I-SMOOTH FOR IMPROVED MINIMUM CLASSIFICATION ERROR TRAINING

Haozheng Li, Cosmin Munteanu

National Research Council of Canada,
{Haozheng.Li, Cosmin.Munteanu}@nrc-cnrc.gc.ca

ABSTRACT

Increasing the generalization capability of Discriminative Training (DT) of Hidden Markov Models (HMM) has recently gained an increased interest within the speech recognition field. In particular, achieving such increases with only minor modifications to the existing DT method is of significant practical importance. In this paper, we propose a solution for increasing the generalization capability of a widely-used training method – the Minimum Classification Error (MCE) training of HMM – with limited changes to its original framework. For this, we define *boundary* data – obtained by applying a large steep parameter, and *confusion* data – obtained by applying a small steep parameter on the training samples, and then do a soft interpolation between these according to the number points of occupancies of *boundary* data and the number points ratio between the *boundary* and the *confusion* occupancies. The final HMM parameters are then tuned in the same manner as in MCE by using the interpolated *boundary* data. We show that the proposed method achieves lower error rates than a standard HMM training framework on a phoneme classification task for the TIMIT speech corpus.

Index Terms— Hidden Markov Model, Speech Recognition, Minimum Classification Errors

1. INTRODUCTION

Discriminative training (DT) has been widely used in speech recognition and it has proved to give significant improvement over the traditional maximum likelihood estimation (MLE) method in many speech recognition tasks. The most common DT methods are minimum classification error (MCE) [1] [2] and maximum mutual information (MMI) [3] [4] (or its variant minimum phone error (MPE) [5]) training. Both these DT methods focus on reducing the empirical error in the training set. However, optimal performance on the training set does not guarantee the same on the test set – the well-known overfitting problem.

Recent research on discriminative training has focused on improving the generalization capability of discriminative training of Hidden Markov Models (HMM). For example, large-margin methods used in conjuction with HMMs (such as in [8], [9] and [7]) can reduce the test risks. However, these represent a significant departure from the existing DT framework. While large-margin methods that preserve the framework do exist, such as the MCE parameter tuning approach introduced in [6], they are affected by practical limitations.

The existing DT has been the focus of research for several years, and its practical use has been proven by its successful incorporation into many commercial speech recognition systems. Unfortunately, one of the more important drawbacks of DT – the overfitting problem – seems to be addressed only through solutions that are either a departure from the DT framework, or have limited practical applicability. In this paper, we propose a method that could increase the generalization capability of DT, with only minor modifications to the existing DT method, while maintaining its practical applicability.

Interpolation between MLE and the discriminative objective functions has been applied in MMIE and MPE in a manner that depends on the amount of data available for each Gaussian (in the case of soft interpolation [5]) or has been directly applied as hard interpolation [10]. In this paper, we use I-smoothing to interpolate between two MCEs with different steepness parameters, as opposed to interpolating between MLE and MCE.

In the following section we briefly review the MLE and MCE training. We describe our proposed I-smooth method in Section 3 and show our experimental results in Section 4, followed by a discussion in Section 5.

2. MLE AND MCE

In an HMM with N underlying states, a state sequence $S = (s_1, s_2, \cdots, s_T)$ generated by the Markov chain cannot be directly observed. Only observation sequences $Y = (y_1, y_2, \cdots, y_T)$ resulted from the state sequence are visible, according to the observation distribution defined by $B = \{b_i(y_t) : 1 \leq i \leq N\}$, with $b_i(y_t) = P(y_t|s_t = i)$, which often takes a Gaussian mixture form. The transition from state i to state j is specified by an $N \times N$ matrix $A = [A_{ij}]$ with $A_{ij} = P(s_t = j|s_{t-1} = i)$. $\pi = [\pi_1, \pi_2, \cdots, \pi_N]$ is the initial state probability vector with $\pi_i = P(s_1 = i)$. Λ is the compact notation for the model parameters in an HMM A, B and π. Maximum likelihood estimation (MLE) opti-

978-1-4244-4296-6/10/$25.00

mizes A, B and π to maximize the probability of observation sequences in the training set. By defining the a posteriori probability variable

$$\gamma_t\left(i\right) = P\left(s_t = i | \Lambda, Y\right)$$

which is the probability of being in state i at time t, given the observation sequence Y and the model Λ, the Baum-Welch algorithm uses it to do soft alignment of the training utterance and assigns the aligned speech to the hidden states to adjust the model's parameters in an iterative procedure.

