A CONSTRAINED LINE SEARCH APPROACH TO GENERAL DISCRIMINATIVE HMM TRAINING

Peng Liu[1]*, Cong Liu*[2]*, Hui Jiang*[3]*, Frank K. Soong*[1]*, and Ren-Hua Wang*[2]

[1]Microsoft Research Asia, Beijing, P. R. China, 100080
[2]University of Science and Technology of China, Hefei, P. R. China, 230027
[3]Department of Computer Science and Engineering, York University, Canada
{pengliu,frankkps}@microsoft.com, yylhbt@ustc.edu, hj@cse.yorku.ca, rhw@ustc.edu.cn

ABSTRACT

Recently, we proposed a novel optimization algorithm called *constrained line search (CLS)* to train Gaussian mean vectors of HMMs in the MMI sense. In this paper, we extend and re-formulate it in a more general framework. The new CLS can optimize any discriminative objective functions including MMI, MCE, MPE/MWE etc. Also, closed-form solutions to update all Gaussian mixture parameters, including means, covariances and mixture weights, are obtained. We investigate the new CLS on several benchmark speech recognition databases, including TIDIGITS, Switchboard mini-train and Switchboard full h5train00 sets. Experimental results show that the new CLS optimization method outperforms the conventional EBW method in both performance and convergence behavior.

Index Terms— Discriminative training, Optimization algorithm, Line search, Kullback-Leibler divergence

1. INTRODUCTION

In past few years, discriminative training (DT) has been a very active research area in automatic speech recognition (ASR). Most discriminative training methods have been formulated to estimate parameters of Gaussian mixture continuous density hidden Markov models (CDHMM) in different speech recognition tasks, ranging from small vocabulary, isolated word recognition to large vocabulary, continuous speech recognition tasks. Discriminative training is a typical optimization problem, where an objective function is optimized, usually in an iterative manner. Popular DT criteria including maximum mutual information (MMI)[1], minimum classification error (MCE)[2], minimum word or phone error (MWE or MPE) [8], minimum divergence (MD)[6], etc. Once the objective function is chosen, an effective algorithm is used to optimize the objective function by adjusting CDHMM parameters. In speech recognition, various algorithms have been proposed to optimize the objective function, including the generalized probabilistic descent (GPD) algorithm based on the first-order gradient descent, the approximate second-order, Hessian based Quickprop method, and the extended Baum-Welch (EBW) algorithm, etc. The GPD and Quickprop methods are mainly used for optimizing the MCE objective function. The EBW method has been initially proposed to optimize a rational objective function and later extended to Gaussian mixture CDHMMs for the MMI and MPE (or MWE) objective functions. Recently, the EBW method has also been generalized for optimizing the MCE objective function [9] as well as the MD objective function [6]. Nowadays, the EBW method has been widely accepted for discriminative training because it is relatively easy to implement the EBW algorithm on word graphs for large scale ASR tasks and it has been demonstrated that the EBW algorithm performs quite well on many ASR tasks.

Recently, we proposed a novel optimization method, called *constrained line search (CLS)*, to optimize Gaussian mean vectors for discriminative training, based on the MMI criterion[7]. We cast discriminative training of CDHMMs as a constrained optimization problem, where a constraint is explicitly imposed for DT based on the Kullback-Leibler divergence (KLD) between model parameters. The constraint is motivated by the fact that all collected estimation statistics are only reliable in a close neighborhood of the original model. Under this constraint, the objective function can be approximated as a smooth function of CDHMM parameters and its sole critical point, if existing, can be easily obtained by setting the derivative to zero. Then, a novel *constrained line search (CLS)* algorithm is proposed to solve the constrained optimization problem. Subject to the KLD constraint, the line search is performed either along a line segment joining the initial model parameters and the critical point of the smoothed objective function, if the critical point exists, or along gradient direction of the objective function, if the critical point does not exist. In this paper, we extend the original CLS formulation to a more general framework, where the proposed CLS method is capable of optimizing any objective function, derived from many popular DT criteria in speech recognition, such as MMI, MCE, MPE (or MWE), MD and so on. After approximating the KLD constraint as quadratic form, we can derive simple closed-form formula to efficiently update all parameters of Gaussian mixture CDHMMs based on the same idea of line search, including not only Gaussian mean vectors but also covariance matrices and mixture weights. The proposed CLS method has been evaluated in discriminative training of Gaussian mixture CDHMMs on several speech recognition tasks, including connected digit string recognition using the TIDIGITS database and large vocabulary continuous speech recognition on the Switchboard task. We have examined several different discriminative training critera in our experiments, including MMI, MPE and MD. Experimental results clearly show that the proposed CLS method consistently outperforms the popular EBW method in all evaluated ASR tasks in terms of final recognition performance and convergence behavior.

