Soft Margin Feature Extraction for Automatic Speech Recognition

Jinyu Li and Chin-Hui Lee

School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA. 30332 USA
{jinyuli, chl}@ece.gatech.edu

Abstract

We propose a new discriminative learning framework, called soft margin feature extraction (SMFE), for jointly optimizing the parameters of transformation matrix for feature extraction and of hidden Markov models (HMMs) for acoustic modeling. SMFE extends our previous work of soft margin estimation (SME) to feature extraction. Tested on the TIDIGITS connected digit recognition task, the proposed approach achieves a string accuracy of 99.61%, much better than our previously reported SME results. To our knowledge, this is the first study on applying the margin-based method in joint optimization of feature extraction and acoustic modeling. The excellent performance of SMFE demonstrates the success of soft margin based method, which targets to obtain both high accuracy and good model generalization.

Index Terms: discriminative feature extraction, hidden Markov model, margin, automatic speech recognition

1. Introduction

Feature exaction is an important component in the automatic speech recognition (ASR) systems. Most current ASR systems use Mel-frequency cepstrum coefficients (MFCCs) [1] as their standard input acoustic features. Usually, lower dimension MFCCs are got by applying DCT transformation on higher dimension log filter bank energies and following an optional cepstrum lifter. The DCT transformation matrix is fixed for all tasks. It has been demonstrated data dependent transformation is usually better than data independent one in classification tasks [2]. Therefore, the DCT transformation may not be the best choice for specific ASR tasks. We hope to get better features with more discriminative information, and these discriminative features will benefit our ASR backend.

Linear discriminant analysis (LDA) [2] may be a choice for the data dependent transformation. In [3][4], LDA has been successfully used in ASR systems for feature extraction. However, there are two major disadvantages of LDA. One is that LDA assumes the underlying sample distribution is Gaussian. The other is that it assumes that the class samples are of equal variance. Heteroscedastic discriminant analysis (HDA) [5] was proposed to remove the equal variance assumption. However, HDA is under the framework of maximum likelihood estimation (MLE). MLE is known to be optimal for density estimation, but it often does not lead to minimum recognition error that is the goal of ASR. As a remedy, several discriminative training (DT) methods have been proposed in recent years to boost the ASR system accuracy. Typical methods are maximum mutual information estimation (MMIE) [6], minimum classification error (MCE) [7], and minimum word/phone error (MWE/MPE) [8]. These DT technologies can also be applied to feature extraction. MCE was used in the work of discriminative feature extraction [9] to get optimal lifter and was used to adjust the artificial neural network based feature in [10]. Feature-space MPE was used to get discriminative feature in [11].

Inspired by the great success of margin-based classifiers in machine learning, there is a new trend to use margin to design new DT methods. Several speech recognizers based on margin maximization were proposed recently [12-16]. They already have shown advantages over some DT methods in some ASR tasks [12][14][15]. Among them, soft margin estimation (SME) [15][16] was proposed to make direct usage of the successful ideas of soft margin in support vector machines to improve generalization capability and decision feedback learning in minimum classification error training to enhance model separation in classifier design. In [16], SME was shown to be able to minimize the approximate test risk from the viewpoint of statistical learning theory [17]. As a result, SME has shown its superiority in several ASR tasks [15][16].

All the above mentioned margin-based methods focus on how to improve the generalization of acoustic models. In this study, we propose soft margin feature extraction (SMFE) to apply the idea of SME to jointly optimize the parameters of feature extraction transformation and the HMMs. Detailed derivation and implantation are given. SMFE shows a good improvement over SME. Tested on the TIDIGITS database [18], even 1-mixture model can achieve a string accuracy of 99.13%. The 16-mixture SMFE model attains a string accuracy of 99.61%. To our knowledge, it is the best result ever reported.