MCE aims to minimize the number of smoothed empirical errors. Compared with MLE training, this discriminative method makes use of a competing model's additional information to train a model. In particular, assume utterance Y is aligned to the correct state sequence and to its competing incorrect state sequences. Let $g_{\text{cor}}\left(Y, \Lambda\right)$ denote the score on the correct model and $g_{\text{com}}\left(Y, \Lambda\right)$ denote the score on its competing model. We use $d(Y)$ to denote the misclassification measure:

$$d\left(Y\right) = -g_{\text{cor}}\left(Y, \Lambda\right) + g_{\text{com}}\left(Y, \Lambda\right).$$

For a specific Gaussian component with mean μ and the variance σ, let $\mathcal{O}$ denote the normalized speech $\frac{y-\mu}{\sigma}$ assigned to this component. As such, $\mathcal{O}^2$ will be $\left(\frac{y-\mu}{\sigma}\right)^2$. We may rewrite the MCE reestimation formula from [1] as:

$$\begin{aligned}\mu^{\text{mce}} &= \mu + \varepsilon\left(\theta^{\text{mce}}_{\text{cor}}\left(\mathcal{O}\right) - \theta^{\text{mce}}_{\text{com}}\left(\mathcal{O}\right)\right)\\ \sigma^{\text{mce}} &= \sigma \cdot \exp\left(\varepsilon\left(\theta^{\text{mce}}_{\text{cor}}\left(\mathcal{O}^2 - 1\right) - \theta^{\text{mce}}_{\text{com}}\left(\mathcal{O}^2 - 1\right)\right)\right)\end{aligned}$$

where μ^{mce} is the new mean and σ^{mce} is the new variance. ϵ is the learning rate; subscript cor and com indicate that the state this component belongs to locates in correct and competing state sequence respectively. $\theta(\mathcal{O})$ indicates that normalized speech $\mathcal{O}$ is weighted by a function L. For the utterance Y, the weighting function is:

$$\begin{aligned}L(d(Y)) &= l^{\text{mce}}(d(Y))(1 - l^{\text{mce}}(d(Y)))\\ l^{\text{mce}}(d(Y)) &= 1/(1 + \exp(-rd(Y)))\end{aligned}$$

where r is the steepness parameter.

Let's take a look at the $L(d(Y))$ whose curve is shown in Fig. 1. If the steepness parameter r is very large, the curve of $L(d(Y))$ is sharp and only has large values when $d(Y)$ is close to 0. In this case, it will only select those utterances that locate in the score boundary area in order to tune the model parameters. As such, we classify the speech data selected by $L(d(Y))$ as *boundary* data. In theory, only if the steepness parameter r is large enough, (ideally, close to $+\infty$), $l^{\text{mce}}(d(Y))$ will accurately measure the misclassification number. In such cases, however, *boundary* data may be too small. On the other hand, if we choose a very small steepness parameter r, say, close to 0, then $L(d(Y))$ is quite flat and selects a wider range of data. At this time, $l^{\text{mce}}(d(Y))$ cannot measure the true misclassification number accurately. In reality, a moderate value such as 1.0 can be used for the steepness parameter, even though the *boundary* data accounts for a very small portion of the whole speech sample. If the number of mixture components is large, which is usually necessary to better fit the training data, the *boundary* data assigned to each gaussian may be very small and may not reflect the true data distribution.

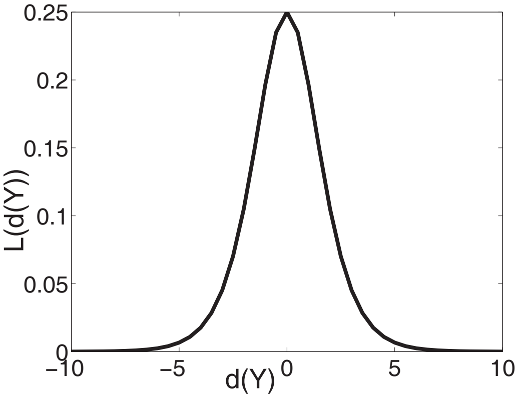

Fig. 1. A weighting function

Although the model parameter can be updated after each utterance is feed in (online optimization), the method proposed in this paper relies instead on the number of occupancies of *boundary* data accumulated in each Gaussian component, so the batch oriented optimization will be used. We need to define $\phi^{\text{mce}}_{\text{cor}}\left(\mathcal{O}\right)$ and $\phi^{\text{mce}}_{\text{com}}\left(\mathcal{O}\right)$, which denote the number of points of $\theta^{\text{mce}}_{\text{cor}}\left(\mathcal{O}\right)$ and $\theta^{\text{mce}}_{\text{com}}\left(\mathcal{O}\right)$ respectively. Batch optimization has a similar performance to online optimization [2].