2. DISCRIMINATIVE TRAINING AS CONSTRAINED OPTIMIZATION PROBLEM

2.1. Criteria of discriminative training

We assume that acoustic model set $\mathbf{\Lambda}$ consists of many individual Gaussian mixture CDHMMs, each of which is represented as

$\boldsymbol{\lambda} = (\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$, where $\boldsymbol{\pi} = \{\pi_1, \pi_2, ..., \pi_N\}$ is the initial state distribution and N is the number of states in HMM, $\boldsymbol{A} = \{a_{ij}\}_{N\times N}$ is transition matrix, and $\boldsymbol{B}$ is state output distribution set, consisting of Gaussian mixture distributions for all states: $b_i(\boldsymbol{x}) = \sum_{k=1}^{K} \omega_{ik}\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_{ik}, \boldsymbol{\Sigma}_{ik})$, where K stands the number of Gaussian mixture components in state i ($1 \le i \le N$), and $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ represents a multivariate Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

Assume that the whole training set consists of R different training utterances $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, $\cdots$, $\boldsymbol{X}_R$ along with their corresponding transcriptions, denoted as W_1, W_2, $\cdots$, W_R. As shown in [6] and [11], objective functions of CDHMMs derived from various discriminative training criteria can be formulated in the following form:

$$\mathcal{F}(\boldsymbol{\Lambda}) = p(\boldsymbol{\Lambda} \mid \{\boldsymbol{X}_r, W_r, \mathcal{M}_r\}_{r=1}^{R}, f, \kappa, G) =$$

$$\frac{1}{R}\sum_{r=1}^{R} f\left(\log\left[\frac{\sum_{W\in\mathcal{M}_r} p^{\kappa}(\boldsymbol{X}_r|W)p(W)G(W, W_r)}{\sum_{W'\in\mathcal{M}_r} p^{\kappa}(\boldsymbol{X}_r|W')p(W')}\right]^{\frac{1}{\kappa}}\right) \quad (1)$$

where $0 < \kappa \le 1$ is acoustic scaling factor, and $\mathcal{M}_r$ stands for all competing hypotheses of utterance $\boldsymbol{X}_r$ which is compactly approximated by a word lattice generated in Viterbi decoding, $f(\cdot)$ is a mapping function to transform the objective function, and $G(W, W_r)$ is the so-called *gain function* to measure dissimilarity between reference W_r and a hypothesis W. Both the mapping function $f(\cdot)$ and the gain function $G(W, W_r)$ take different functional forms in various discriminative training criteria (see [6]). In this study, we assume that language model score $p(W)$ is fixed.

2.2. Constrained optimization for discriminative training

From eq.(1), we can see that the general DT objective function, $\mathcal{F}(\boldsymbol{\Lambda})$, is a highly complicated nonlinear function, which is difficult to optimize directly. Therefore, we normally make the following assumptions: i) all competing hypothesis spaces $\mathcal{M}_r$ remain unchanged during optimization; ii) all collected estimation statistics, such as state occupancies and Gaussian kernel occupancies, remain unchanged during optimization. Meanwhile, we also use a sufficiently small scaling factor κ ($\kappa \ll 1$) to smooth the original objective function. Because of these, it makes sense to explicitly impose a constraint that HMM model parameters $\boldsymbol{\Lambda}$ do not deviate too much from their initial values, $\boldsymbol{\Lambda}^0$. This constraint ensures that all of the above assumptions remain valid during optimization since the initial models, $\boldsymbol{\Lambda}^0$, have been used to generate all word lattices $\{\mathcal{M}_r\}$ and to accumulate statistics from training data prior to optimization.

Obviously, this kind of constraint can be quantitatively defined with the Kullback-Leibler divergence (KLD) between models. Therefore, given an initial model set $\boldsymbol{\Lambda}^0$, we propose to formulate discriminative training of CDHMMs as the following constrained maximization problem:

$$\boldsymbol{\Lambda}^* = \arg\max_{\boldsymbol{\Lambda}} \mathcal{F}(\boldsymbol{\Lambda}) \quad (2)$$

$$\text{subject to} \quad \mathcal{D}(\boldsymbol{\Lambda}||\boldsymbol{\Lambda}^0) \le \rho^2, \quad (3)$$

where $\mathcal{D}(\boldsymbol{\Lambda} \mid\mid \boldsymbol{\Lambda}^{\mathbf{0}})$ is the KLD between $\boldsymbol{\Lambda}$ and $\boldsymbol{\Lambda}^0$, and $\rho > 0$ is a pre-set constant to control the search range. The constraint in eq.(3) intuitively specifies a *trust region* for objective function optimization.

3. KLD CONSTRAINTS FOR CDHMMS

First of all, we consider to formulate the KLD-based model constraint in eq.(3) for different CDHMM parameters.

