a model separation related to the parameter, w. Therefore, it is highly desirable, both in theory and in practice, that the margin can also be a function of parameters to characterize the generalization issue of HMMs in the ASR field.

4.3. SME Formulation with Divergence

To overcome the above deficiencies of the original SME, it is desirable to express the margin as a function of the HMM parameters. However, it is difficult to determine the exact mapping function, given the HMM system is too complex. Instead, we observed that the square root of the divergence in Eq. (4) is similar to the margin used in our original SME work, as shown in Table 1 later. To this end, the square root of the divergence in Eq. (4) is used to define a model-based (or divergence-based) margin for HMMs. Hence, the model-based margin as a function of the HMM parameters is defined as:

$$\rho(\Lambda) = \frac{1}{NG} \sum_{i} D_{GMM}(i, nearest(i))^{\frac{1}{2}}, \tag{9}$$

By embedding this model-based margin into the SME framework, the new SME objective function becomes:

$$
\begin{aligned}
L^{SME}(\Lambda) &= \frac{\lambda}{\rho(\Lambda)} + \frac{1}{N}\sum_{i=1}^{N}(\rho(\Lambda) - d(O_i,\Lambda))_+ \\
&= \frac{\lambda}{\rho(\Lambda)} + \frac{1}{N}\sum_{i=1}^{N}(\rho(\Lambda) - d(O_i,\Lambda))I((\rho(\Lambda) - d(O_i,\Lambda)) > 0) \\
&= \frac{\lambda}{\rho(\Lambda)} + \\
&+ \frac{1}{N}\sum_{i=1}^{N}(\rho(\Lambda) - d(O_i,\Lambda))\frac{1}{1 + exp(-r((\rho(\Lambda) - d(O_i,\Lambda)))},
\end{aligned}
\tag{10}
$$

where r is the tilting parameter for sigmoid function. The right hand side of Eq. (10) is obtained by smoothing the indicator with a sigmoid function so it can be minimized with a generalized probabilistic descent (GPD) [9] algorithm. The model parameter is updated sequentially such that

$$
\Lambda_{n+1} = \Lambda_n - \epsilon_n \frac{\partial L^{SME}(\Lambda)}{\partial \Lambda} \tag{11}
$$

The key of GPD is to get the derivatives of loss function with respect to model parameters Λ, which denotes the set of mean and covariance parameters in GMM. The detailed formulation of the derivatives is in the following.

$$
\frac{\partial L^{SME}(\Lambda)}{\partial \Lambda} = \left\{ \begin{array}{c} \left[-\frac{\lambda}{\rho^2(\Lambda)} + \frac{1}{N}\sum_{i=1}^{N}\{l(1+\gamma(\rho(\Lambda) - d(O_i,\Lambda))(1-l))\}\right]\frac{\partial(\rho(\Lambda))}{\partial\Lambda} \\ +\frac{1}{N}\sum_{i=1}^{N}\{l(1+\gamma(\rho(\Lambda) - d(O_i,\Lambda))(1-l))\}\frac{\partial(-d(O_i,\Lambda))}{\partial\Lambda} \end{array} \right\}. \tag{12}
$$

Since $d(O_i,\Lambda)$ is a normalized log likelihood ratio, $\frac{\partial(-d(O_i,\Lambda))}{\partial\Lambda}$ can be computed similarly to what has been done in minimum classification error (MCE) training. Please refer [8] for the detailed formulations of those derivatives. $\frac{\partial(\rho(\Lambda))}{\partial\Lambda}$ can be obtained in a similar way since $\rho(\Lambda)$ can be written as a function of mean and covariance parameters of GMM. After getting all the derivatives,

GPD is used to update all GMM parameters.

It should be noted that although the empirical approximation of margin is not precise, it can still work well under the framework of SME. As opposed to the separable case, there is no unique soft margin value for the cases of inseparable classification. The final soft margin value is affected by the choice of the balance coefficient, λ. In essence, Eq. (10) works well for generalization in two parts. The first is to pull the samples away from the decision boundary with a distance of margin by reducing the empirical risk. The second is to make this margin as a function of the system model distance, and then maximize it by minimizing the objective function in Eq. (10). Recently, divergence was used in a new DT method, called minimum divergence training (MDT) [3]. A major difference between model-based SME and MDT is that SME is a margin-based DT method while MDT is still a conventional one. SME jointly maximizes the system divergence and minimize the empirical risk, while MDT minimizes the divergence between the recognized model sequence and the correct model sequence in training utterances. Furthermore, SME uses divergence globally, as a measure of system generalization, while MDT evaluates divergence utterance by utterance, as a local measure of the dissimilarity between the competing models and correct models in each utterance.