Beside the use of batch updates, we should also mention the following implementation details for the baseline MCE:
(1) Average frame misclassification measure $d(Y)/T$ is used instead of $d(Y)$ to help the tuning process.
(2) We use a chopped sigmoid,

$$l^{\text{mce}}(d(Y)/T) = \left\{ \begin{array}{ll} 0, & \left|d\left(Y\right)/T\right| > Q \\ 1/\left(1 + \exp\left(-rd(Y)/T\right)\right), & \text{otherwise} \end{array}\right.$$

where Q is a positive scalar. Removing correctly recognized utterances with d values larger than Q does not cause a loss, while cases with d smaller than $-Q$ are regarded as outliers and will not be used to tune the model parameters. Within $[-Q, Q]$ only up to eight best competitors are considered.
(3) Variable learning rates $\xi/(\phi^{\text{mce}}_{\text{cor}}\left(\mathcal{O}\right) - \phi^{\text{mce}}_{\text{com}}\left(\mathcal{O}\right))$ are used when $|(\phi^{\text{mce}}_{\text{cor}}\left(\mathcal{O}\right) - \phi^{\text{mce}}_{\text{com}}\left(\mathcal{O}\right))|$ is large, where ξ is a small positive scalar. A fixed small learning rate is used when $|(\phi^{\text{mce}}_{\text{cor}}\left(\mathcal{O}\right) - \phi^{\text{mce}}_{\text{com}}\left(\mathcal{O}\right))|$ is less than a threshold.

3. I-SMOOTH METHOD ON MCE

I-smooth has been introduced into MMIE and MPE to improve their generalization capability [5, 6]. In the context of

MMIE, I-smoothing means increasing the number of Gaussian occupancies, but keeping invariant the average data values and average squared data values. In the context of MPE training, it directly adds MLE occupancies to the numerator occupancies and then use them in MPE training. Both are a way of applying an interpolation between MLE and a discriminative objective function, which depends on the amount of data available for each Gaussian component.

We propose a different strategy to deal with the problem of over-training in that the interpolation is applied between the occupancies obtained in MCE with different steepness parameters rather than interpolation with MLE.

To describe the proposed method, we use misclassification measure $d(Y)$ to define another data selection criteria:

$$l^{\mathrm{con}}(d(Y)/T) = \{ \begin{array}{ll} 0, & |d\left(Y\right)/T| > Q \\ 1, & \text{otherwise} \end{array}$$

Compared to $l^{\mathrm{mce}}(d(Y)/T)$, only the chop on $d(Y)$ is applied while the sigmoid is not being used. The selected data are close to the boundary area and may cause some confusion during training; as such, the selected speech data is referred to as *confusion* data.

Assuming the *confusion* data is aligned to the correct state sequence and its competing incorrect state sequences, we may get another set of accumulators for the specific Gaussian component: $\theta^{\mathrm{con}}_{\mathrm{cor}}$, $\theta^{\mathrm{con}}_{\mathrm{com}}$, $\phi^{\mathrm{con}}_{\mathrm{cor}}$, and $\phi^{\mathrm{con}}_{\mathrm{com}}$. They have same the meaning as their corresponding expressions in MCE described in the previous Section, except that they are used for the *confusion* data.

Because the *boundary* data is obtained by applying the weighting function L to the *confusion* data, they have some inherent relations. To explain this kind of relation, we define

$$\begin{array}{l} Bound\_OCC = \phi^{\mathrm{mce}}_{\mathrm{cor}}\left(\mathcal{O}\right) + \phi^{\mathrm{mce}}_{\mathrm{com}}\left(\mathcal{O}\right) \\ Conf\_OCC = \phi^{\mathrm{con}}_{\mathrm{cor}}\left(\mathcal{O}\right) + \phi^{\mathrm{con}}_{\mathrm{com}}\left(\mathcal{O}\right) \end{array}$$

and their relation is as follows:
(1) In theory, the *boundary* data define a more accurate score boundary, but because of the lack of availability, it may not represent the true score boundary data distribution.
(2) On the other side, $Conf\_OCC$ is much larger than $Bound\_OCC$, however, the *confusion* data define a less accurate score boundary.