2. Soft Margin Feature Extraction

In this section, the theory of soft margin estimation is first briefly reviewed. Then we propose soft margin feature extraction to jointly optimize the feature transformation matrix and HMM parameters under the framework of SME. Because the purpose of feature extraction in this study is to find a transformation matrix to reduce the dimension of log filter bank energies, dimension reduction and feature extraction are interchanged frequently in this paper.

2.1 Soft Margin Estimation

Here, we briefly introduce SME. Please refer to [15][16] for detailed discussion. According to statistical learning theory [17], a test risk is bounded by the summation of two terms: an empirical risk (i.e., the risk on the training set) and a generalization function. The generalization function is a monotonic increasing function of Vapnik &

Chervonenkis dimension, or VC dimension (VC_{dim}) [17]. It can be shown that VC_{dim} is bounded by a decreasing function of the margin [17]. Hence, VC_{dim} can be reduced by increasing the margin. This is the key idea of the margin-based method.

As analyzed, we have two targets for optimization, one is to minimize the empirical risk, and the other is to maximize the margin. The test risk bound is approximated by combining these two targets into a single SME objective function:

$$L^{SME}(\Lambda) = \frac{\lambda}{\rho} + R_{emp}(\Lambda) = \frac{\lambda}{\rho} + \frac{1}{N}\sum_{i=1}^{N} \ell_i(O_i, \Lambda). \quad (1)$$

Λ denotes HMM parameters, $\ell_i(O_i, \Lambda)$ is a loss function for utterance O_i, and N is the number of training utterances. ρ is the soft margin, and λ is a coefficient to balance the soft margin maximization and the empirical risk minimization.

The loss function can be defined as:

$$\ell_i(O_i, \Lambda) = (\rho - d_i(O_i, \Lambda))_+ = \begin{cases} \rho - d_i(O_i, \Lambda), & \text{if } \rho - d_i(O_i, \Lambda) > 0, \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

with the separation of the models defined as:

$$d_i(O_i, \Lambda) = \frac{1}{n_i}\sum_j \log\left[\frac{p(O_{ij}|S_i)}{p(O_{ij}|\hat{S}_i)}\right] I(O_{ij} \in F_i), \quad (3)$$

where F_i is the frame set in which the frames have different labels in the competing strings. I is an indicator function, and O_{ij} is the jth frame for utterance O_i. n_i denotes the number of frames that have different labels in target and competing strings for utterance O_i. $p(O_{ij}|S_i)$ and $p(O_{ij}|\hat{S}_i)$ are the likelihood scores for the target string S_i and the most competitive string $\hat{S}_i$. Eq. (1) can be solved by using generalized probabilistic descent (GPD) [19] algorithm as stated in [15] to find HMM parameters, Λ.

2.2 SMFE for Gaussian Observations

The objective of feature exaction is to find a matrix W to transform the original n dimension feature vector x into a new d ($d<n$) dimension vector y. It is formulated as: $y=Wx$. To simplify the formula, we use x to stand for O_{ij}, which is used in Eqs. (3). To embed W into the framework of SMFE, we need to express $\log p(x|S_i) - \log p(x|\hat{S}_i)$ as a function of W.

We first investigate a simple case, in which the state observation probability is modeled by a Gaussian distribution.

An $n*n$ dimension matrix V is used to obtain $z=Vx$ (z is a n dimension vector). If the probability density function (pdf) of z is modeled by a Gaussian with (u_n, Σ_n) as the mean and covariance, the pdf of x can be expressed as:

$$p(x|u_n, \Sigma_n) = \frac{|V|}{(2\pi)^{n/2}|\Sigma_n|^{1/2}} \exp\left\{-\frac{1}{2}(Vx - u_n)^T \Sigma_n^{-1}(Vx - u_n)\right\}.$$