5. Speech Recognition Experiments

SME has been applied to Aurora-2 for robustness issue [12, 18], and TIDIGITS and WSJ05 for various complexity issues [13, 14, 15]. In this paper, we just want to study the need for a model-based formulation. Therefore, we validate the utility of our modified SME framework with model-based margin using the TIDIGITS and Aurora-2 tasks, a connected-digit recognition task. The discovery can be easily generalized to other tasks with different robustness and

complexity issues given the consistent results we observed [12, 13, 14, 15, 18]. Through this recognition experiments, it will be shown that the square root of the divergence in Eq. (4) is very close in value to the margin preset using expert knowledge, so it is a viable solution. Furthermore, it will be shown that the model-based margin could be used as a figure to compare HMM-based ASR systems one to another and predict which one would perform better on a given task without running any experiment on a separate set of test samples.

The TIDIGITS [11] and Aurora-2 [16] corpora are two corpora of connected digit recognition.

5.1. TIDIGITS

5.1.1. Experimental Setup

There are 8623 digit strings in the training set and 8700 digit strings for testing. The hidden Markov model toolkit (HTK) was used to build the baseline maximum likelihood estimation (MLE) models. There were 11 whole-digit HMMs, one for each of the 10 English digits plus the word "o". GMMs were used as state-conditional probability distributions. From each speech signal, a sequence of feature vectors were extracted using a 25 ms Hamming window and a window shift of 10 ms. Each feature vector consisted of 12 Mel-frequency cepstral coefficients (MFCC) and the frame energy, augmented with their delta and acceleration coefficients. This resulted in 39-dimensional vectors. Models of MCE [8], a traditional DT, were also trained for comparison. SME models were initiated with the MLE models. Digit decoding was based on unknown length without imposing any language model or insertion penalty.

5.1.2. Connected Digit Recognition Results

Table 1 compares string accuracies of different training methods by varying the number of mixture components in each GMM in each HMM state. The

Table 1: *EVALUATION SET STRING ACCURACY COMPARISON WITH DIFFERENT METHODS ON THE TIDIGITS TASK.*

	MLE	MCE	SME_D	SME_C
1-mixture	95.20%	96.94%	98.76%	98.64%
2-mixture	96.90%	97.40%	98.91%	98.90%
4-mixture	97.80%	98.24%	99.15%	99.10%
8-mixture	98.03%	98.66%	99.29%	99.23%
16-mixture	98.36%	98.87%	99.29%	99.24%

column labeled with SME_D and SME_C show the accuracy of the SME method with model-based margin, and the of the original SME method [14], respectively. Both SME methods outperform MLE and MCE significantly. Furthermore, SME_D accuracy is close to the original SME_C technique, which implicitly confirms the viability of the proposed solution, that is, the use of the square root of the divergence to link the HMM margin and the HMM parameters within the SME framework.

For 1-mixture SME_D models, the SME_D string accuracy is 98.76%, which is better than that of the 16- 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_D, even 1-mixture models can achieve satisfactory model separation. Finally, string accuracies as high as 99.29%, or 99.24%, which are listed in the bottom row of Table 1, represent excellent results on the TIDIGITS task. This good SME performance can be attributed to the well defined model separation measure, good objective function for generalization and better handling of difficult training samples than conventional discriminative training techniques, such as MCE. The values of λ and r for the SME_D and the SME_C systems were set to 10 and 2.0, respectively.

5.1.3. System Divergence Evaluation

In Table 2 we list the square root values of the system divergence of all the models. The divergence trend of the three training methods is clear within the

Table 2: *SQUARE ROOT OF SYSTEM DIVERGENCE ($\rho(\Lambda)$) WITH DIFFERENT METHODS ON THE TIDIGITS TASK.*

	MLE	MCE	SME_D	SME_C
1-mixture	3.68	5.12	5.55	5.00
2-mixture	5.52	6.30	6.47	6.00
4-mixture	6.83	6.99	7.15	7.50
8-mixture	7.96	8.16	8.57	8.50
16-mixture	8.99	9.09	9.97	9.00

same model configuration. The divergence of MLE is the smallest and that of SME_D is the largest in all model configurations. Furthermore, the value of the model-based margin of group of classifiers with competing designs (systems referring to the same row in Table 2), one can predict which system might perform the best without running any experiment on the evaluation data.