If we could take advantage of the *boundary* data and the *confusion* data and find an appropriate interpolation between them, it is possible to better model the score boundary data distribution and thus increase the generalization of the MCE. To decide which Gaussian component need to be interpolated, we make two assumptions:
(1) If a component has $Bound\_OCC$ larger than a certain threshold, we assume its assigned *boundary* data could reflect the true boundary data distribution and does not need to be interpolated.
(2) Assume $Conf\_OCC$ is less affected by some random factor and the ratio between $Bound\_OCC$ and $Conf\_OCC$ should locate in a certain range for each Gaussian component. We use $Conf\_OCC$ as the reference and if the ratio is much less than the bottom of the certain range, the *boundary* data need to be interpolated with the *confusion* data.

To deal with the ratio value varying across different iterations, individual ratios are normalized by the ratio of total $Bound\_OCC$ and $Conf\_OCC$ of all Gaussian components.

We adopt a soft interpolation between the *boundary* and the *confusion* occupancies and the smooth factor is:

$$\omega = 1.0 - \exp(-A \times Boundary\_OCC - B \times Ratio + C)$$

which is shown in Fig. 2.

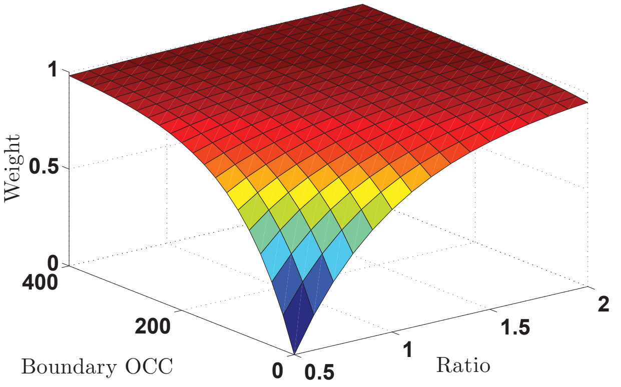

Fig. 2. Smooth Weight

Then the accumulators for $\mathcal{O}$ and $(\mathcal{O}^2 - 1)$ are smoothed:

$$\begin{array}{l} \theta^{'}_{\mathrm{cor}} - \theta^{'}_{\mathrm{com}} = \omega\left(\theta^{\mathrm{mce}}_{\mathrm{cor}} - \theta^{\mathrm{mce}}_{\mathrm{com}}\right) \\ \qquad + (1-\omega)\,\frac{\theta^{\mathrm{con}}_{\mathrm{cor}} - \theta^{\mathrm{con}}_{\mathrm{com}}}{\phi^{\mathrm{con}}_{\mathrm{cor}} - \phi^{\mathrm{con}}_{\mathrm{com}}}\left(\phi^{\mathrm{mce}}_{\mathrm{cor}} - \phi^{\mathrm{mce}}_{\mathrm{com}}\right) \end{array}$$

and the mean and the variance will be updated by $\theta^{'}_{\mathrm{cor}}$ and $\theta^{'}_{\mathrm{com}}$ in a way same to that in MCE.

4. EXPERIMENTAL RESULTS

Although the TIMIT phone classification is a much easier task than large-scale speech recognition, it is a useful benchmark to test the effectiveness of new discriminative training methods without the influence of other factors such as language models or noisy environments. We have carried out several experiments to test our model on a speech phoneme classification task, i.e., assuming the segmentation time is known, the task will recognize the unigram and context-independent phones. Our training and testing sets are created with the 'sx' and 'si' training and testing sentences from TIMIT with 3696 and 192 core test sets, respectively. The standard phonetic clustering [11] was used, resulting in 48 phone models and further mapping to 39 phones during the test. We use HTK as the baseline, in which the three-state left-to-right models are adopted. We use 12 Mel Cepstral coefficients, plus the energy parameter, and their first and second order difference as the output feature. Thus the total dimension of the feature vector is 39. These parameters are derived from 25 ms long window frames with a 10 ms shift rate. The numbers of Gaussian component for each state is 32. Each mixture component covariance is modelled as a diagonal matrix; no parameter-tying is applied to different states or mixture components.