For the purpose of dimension reduction, only the first d dimension feature is needed. Therefore, the n dimension vector of z and u_n can be split into 2 sub-vectors with dimension d and n-d, respectively. Also, the $n*n$ matrix of V can be split into two matrixes with dimension $d*n$ and $(n-d)*n$, respectively. That is:

$$z = \begin{bmatrix} y \\ y_{n-d} \end{bmatrix}, \; u_n = \begin{bmatrix} u \\ u_{n-d} \end{bmatrix}, \text{ and } V = \begin{bmatrix} W \\ W_{(n-d)*n} \end{bmatrix}.$$

Block approximation for the covariance matrix is made as:

$$\Sigma_n = \begin{bmatrix} \Sigma & 0 \\ 0 & \Sigma_{n-d} \end{bmatrix},$$

where Σ and Σ_{n-d} are the matrixes with dimension $d*d$ and $(n-d)*(n-d)$, respectively.

Let (u_{n+}, Σ_{n+}) and (u_{n-}, Σ_{n-}) denote the means and covariance matrixes of Gaussians of the target and competing labels, respectively. The difference between the log values of two Gaussian distributions will be:

$$\begin{aligned}
&\log p(x|S_i) - \log p(x|\hat{S}_i) \\
&= \log p(x|u_{n+}, \Sigma_{n+}) - \log p(x|u_{n-}, \Sigma_{n-}) \\
&= \log|V| - \frac{1}{2}\log|\Sigma_{n+}| - \frac{1}{2}(Vx - u_{n+})^T \Sigma_{n+}^{-1}(Vx - u_{n+}) \\
&\quad - \left\{\log|V| - \frac{1}{2}\log|\Sigma_{n-}| - \frac{1}{2}(Vx - u_{n-})^T \Sigma_{n-}^{-1}(Vx - u_{n-})\right\} \\
&= \frac{1}{2}\log|\Sigma_{n-}| + \frac{1}{2}(Vx - u_{n-})^T \Sigma_{n-}^{-1}(Vx - u_{n-}) \\
&\quad - \frac{1}{2}\log|\Sigma_{n+}| - \frac{1}{2}(Vx - u_{n+})^T \Sigma_{n+}^{-1}(Vx - u_{n+}) \\
&= R + R_{n-d}
\end{aligned}$$

$$R = \frac{1}{2}\log|\Sigma_-| + \frac{1}{2}(Wx - u_-)^T \Sigma_-^{-1}(Wx - u_-) - \frac{1}{2}\log|\Sigma_+| - \frac{1}{2}(Wx - u_+)^T \Sigma_+^{-1}(Wx - u_+)$$

$$R_{n-d} = \frac{1}{2}\log|\Sigma_{(n-d)-}| + \frac{1}{2}(W_{(n-d)*n}x - u_{(n-d)-})^T \Sigma_{(n-d)-}^{-1}(W_{(n-d)*n}x - u_{(n-d)-}) - \frac{1}{2}\log|\Sigma_{(n-d)+}| - \frac{1}{2}(W_{(n-d)*n}x - u_{(n-d)+})^T \Sigma_{(n-d)+}^{-1}(W_{(n-d)*n}x - u_{(n-d)+})$$

If $\frac{1}{|\Sigma_{n-d}|^{1/2}} \exp\left\{-\frac{1}{2}(W_{(n-d)*n}x - u_{n-d})^T \Sigma_{n-d}^{-1}(W_{(n-d)*n}x - u_{n-d})\right\}$ is assumed to be a constant, then R_{n-d} is equal to 0. This assumption makes sense, because the target of dimension reduction is to discard unnecessary information and this additional dimension cannot be removed if the n-d dimension features of different classes are very different. Similar assumption is used in HDA [5]. Now, we only need to concern about R, which is a function of W. Therefore, in Eq. (3), we use R to replace $\log p(O_{ij}|S_i) - \log p(O_{ij}|\hat{S}_i)$.