3.1. Constraint Decomposition for Gaussian Mixtures

Assume the whole model set $\boldsymbol{\Lambda}$ is composed of many physical states, the overall KLD constraint in eq.(3) can be relaxed into many local constraints for all individual state output distributions, e.g., $\mathcal{D}(b_i||b_i^0)$ ($1 \le i \le N$). Based upon the inequality of KLD between two mixture densities [12], we have:

$$\mathcal{D}(b_i||b_i^0) \le \mathcal{D}(\boldsymbol{\omega}_i||\boldsymbol{\omega}_i^0) + \sum_{k=1}^{K}\omega_{ik}\mathcal{D}(\mathcal{N}(\boldsymbol{\mu}_{ik}, \boldsymbol{\Sigma}_{ik})||\mathcal{N}(\boldsymbol{\mu}_{ik}^0, \boldsymbol{\Sigma}_{ik}^0)) \quad (4)$$

where $\boldsymbol{\omega}_i = (\omega_{i1}, \omega_{i2}, \cdots, \omega_{iK})'$ denotes all Gaussian mixture weights.

We can further break down the above constraint into separate independent constraints for Gaussian mean vectors, covariance matrices, and weights, respectively:

$$\begin{aligned}
&\mathcal{D}(\boldsymbol{\mu}_{ik} \mid\mid \boldsymbol{\mu}_{ik}^0) = (\boldsymbol{\mu}_{ik} - \boldsymbol{\mu}_{ik}^0)'(\boldsymbol{\Sigma}_{ik}^0)^{-1}(\boldsymbol{\mu}_{ik} - \boldsymbol{\mu}_{ik}^0) \le \rho^2 \\
&\mathcal{D}(\boldsymbol{\Sigma}_{ik} \mid\mid \boldsymbol{\Sigma}_{ik}^0) = \text{tr}\left[(\boldsymbol{\Sigma}_{ik}^0)^{-1}\boldsymbol{\Sigma}_{ik}\right] + \log|\boldsymbol{\Sigma}_{ik}^{-1}\boldsymbol{\Sigma}_{ik}^0| - D \le \rho^2 \\
&\mathcal{D}(\boldsymbol{\omega}_i \mid\mid \boldsymbol{\omega}_i^0) = \boldsymbol{\omega}_i' \cdot (\log\boldsymbol{\omega}_i - \log\boldsymbol{\omega}_i^0) \le \rho^2
\end{aligned} \quad (5)$$

where D is the dimension of feature space.

Obviously, the constraints of Gaussian mean vectors follow quadratic form which can be represented as:

$$(\boldsymbol{\mu}_{ik} - \boldsymbol{\mu}_{ik}^0)'(\boldsymbol{\Sigma}_{ik}^0)^{-1}(\boldsymbol{\mu}_{ik} - \boldsymbol{\mu}_{ik}^0) \equiv Q(\boldsymbol{\mu}_{ik} - \boldsymbol{\mu}_{ik}^0, \boldsymbol{\Sigma}_{ik}^0) \quad (6)$$

where $Q(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ stands for a standard quadratic form with a positive-definite matrix $\boldsymbol{\Sigma}$.

3.2. Quadratic Approximation for KLD Constraints

As we will show in section 4, for a quadratic form constraint, eq.(2) can be easily solved. Based on the assumption that model parameters stay in a close neighborhood of the original model, we can use Taylor series to approximate the constraints of covariance and weights into quadratic form as well.

In this study, we assume all Gaussian covariance matrices $\boldsymbol{\Sigma}_{ik}$ are diagonal: $\boldsymbol{\Sigma}_{ik} = \text{diag}(\sigma_{ik1}^2, \cdots, \sigma_{ikD}^2)$. For computational convenience, we represent each diagonal covariance matrix as a vector in the logarithm domain: $\boldsymbol{\sigma}_{ik} = (\log\sigma_{ik1}^2, \cdots, \log\sigma_{ikD}^2)'$. Then, we have:

$$\begin{aligned}
\mathcal{D}(\boldsymbol{\Sigma}_{ik}||\boldsymbol{\Sigma}_{ik}^0) &= \text{tr}[\boldsymbol{\Sigma}_{ik}(\boldsymbol{\Sigma}_{ik}^0)^{-1}] + \log|\boldsymbol{\Sigma}_{ik}^{-1}\boldsymbol{\Sigma}_{ik}^0| - D \\
&= \sum_{d=1}^{D}(e^{y_{ikd}} - y_{ikd} - 1) \approx \sum_{d=1}^{D} y_{ikd}^2/2 \\
&= \frac{1}{2}(\boldsymbol{\sigma}_{ik} - \boldsymbol{\sigma}_{ik}^0)'(\boldsymbol{\sigma}_{ik} - \boldsymbol{\sigma}_{ik}^0) \\
&\equiv Q(\boldsymbol{\sigma}_{ik} - \boldsymbol{\sigma}_{ik}^0, \boldsymbol{I})
\end{aligned} \quad (7)$$

where we denote $y_{ikd} = \log(\sigma_{ikd}/\sigma_{ikd}^0)^2$ and we have used the second-order Taylor series to approximate exponential function as $e^y - y - 1 \approx y^2/2$.