On another hand, the model-based margin is not the only indicator for accuracy. For 1-mixture SME models, the string accuracy is 98.76%, which is better than that of the 16- mixture MLE models. Nevertheless, the system divergence of the 1-mixture SME_D models is far less than that of the 16-mixture MLE models. Hence, generalization is not only the factor that determines the recognition performance; nonetheless, it is a very good index when the model setup is the same (e.g., the same number of Gaussians)

Finally, the rightmost column of Table 2 with label SME_C lists the empirical margins used in the original SME method [14] that have been selected by hand using expert knowledge. It is easy to see that the model-based margins (SME_D) are similar in value to these empirical margins (SME_C). This result confirms that the proposed approach to define a margin dependent upon the HMM parameters represents indeed a good solution.

5.2. Aurora-2

In this work, we choose λ equal to 10, but the interested reader is referred to [12] for an investigation on the system performance on the Aurora-2 task as

the λ varies. The value of r was set equal to 2.0.

5.2.1. Experimental Setup

The Aurora-2 task defines two training modes: (a) clean training mode in which the acoustic model is trained on clean data alone and (b) multi-conditional training where training is done using both clean and noisy data. Three testing sets are provided for the evaluation of the Aurora-2 task. Each set has 4 subsets of 1001 utterances. The first testing set, *set A* contains four sets of 1001 sentences, corrupted by subway, babble, car, and exhibition hall noises, respectively, at different SNR levels. The noise types included in this set are the same as those in the multi-conditional training. The second set, *set B* contains 4 sets of 1001 sentences each, corrupted by restaurant, street, airport, and train station noises at different SNR levels. These noise types are different from the ones used in the multi-conditional training. The test *set C* contains 2 sets of 1001 sentences, corrupted by subway, and street and airport noises. The data set C was filtered with the MIRS filter before the addition of noise in order to evaluate the robustness of the algorithm under convolutional distortion mismatch.

The acoustic features are 13-dimension MFCCs, appended by their first, and second-order time derivatives. The baseline experiment configuration follows the standard script provided by ETSI [16].

Table 3: *AURORA-2 WORD RECOGNITION ACCURACY COMPARISON WITH DIFFERENT TRAINING CONDITIONS AND DIFFERENT EVALUATION SETS. THE LAST ROW SHOWS THE QUARE ROOT OF SYSTEM DIVERGENCE ($\rho(\Lambda)$) .*

Training Conditions	System	Averaged Word Accuracy [16]	$\rho(\Lambda)$
Clean	MLE	61.15%	4.78
	SME_C ($\lambda = 10$)	66.86%	8.88
	SME_D ($\lambda = 10$)	66.88%	8.96
Multi-conditioning	MLE	86.41%	7.45
	SME_C ($\lambda = 10$)	87.72%	11.96
	SME_D ($\lambda = 10$)	87.89%	12.32

5.2.2. Recognition Results & System Divergence Evaluation

Table 3 compares word accuracies of different training methods averaging over different noise types and levels as suggested in [16]. The proposed SME_D method outperforms MLE for both clean and multi-conditional training, and compares well with the SME_C method in all training conditions. The last column of Table 3 demonstrates that the system divergence of MLE is the smallest and that of SME_D is the largest in all model configurations. Those results confirm the validity of the proposed model-based formulation approach for robust ASR applications as well.

6. Conclusion

In this work, we extend the margin from a constant value in our previous work to a function of the system model distance. This distance-based margin is a function of all the model parameters and well characterizes the system generalization. A system divergence is defined as the model distance of GMMs or HMMs. A model-based margin is plugged into the objective function of SME. The modified SME achieved a similar performance to that obtained with the previously proposed method, and with a better theoretic foundation. This facilitates us to design SME quickly without the need to try different constants for systems with different complexities. From the experiment, for each mixture model setting, the divergence follows the same trend as the accuracy. Meanwhile, a greater divergence result in greater accuracy in the digit recognition task. This demonstrates better generalization results in better accuracy. Furthermore, the value of the model-based margin, it is not the only factor to determine the system performance. The work in this letter goes beyond the margin definition in SVM. In binary SVMs, the margin is easily viewed as the separation distance of the two competing classifiers. For multiple classifiers,