2.3 SMFE for GMM Observations

For the case that the state observation probability is modeled by a GMM, we have

$$\ln p(x|c, u_n, \Sigma_n) = \log|V| + \log\left(\sum_j \frac{c_j}{(2\pi)^{n/2}|\Sigma_{jn}|^{1/2}} \exp\left\{-\frac{1}{2}(Vx - u_{jn})^T \Sigma_{jn}^{-1}(Vx - u_{jn})\right\}\right)$$

Here, $(c_j, u_{jn}, \Sigma_{jn})$ are the weight, mean, and covariance matrix of the jth Gaussian component of GMM. The $\log|V|$ item can still be removed by the subtraction of two

log values of state observation probabilities of the target and competing class. By applying the same splitting strategy for each mixture component of GMMs as Section 2.2 and making the constant assumption for

$$\frac{1}{\left|\Sigma_{j,n-d}\right|^{1/2}}\exp\left\{-\frac{1}{2}(W_{(n-d)*n}x-u_{j,n-d})^T\Sigma_{j,n-d}^{-1}(W_{(n-d)*n}x-u_{j,n-d})\right\},$$

the difference of those log values can be written as:

$$\log p(x|S_i)-\log p(x|\hat{S}_i)=R$$

$$=\log\left(\sum_j\frac{c_{j+}}{\left|\Sigma_{j+}\right|^{1/2}}\exp\left\{-\frac{1}{2}(Wx-u_{j+})^T\Sigma_{j+}^{-1}(Wx-u_{j+})\right\}\right). \quad (4)$$

$$-\log\left(\sum_j\frac{c_{j-}}{\left|\Sigma_{j-}\right|^{1/2}}\exp\left\{-\frac{1}{2}(Wx-u_{j-})^T\Sigma_{j-}^{-1}(Wx-u_{j-})\right\}\right)$$

where $(c_{j+},u_{j+},\Sigma_{j+})$ and $(c_{j-},u_{j-},\Sigma_{j-})$ are the weight, mean, and covariance matrix of the *j*th component of GMM for target and competing classes, respectively. We can also replace Eq. (4) into Eqs. (1)-(3).

2.4 Implementation of SMFE

In our implementation of SMFE, we simplify the process by using the same transformation matrix *W* for the static, first and second order time derivatives of the log filter bank energies. Let *x* denote the static log filter bank energies, Δx and $\Delta\Delta x$ denote the first and second order derivatives of *x*. Then the new transformed static feature vector is given by: $y=Wx$, and the dynamic features of *y* are: $\Delta y=W\Delta x$ and $\Delta\Delta y=W\Delta\Delta x$. The final feature *Q* is composed of *y*, Δy, $\Delta\Delta$y, log energy *e* and its derivatives Δe, $\Delta\Delta e$ as Q = (Wx $W\Delta x$ $W\Delta\Delta x$ e Δe $\Delta\Delta e$)^T. Then, *R* in Eq. (4) can be expressed:

$$R=\log\left(\sum_j\frac{c_{j+}}{\left|\Sigma_{j+}\right|^{1/2}}\exp\left\{-\frac{1}{2}(Q-u_{j+})^T\Sigma_{j+}^{-1}(Q-u_{j+})\right\}\right)$$

$$-\log\left(\sum_j\frac{c_{j-}}{\left|\Sigma_{j-}\right|^{1/2}}\exp\left\{-\frac{1}{2}(Q-u_{j-})^T\Sigma_{j-}^{-1}(Q-u_{j-})\right\}\right) \quad (5)$$

Eq. (5) is a function of the matrix *W*. Now, we can embed *R* into Eqs. (1)-(3) to replace $\log p(O_{ij}|S_i)-\log p(O_{ij}|\hat{S}_i)$ and use GPD to get the final parameters of transformation matrix *W* and all the HMM parameters.