For Gaussian weights $\boldsymbol{\omega}_i$, we denote $z_{ik} = \omega_{ik}/\omega_{ik}^0$. Adopting the Taylor series approximation $\log z \approx z - 1$, we have:

$$\begin{aligned}
\mathcal{D}(\boldsymbol{\omega}_i||\tilde{\boldsymbol{\omega}}_i) &= \boldsymbol{\omega}_i' \cdot (\log\boldsymbol{\omega}_i - \log\hat{\boldsymbol{\omega}}_i) \\
&= \sum_{k=1}^{K}\omega_{ik}\log z_{ik} \approx \sum_{k=1}^{K}\omega_{ik}(z_{ik} - 1) \\
&= (\boldsymbol{\omega}_i - \boldsymbol{\omega}_i^0)'(\boldsymbol{\Pi}_i^0)^{-1}(\boldsymbol{\omega}_i - \boldsymbol{\omega}_i^0) \\
&\equiv Q(\boldsymbol{\omega}_i - \boldsymbol{\omega}_i^0, \boldsymbol{\Pi}_i^0)
\end{aligned} \quad (8)$$

where $\mathbf{\Pi}_i^0 = \text{diag}(\omega_{i1}^0, \cdots, \omega_{iK}^0)$ is a $K \times K$ diagonal positive-definite matrix. In additional to the above quadratic constraint, mixture weights, $\boldsymbol{\omega}_i$, must satisfy an affine constraint $\sum_{k=1}^K \omega_{ik} = 1$. Note that we have explicitly applied the constraints $\sum_{k=1}^K \omega_{ik} = \sum_{k=1}^K \omega_{ik}^0 = 1$ to derive the approximation in eq.(8).

In summary, we approximate the original KLD-based model constraints by the following positive-definite quadratic constraints for all of the model parameters:

$$\left\{ \begin{array}{lll} Q(\boldsymbol{\mu}_{ik} - \boldsymbol{\mu}_{ik}^0, \mathbf{\Sigma}_{ik}^0) \le \rho^2 & (1 \le i \le N) & (1 \le k \le K) \\ Q(\boldsymbol{\sigma}_{ik} - \boldsymbol{\sigma}_{ik}^0, \boldsymbol{I}) \le \rho^2 & (1 \le i \le N) & (1 \le k \le K) \\ Q(\boldsymbol{\omega}_i - \boldsymbol{\omega}_i^0, \mathbf{\Pi}_i^0) \le \rho^2 & (1 \le i \le N). & \end{array} \right. \tag{9}$$

4. CONSTRAINED LINE SEARCH

In this section, we consider how to solve the constrained optimization problem in eqs. (2) and (3) and derive our closed-form solution of Gaussian mixtures update in line search.

Firstly, we calculate partial derivatives of the general DT objective function $\mathcal{F}(\mathbf{\Lambda})$ with respect to any CDHMM parameter $\boldsymbol{\lambda}_{ik}$.

$$\nabla \mathcal{F}(\boldsymbol{\lambda}_{ik}) \equiv \frac{\partial}{\partial \boldsymbol{\lambda}_{ik}} \mathcal{F}(\mathbf{\Lambda}) = \frac{1}{R} \sum_{r=1}^R f_r' \cdot \frac{1}{\kappa} \cdot \sum_{W \in \mathcal{M}_r} \Bigg[\underbrace{\frac{p^\kappa(\boldsymbol{X}_r|W) \cdot p(W) G(W, W_r)}{\sum_{W' \in \mathcal{M}_r} p^\kappa(\boldsymbol{X}_r|W') p(W') G(W', W_r)}}_{G(W, W_r|\boldsymbol{X}_r)} - \underbrace{\frac{p^\kappa(\boldsymbol{X}_r|W) p(W)}{\sum_{W' \in \mathcal{M}_r} p^\kappa(\boldsymbol{X}_r|W') p(W')}}_{p(W|\boldsymbol{X}_r)} \Bigg] \frac{\partial \log p(\boldsymbol{X}_r|W)}{\partial \boldsymbol{\lambda}_{ik}} \tag{10}$$