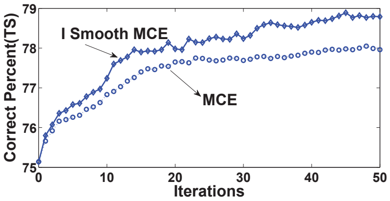

Fig. 3. MCE and I-MCE results on the test set

The correct percentages on the test set for both methods after each iteration are shown in Figure 3. For the MCE, our best result is 78.05%; and for the I-Smooth MCE, our best result is 78.89%. The average gain over MCE across the iterations is 0.56%.

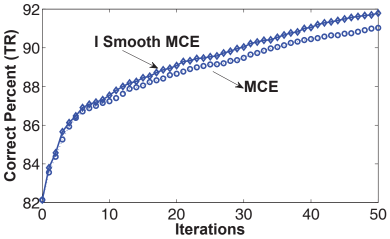

Fig. 4. MCE and I-MCE results on the training set

The I-Smooth MCE also give a higher correct percentage than the MCE on the training set shown in Figure 4. However, the average gain over MCE across the iterations is 0.46%, which is less than that on the test set. This suggests that the I-smooth method could increase the generalization capability for MCE as well as help reduce the MCE empirical risk.

5. CONCLUSIONS AND FUTURE WORK

In this paper we propose a method for applying interpolation between the *boundary* data and the *confusion* data, aimed to alleviate the overfitting problem in MCE. In theory, *Boundary* data define a more accurate score boundary, although it is affected more by random factors. *Confusion* is a wider range of *boundary* data that defines a rough score boundary while being less affected by random factors. We interpolate the *boundary* data with the *confusion* data when the number of the *boundary* occupancy is small and when the number ratio between them is small. The I-smooth MCE yields a 0.84% absolute lower phone error rate than our best MCE result without I-smoothing. The classification error rate of 21.11% is the best result known to us for a standard HMM on this well-known TIMIT phoneme classification task. Our future work will investigate the application of this technique to a large vocabulary speech recognition task.

6. REFERENCES

[1] B.-H. Juang, W. Chou and C.-H. Lee, "Minimum classification error rate methods for speech recogntion," *IEEE Tran. SAP*, vol. 5, no. 3, pp. 257–265, 1997.

[2] E. McDermott, T. J. Hazen, J. Le Roux, A. Nakamura and S. Katagiri, "Discriminative Training for Large-Vocabulary Speech Recognition Using Minimum Classification Error," *Proc. ASLP*, pp. 203–223, 2007.

[3] L. R. Bahl, P. V. de Souza P. F. Brown and R. L. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," *Proc. ICASSP*, pp. 49–52, 1986.

[4] Y. Normandin, R. Cardin and R. Demori, "High-performance connected digit recognition using maximum mutual information estimation," *IEEE Tran. SAP*, vol. 2, no. 2, pp. 299–311, 1994.

[5] D. Povey and P. C. Woodland, "Minimum phone error and I-smooth for improved discriminative training," *Proc. ICASSP*, pp. 105–108, 2002.

[6] D. Yu, L. Deng, X. He and A. Acero, "Large-margin minimum classification error training for large-scale speech recognition tasks," *Proc. ICASSP*, pp. 1137–1140, 2007.

[7] X. Li, H. Jiang and C. Liu, "Large margin HMMs for speech recognition," *Proc. ICASSP*, pp. 513–516, 2005.

[8] F. Sha and L. Saul, "Large-margin Gaussian mixture modeling for phonetic classification and recognition," *Proc. ICASSP*, pp. 265–268, 2006.

[9] J. Li, M. Yuan and C. H. Lee, "Approximate test risk bound minimization through soft margin estimation," *IEEE Trans. on Audio, Speech, and Language Proc.*, vol. 15, no. 8, pp. 2393–2404, 2007.

[10] P.S. Gopalakrishnan, D. Kanevsky, A. Nadas D. Nahamoo and M.A. Picheney, "Decoder Selection Based on Cross-Entropies," *Proc. ICASSP*, pp. 20–23, 1988.

[11] K. Lee and H. Hon, "Speaker-independent phone recognition using hidden Markov models," *IEEE Tran. ASSP*, vol. 37, pp. 1641–1648, 1989.