3. Experiment

The proposed framework was evaluated on the TIDIGITS database. 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 MLE HMMs. We used 11 whole-digit HMMs, one for each of the 10 English digits, plus the word "oh". Each HMM has 12 states and each state observation density is characterized by a mixture Gaussian density. Models with 1, 2, 4, 8, and 16 mixture components were trained. The input features were 12MFCCs + energy, and their first and second order time derivatives. Our SME models were initiated with the MLE models. Digit decoding was based on unknown length without imposing any language model or insertion penalty. The MLE and SME models trained with MFCCs are denoted as MLE_M and SME_M in Table 1.

In parallel, we used LDA to extract the acoustic features. For each speech frame, we have 24 log filter bank energies. LDA was applied to reduce the dimension from 24 to 12. To get the LDA transformation, each HMM-state was chosen as a class. This dimension reduced feature is concatenated with energy, and then extended with the first and second order derivatives to form a new 39-dimension feature. We also trained MLE and SME models based on this new LDA-based feature. These models are MLE_L and SME_L in Table 1.

Finally, initiated with the models MLE_L and the LDA transformation matrix, SMFE models were trained to get the optimal HMM parameters and transformation matrix. The results of SMFE are also listed in Table 1.

Only string accuracies of the TIDIGITS testing set 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. It is clear that LDA-based feature outperforms MFCCs for both the MLE and SME models. Shown in the last column of Table 1, SMFE models achieved the best performance. Even 1-mixture SMFE model can get better performance than 16-mixture MLE models with MFCCs or LDA-based features. SMFE got 99.61% string accuracy for 16-mixture model. This is a large improvement from original SME work [15], in which MFCCs were used. The excellent SMFE performance is attributed to the joint optimization of acoustic feature and HMM parameters by directly using soft margin to improve generalization capability, and using decision feedback learning to enhance model separation in classifier design.

Table 1: String accuracy comparison with different methods on the TIDIGITS testing set

	MLE_M	SME_M	MLE_L	SME_L	SMFE
1-mix	95.20% *	98.76% *	96.82% *	98.91% *	99.13%
2-mix	96.90% *	98.95% *	97.82% *	99.15% *	99.36%
4-mix	97.80% *	99.20% *	98.51% *	99.31%	99.44%
8-mix	98.03% *	99.29% *	98.63% *	99.39% *	99.56%
16-mix	98.36% *	99.30% *	98.93% *	99.46% *	99.61%

For a high accuracy task such as TIDIGITS, it is interesting to test whether SMFE is significantly better than other methods. For each mixture model setting, we denote p_1 as the accuracy of SMFE model, and denote p_2 as the accuracy for other models from MLE_M, SME_M, MLE_L, and SME_L. If we consider p_1 and p_2 are independent, then we have the following hypothesis testing problem [20] with H_0: $p_1 = p_2$ against H_1: $p_1 > p_2$. We can get the statistic: $z=\frac{\sqrt{N}(p_1-p_2)}{\sqrt{p_1(1-p_1)}+\sqrt{p_2(1-p_2)}}$, where *N*

denotes the total number of samples (8700 here).

We make decision according to the following:

$$\begin{cases} accept\ H_0 & if \quad z < Z_\alpha \\ reject\ H_0 & if \quad z > Z_\alpha \end{cases}. \tag{6}$$

Z_α is called upper α quantile. Here, we set α to 0.1. If hypothesis H_0 is rejected, SMFE is significantly better at the confidence level of 90%. For every mixture setting, the hypothesis testing was performed according to Eq. (6). In Table 1, an asterisk is used to denote when the performance of SMFE is significantly better. It is shown that SMFE is significantly better than nearly all the other models at the significance level of 90%. The only exception is the 4-mixture SME_L model.

4. Conclusion

By extending our previous work of SME, we proposed a new discriminative training method, called SMFE, to achieve even higher accuracy and better model generalization. By jointly optimizing the acoustic feature and HMM parameters under the framework of SME, SMFE performs much better than SME, and significantly better than MLE. Tested on the TIDIGITS database, even 1-mixture model can get string accuracy of 99.13%. And 99.61% string accuracy was got with 16-mixture SMFE model. This is a great improvement comparing to our original SME work that uses MFCCs as acoustic feature. This SMFE work again demonstrates the success of soft margin based method, which directly makes 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.