When the smoothing factor κ is sufficiently small ($\kappa \to 0$) and the models do not deviate too much in each iteration, we can assume that all the three terms, i.e., f_r', $G(W, W_r|\boldsymbol{X}_r)$ and $p(W|\boldsymbol{X}_r)$, are approximately constants. Accordingly, we have:

$$\nabla \mathcal{F}(\boldsymbol{\lambda}_{ik}) = \frac{1}{R} \sum_{r=1}^R \sum_{W \in \mathcal{M}_r} C_r(W) \frac{\partial \log p(\boldsymbol{X}_r|W)}{\partial \boldsymbol{\lambda}_{ik}} \tag{11}$$

where we denote $C_r(W) = \frac{f_r'}{\kappa} \cdot [G(W, W_r|\boldsymbol{X}_r) - p(W|\boldsymbol{X}_r)]$ and $C_r(W)$ is approximately regarded to be independent of model parameters. Furthermore, we have

$$\frac{\partial \log p(\boldsymbol{X}_r|W)}{\partial \boldsymbol{\lambda}_{ik}} = \sum_{t=1}^T \gamma_{ik}^W(r,t) \cdot \frac{\partial \log \omega_{ik} \mathcal{N}(\boldsymbol{x}_{rt}; \boldsymbol{\mu}_{ik}, \mathbf{\Sigma}_{ik})}{\partial \boldsymbol{\lambda}_{ik}} \tag{12}$$

where $\gamma_{ik}^W(r,t)$ denotes posterior probabilities collected for k^{th} Gaussian component in i^{th} state of the composite HMM corresponding to W based on $\boldsymbol{X}_r$.

Next, if we substitute $\boldsymbol{\mu}_{ik}$, $\boldsymbol{\sigma}_{ik}$ or $\boldsymbol{\omega}_i$ in place of $\boldsymbol{\lambda}_{ik}$, we can derive partial derivatives of $\mathcal{F}(\mathbf{\Lambda})$ w.r.t. Gaussian means $\boldsymbol{\mu}_{ik}$, Gaussian variances $\boldsymbol{\sigma}_{ik}$ and Gaussian weight $\boldsymbol{\omega}_i$ as follows:

$$\begin{aligned} \nabla \mathcal{F}(\boldsymbol{\mu}_{ik}) &= \mathbf{\Sigma}_{ik}^{-1} \Big[\mathcal{O}_{ik}(\boldsymbol{x}) - \mathcal{O}_{ik}(1) \boldsymbol{\mu}_{ik} \Big] \\ \nabla \mathcal{F}(\boldsymbol{\sigma}_{ik}) &= \frac{\mathbf{\Sigma}_{ik}^{-1}}{2} \Big[\mathcal{O}_{ik}(\boldsymbol{x}^2) - \frac{\mathcal{O}_{ik}^2(\boldsymbol{x})}{\mathcal{O}_{ik}(1)} \Big] - \mathcal{O}_{ik}(1) \cdot \boldsymbol{\mu}_{ik} \\ \nabla \mathcal{F}(\boldsymbol{\omega}_i) &= \mathbf{\Pi}_i^{-1} \cdot \big(\mathcal{O}_{i1}(1), \cdots, \mathcal{O}_{ik}(1) \big)' \end{aligned} \tag{13}$$

where we denote $\mathcal{O}_{ik}(g(\boldsymbol{x})) = \sum_r \sum_W C_r(W) \sum_t \gamma_{ik}^W g(\boldsymbol{x})$.

Under the constraints in eq.(9), $\mathcal{F}(\mathbf{\Lambda})$ becomes a smooth function so that its unique critical point can be obtained by setting its derivative to zero, i.e., $\nabla \mathcal{F}(\mathbf{\Lambda}) = 0$. After solving the equations: $\nabla \mathcal{F}(\boldsymbol{\mu}_{ik}) = 0$, $\nabla \mathcal{F}(\boldsymbol{\sigma}_{ik}) = 0$, we can easily derive the critical point of the above smoothed objective for Gaussian mean and variances. For Gaussian weights, subject to the constraint of $\sum_k \omega_{ik} = 1$, we can use Lagrange multiplier to obtain the critical point. All critical points are obtained as follows:

$$\begin{aligned} \hat{\boldsymbol{\mu}}_{ik} &= \mathcal{O}_{ik}(\boldsymbol{x}) / \mathcal{O}_{ik}(1) \\ \hat{\boldsymbol{\sigma}}_{ik} &= \mathbf{log} \big[\mathcal{O}_{ik}(1) \cdot \mathcal{O}_{ik}(\boldsymbol{x}^2) - \mathcal{O}_{ik}^2(\boldsymbol{x}) \big] / \mathcal{O}_{ik}^2(1) \\ \hat{\boldsymbol{\omega}}_i &= \Big(\mathcal{O}_{i1}(1), \cdots, \mathcal{O}_{ik}(1) \Big)' / \sum\nolimits_{k=1}^K \mathcal{O}_{ik}(1) \end{aligned} \tag{14}$$