there is no obvious extension. We directly relate the margin with model parameters to characterize the generalization issue. Although this work uses HMMs as classifiers, we believe this model-based margin can be generalized to benefit other multi-class margin-based methods. As stated in this letter, a square root operation is adopted to map the system divergence to the soft margin. Although satisfactory results were obtained, many efforts are needed to investigate what is the underlined theory and whether there are other functions to associate the system divergence with margin in SME. In [5], some precise realizations of divergence with heavy computation cost are discussed. In our study, the computation of divergence is simple and easy for optimization, but with some potential precision loss. We will investigate whether better precision modeling of divergence will bring out further improvements. In [13] we have shown that SME also works well on a large vocabulary continuous speech recognition (LVCSR) task. We will demonstrate the effectiveness of this model-based SME on a LVCSR task in future works.

7. References

[1] Bahl, L.R., Brown, P.F., deSouza, P.V., Mercer, R.L., 1986. Maximum mutual information estimation of HMM parameters for speech recognition, in: Proc. ICASSP, Tokyo, Japan.

[2] Burges, C., 1998. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 2 (2), 121–167.

[3] Du, J., P. Liu, F.K.S., Zhou, J.L., Wang, R.H., 2006. Minimum divergence based discriminative training, in: Proc. ICASSP, Pittsburgh, PA, USA. pp. 2410–2413.

[4] Fukunaga, K., 1990. Introduction to Statistical Pattern Recognition. Academic Pr Inc (2nd ed.), New York.

[5] Hershey, J., Olsen, P., 2007. Approximating the Kullback Leibler divergence between Gaussian mixture models, in: Proc. ICASSP, Honolulu, Hawaii, USA. pp. IV 317–IV 320.

[6] Huang, C.S., Wang, H.C., Lee, C.H., 2003. A study on model-based error rate estimation for automatic speech recognition. IEEE Trans. Speech and Audio Proc. 11 (6), 581–589.

[7] Jiang, H., Li, X., Liu, C., 2006. Large margin hidden markov models for speech recognition. IEEE Trans. Audio, Speech and Language Proc. 14 (5), 1584–1595.

[8] Juang, B.H., Chou, W., Lee, C.H., 1997. Minimum classification minimum error rate methods for speech recognition. IEEE Trans. Speech and Audio Processing 5, 257–265.

[9] Katagiri, S., Juang, B.H., Lee, C.H., 1998. Pattern recognition using a family of design algorithms based upon generalized probability descent method. Proc. IEEE 86 (11), 2345–2373.

[10] Kullback, S., 1968. Information Theory and Statistics. Dover, New York.

[11] Leonard, R.G., 1984. A database for speaker-independent digit recognition, in: Proc. IEEE ICASSP, San Diego, CA, USA. pp. 328–331.

[12] Li, J., Lee, C.H., 2008. On a generalization of margin-based discriminative training to robust speech recognition, in: INTERSPEECH, pp. 1992–1995.

[13] Li, J., Siniscalchi, S.M., Lee, C.H., 2007a. Approximate test risk minimization through soft margin estimation, in: Proc. ICASSP, Honolulu, Hawaii, USA. pp. IV–653–IV–656.

[14] Li, J., Yuan, M., Lee, C.H., 2006. Soft margin estimation of hidden markov model parameters, in: Proc. Interspeech, Pittsurgh, USA. pp. 2422–2425.

[15] Li, J., Yuan, M., Lee, C.H., 2007b. Approximate test risk bound minimization through soft margin estimation. IEEE Trans. Audio, Speech and Language Proc. 15 (8), 2393–2404.

[16] Pearce, D., Hirsch, H.G., 2000. The aurora experimental framework for the performance evaluation of speech recognition system under noisy condition, in: Proc. ICSLP, Beijing, China. pp. 29–32.

[17] Sha, F., Saul, L.K., 2007. Large margin hidden markov models for automatic speech recognition. Advances in Neural Information Processing Systems 19 B. Schlkopf, J.C. Platt, and T. Hofmann, Eds., MIT Press, 1249–1256.

[18] Xiao, X., Li, J., Chng, E., Li, H., Lee, C.H., 2010. A study on the generalization capability of acoustic models for robust speech recognition. IEEE Transactions on Audio, Speech & Language Processing 18, 1158–1169.

[19] Yu, D., Deng, L., He, X., Acero, A., 2006. Use of incrementally regulated discriminative margins in mce training for speech recognition, in: Proc. Interspeech, Pittsburgh, PA, USA. pp. 2418–2421.