This paper only presents our initial study. We are now working on many related research issues to further complete the work of SMFE. In this study, feature transformation matrix only works on the static log filter bank energies of the current frame. In [5], great benefits were obtained by using the frames in context before and after the current frame. We will try to incorporate these context frames into SMFE optimization. Secondly, in [16] we have shown that SME also worked well on large vocabulary continuous speech recognition task. We will try to demonstrate the effectiveness of SMFE on the Wall Street Journal task in future work.

5. Acknowledgements

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.

6. References

[1] Davis, S. and Mermelstein, P., "Comparison of parametric representations for monosyllable word recognition in continuously spoken sentences," *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol. 28, no.4, pp. 357-366, 1980.

[2] Duda, R. O., Hart, P. E., and Stork, D. G., *Pattern Classification*, 2nd ed. John Wiley & Sons, 2001.

[3] Hunt, M. J. and Lefebvre, C. "A comparison of several acoustic representations for speech recognition with degraded and undegraded speech." *Proc. ICASSP*, pp. 262-265, 1989.

[4] Haeb-Umbach, R., Geller, D., and Ney, H. "Improvement in connected digit recognition using linear discriminant analysis and mixture densities." *Proc. ICASSP,* pp. 239-242, 1993.

[5] Kumar, N. and Andreou, A. G., "Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition," *Speech Communcation*, vol. 26, pp. 283–297, 1998.

[6] Bahl, L. R., Brown, P. F., de Souza, P. V., and Mercer, R. L., "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," *Proc.ICASSP*, pp.I49-I52, 1986.

[7] 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.

[8] Povey, D. and Woodland, P. C., "Minimum phone error and I-smoothing for improved discriminative training," *Proc. ICASSP*, vol. 1, pp. 105-108, 2002.

[9] Biem, A., Katagiri, S., and Juang, B. -H., "Pattern recognition using discriminative feature extraction." *IEEE Trans. Signal Processing*, vol. 45, no. 2, pp. 500-504, 1997.

[10] Rahim, M. and Lee, C. -H., "Simultaneous Feature and HMM Design Using String-Based Minimum Classification Error Training Criterion," *Proc. ICSLP-96,* pp. 1820-1823, 1996.

[11] Povey, D., Kingsbury, B., Mangu, L., Saon, G., Soltau, H., and Zweig, G., "FMPE: discriminatively trained features for speech recognition," *Proc. ICASSP*, pp. I961-I964, 2005.

[12] Li, X. and Jiang, H., "Solving large margin estimation of HMMs via semidefinite programming," *Proc. Interspeech,* pp. 2414-2417, 2006.

[13] Sha, F. and Saul, L. K., "Large margin hidden Markov models for automatic speech recognition," *Advances in Neural Information Processing Systems 19*, Sch¨olkopf, B., Platt, J. C., and Hofmann, T., Eds., MIT Press, 2007.

[14] Yu, D., Deng, L., He, X., and Acero, A., "Use of incrementally regulated discriminative margins in MCE training for speech recognition," *Proc. Interspeech,* pp. 2418-2421, 2006.

[15] Li, J., Yuan, M., and Lee, C. -H., "Soft margin estimation of hidden Markov model parameters," *Proc. Interspeech*, pp.2422-2425, 2006.

[16] Li, J., Siniscalchi, M., and Lee, C. -H., "Approximate test risk minimization through soft margin estimation," *Proc. ICASSP*, 2007.

[17] Vapnik, V., *The Nature of Statistical Learning Theory,* Springer-Verlag, New York, 1995.

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

[19] 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.

[20] Lehmann, E. L., *Testing Statistical Hypothesis,* 2nd ed. Wiley, 1986.