However, the above critical point, $\hat{\boldsymbol{\lambda}}$, may be a maximum, a minimum, or a saddle point of $\mathcal{F}(\mathbf{\Lambda})$. It may not exist in some special cases. We conceptually depict all possible situations in Figure 1. In total, we may have five cases: i) $\hat{\boldsymbol{\lambda}}$ is maximum and it is located inside the trust region , as shown in case 1; ii) $\hat{\boldsymbol{\lambda}}$ is maximum but outside the trust region, as in case 2; iii) $\hat{\boldsymbol{\lambda}}$ is a minimum, as in case 3; iv) $\hat{\boldsymbol{\lambda}}$ is a saddle point, as shown in case 4; v) no critical point exists, as shown in case 5. Among these cases, even when $\hat{\boldsymbol{\lambda}}$ is indeed a maximum, it may not be a good solution to eq.(2) since it may be too far from the initial point so that the constraint in eq.(9) is not satisfied, as in case 2.

Obviously, our ultimate goal is to optimize the objective function $\mathcal{F}(\mathbf{\Lambda})$ subject to the constraints given in eq.(9). In this work, we propose to use a line search method to solve the constrained optimization problem. Firstly, we determine a search direction for the line search. For cases 1, 2 and 3, it is reasonable to conduct line search along the line segment joining the initial point, $\boldsymbol{\lambda}^0$, and the calculated critical point, $\hat{\boldsymbol{\lambda}}$. However, for cases 4 and 5, it makes more sense to conduct line search along the gradient direction of the objective function calculated at the initial point, $\boldsymbol{\lambda}^0$. In summary, the search direction $\boldsymbol{d}$ for the line search is selected as follows:

$$\boldsymbol{d} = \left\{ \begin{array}{ll} \hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda}^0 & \hat{\boldsymbol{\lambda}} \text{ exists and is not a saddle point} \\ \nabla \mathcal{F}(\boldsymbol{\lambda}^0) & \text{otherwise.} \end{array} \right. \tag{15}$$

Secondly, the problem in eq.(2) can be formulated as the following constrained line search problem to optimize an interpolation weight ϵ along the pre-determined search direction $\boldsymbol{d}$ as:

$$\begin{aligned} \epsilon^* \quad = \quad & \arg\max_{\epsilon} \quad \mathcal{F}[\boldsymbol{\lambda}(\boldsymbol{\epsilon})] \\ & \text{subject to} \quad \mathcal{D}[\boldsymbol{\lambda}(\boldsymbol{\epsilon}) \;||\; \boldsymbol{\lambda}^0] \le \rho^2, \end{aligned} \tag{16}$$

where $\boldsymbol{\lambda}(\epsilon) = \boldsymbol{\lambda}^0 + \epsilon \cdot \boldsymbol{d}$ stands for model parameters linearly interpolated along the line specified in the direction of $\boldsymbol{d}$.

As long as we adopt the quadratic constraints in eq.(9), the above line search problem can be solved efficiently and the optimal interpolation weight ϵ^* can be computed in a closed-form for all five different cases in Figure 1 without any exhaustive search. For case 1, it is obvious that the optimal weight $\epsilon^* = 1$ since the computed critical point, $\hat{\boldsymbol{\lambda}}$, is the solution to eq.(16). For all other cases, it is clear that the optimal point is the intersection point of the search line with the quadratic constraint surface. In other words, the optimal interpolation weight ϵ^* satisfies $\mathcal{D}(\boldsymbol{\lambda}^0 + \epsilon^* \cdot \boldsymbol{d} || \boldsymbol{\lambda}^0) = \rho^2$. After substituting eq.(9) into it, we have

$$\epsilon^{*2} \cdot Q(\boldsymbol{d}, \boldsymbol{\phi}) = \rho^2. \tag{17}$$

Therefore, ϵ^* can be computed as $\epsilon^* = \pm\rho \cdot Q^{-\frac{1}{2}}(\boldsymbol{d}, \boldsymbol{\phi})$. Obviously, $\epsilon^* = -\rho \cdot Q^{-\frac{1}{2}}(\boldsymbol{d}, \boldsymbol{\phi})$ for case 3 while $\epsilon^* = \rho \cdot Q^{-\frac{1}{2}}(\boldsymbol{d}, \boldsymbol{\phi})$ for cases 2, 4 and 5. The results are summarized in Table 1. In each case, model parameter is updated as $\boldsymbol{\lambda}^* = \boldsymbol{\lambda}^0 + \epsilon^* \cdot \boldsymbol{d}$.

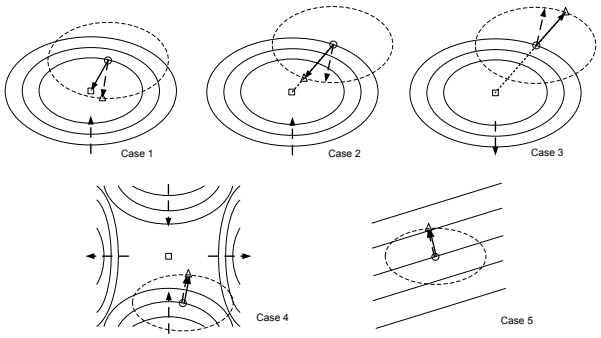

Fig. 1. Illustration of Constrained Line Search for maximizing the objective function in different cases.
($\bigcirc$: $\boldsymbol{\lambda}^0$, the initial point; $\square$: $\hat{\boldsymbol{\lambda}}$, the critical point; $\triangle$: $\boldsymbol{\lambda}^* = \boldsymbol{\lambda}(\epsilon^*)$, the optimal point; — : contours of $\mathcal{F}$; $\cdots$: the trust region; $\leftarrow$: search direction; $\leftarrow\!\!-\!\!-$: gradient direction)

Table 1. The CLS updating formula

case	condition	$\boldsymbol{d}$	ϵ^*
1	$\hat{\boldsymbol{\lambda}}$ is a maximum $Q(\boldsymbol{d}, \boldsymbol{\phi}) \leq \rho^2$	$\hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda}^0$	1
2	$\hat{\boldsymbol{\lambda}}$ is a maximum $Q(\boldsymbol{d}, \boldsymbol{\phi}) > \rho^2$		$+\rho \cdot Q^{-\frac{1}{2}}(\boldsymbol{d}, \boldsymbol{\phi})$
3	$\hat{\boldsymbol{\lambda}}$ is a minimum		$-\rho \cdot Q^{-\frac{1}{2}}(\boldsymbol{d}, \boldsymbol{\phi})$
4	$\hat{\boldsymbol{\lambda}}$ is a saddle point	$\nabla\mathcal{F}(\boldsymbol{\lambda}^0)$	$+\rho \cdot Q^{-\frac{1}{2}}(\boldsymbol{d}, \boldsymbol{\phi})$
5	$\hat{\boldsymbol{\lambda}}$ doesn't exist		

4.1. Updating Gaussian Means

For Gaussian mean vectors, the critical point, $\hat{\boldsymbol{\mu}}_{ik}$, can be easily calculated according to eq.(14). Now we need to examine conditions under which the computed critical point is a maximum, minimum or saddle point. From eq.(13), it is easy to show that $\nabla^2\mathcal{F}(\boldsymbol{\mu}_{ik}) = \mathcal{O}_{ik}(1) \cdot \boldsymbol{\Sigma}_{ik}^{-1}$. Since $\boldsymbol{\Sigma}_{ik}^{-1}$ is always a positive definite matrix, $\hat{\boldsymbol{\mu}}_{ik}$ can not be a saddle point. It is a maximum or minimum point depending on the sign of $\mathcal{O}_{ik}(1)$. If $\mathcal{O}_{ik}(1) > 0$, it is a maximum point; Otherwise it is a minimum point. If $\mathcal{O}_{ik}(1) = 0$, the objective function, $\mathcal{F}(\boldsymbol{\Lambda})$, degenerates into a linear function of $\boldsymbol{\mu}_{ik}$ and the critical point, $\hat{\boldsymbol{\mu}}_{ik}$, does not exist.

Furthermore, we can determine whether the critical point, $\hat{\boldsymbol{\mu}}_{ik}$, satisfies the constraint in eq.(9) by checking $Q(\hat{\boldsymbol{\mu}}_{ik} - \boldsymbol{\mu}_{ik}^0, \boldsymbol{\Sigma}_{ik}^0)$: If $Q(\hat{\boldsymbol{\mu}}_{ik} - \boldsymbol{\mu}_{ik}^0, \boldsymbol{\Sigma}_{ik}^0) < \rho^2$, $\hat{\boldsymbol{\mu}}_{ik}$ locates inside the trust region, as in case 1; Otherwise, it locates outside the trust region as in case 2.

In summary, if $\mathcal{O}_{ik}(1) > 0$ and $Q(\hat{\boldsymbol{\mu}}_{ik} - \boldsymbol{\mu}_{ik}^0, \boldsymbol{\Sigma}_{ik}^0) < \rho^2$, update as in case 1; If $\mathcal{O}_{ik}(1) > 0$ and $Q(\hat{\boldsymbol{\mu}}_{ik} - \boldsymbol{\mu}_{ik}^0, \boldsymbol{\Sigma}_{ik}^0) \geq \rho^2$, update as in case 2; If $\mathcal{O}_{ik}(1) < 0$, update as in case 3; If $\mathcal{O}_{ik}(1) = 0$, update as in case 5.

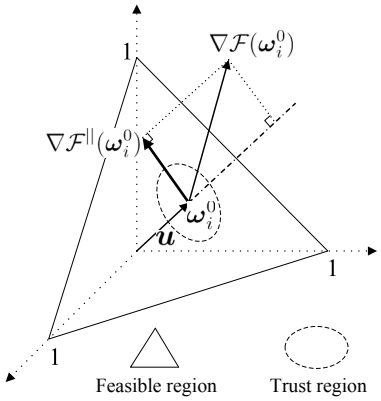

Fig. 2. Illustration of solving CLS problems for weight vectors by using the projected gradient

4.2. Updating Gaussian Variances

For Gaussian variances, the critical point, $\hat{\boldsymbol{\sigma}}_{ik}$, can be calculated according to eq.(14). We can see that $\hat{\boldsymbol{\sigma}}_{ik}$ exists only when the condition $\mathcal{O}_{ik}(1) \cdot \mathcal{O}_{ik}(\boldsymbol{x}^2) - \mathcal{O}_{ik}^2(\boldsymbol{x}) > 0$ holds. If $\mathcal{O}_{ik}(1) \cdot \mathcal{O}_{ik}(\boldsymbol{x}^2) - \mathcal{O}_{ik}^2(\boldsymbol{x}) < 0$, we have to conduct line search along gradient direction as in case 5 since the critical point, $\hat{\boldsymbol{\sigma}}_{ik}$, does not exist.

Furthermore, based on eq.(13), it is straightforward to show that $\nabla^2\mathcal{F}(\boldsymbol{\sigma}_{ik}) = -0.5 \cdot \mathcal{O}_{ik}(1) \cdot \boldsymbol{\Sigma}_{ik}^{-1} \cdot \exp(\hat{\boldsymbol{\sigma}}_{ik})$. If the critical point, $\hat{\boldsymbol{\sigma}}_{ik}$, exist, i.e., $\mathcal{O}_{ik}(1)\mathcal{O}_{ik}(\boldsymbol{x}^2) - \mathcal{O}_{ik}^2(\boldsymbol{x}) > 0$, we can easily derive that $\mathcal{O}_{ik}(1) > 0$. As the result, the second partial derivative $\nabla^2\mathcal{F}(\boldsymbol{\sigma}_{ik})$ is always negative-definite. Therefore, cases 3 and 4 never happen for Gaussian variances.

In summary, if $\mathcal{O}_{ik}(1)\mathcal{O}_{ik}(\boldsymbol{x}^2) > \mathcal{O}_{ik}^2(\boldsymbol{x})$ and $Q(\hat{\boldsymbol{\sigma}}_{ik} - \boldsymbol{\sigma}_{ik}^0, \boldsymbol{I}) < \rho^2$, update as in case 1; If $\mathcal{O}_{ik}(1)\mathcal{O}_{ik}(\boldsymbol{x}^2) > \mathcal{O}_{ik}^2(\boldsymbol{x})$ and $Q(\hat{\boldsymbol{\sigma}}_{ik} - \boldsymbol{\sigma}_{ik}^0, \boldsymbol{I}) \geq \rho^2$, update as in case 2; If $\mathcal{O}_{ik}(1)\mathcal{O}_{ik}(\boldsymbol{x}^2) \leq \mathcal{O}_{ik}^2(\boldsymbol{x})$, update as in case 5.

4.3. Updating Mixture Weights

For Gaussian weights $\boldsymbol{\omega}_i = (\omega_{i1}, \omega_{i2}, \cdots, \omega_{iK})'$, we can obtain the critical point $\hat{\boldsymbol{\omega}}_i$ shown in eq.(14), subject to the constraint of $\sum_{k=1}^K \omega_{ik} = 1$. Also, it is straightforward to verify that $\hat{\boldsymbol{\omega}}_i$ is a maximum when $\mathcal{O}_{ik}(1) > 0$ for all k, as in case 1 or 2; And $\hat{\boldsymbol{\omega}}_i$ is a minimum when $\mathcal{O}_{ik}(1) < 0$ for all k, as in case 3. Otherwise, $\hat{\boldsymbol{\omega}}_i$ is neither maximum nor minimum. In this case, we follow the gradient to update $\boldsymbol{\omega}_i$. To ensure that the weights remain a valid discrete probability distribution, we need project the gradient, i.e. $\nabla\mathcal{F}(\boldsymbol{\omega}_i^0)$, onto the hyperplane $\sum_{k=1}^K \omega_{ik} = 1$, as shown in Figure 2:

$$\nabla\mathcal{F}^{||}(\boldsymbol{\omega}_i^0) = \nabla\mathcal{F}(\boldsymbol{\omega}_i^0) - \Big[\nabla\mathcal{F}(\boldsymbol{\omega}_i^0) \cdot \boldsymbol{u}\Big]\boldsymbol{u} \tag{18}$$

where $\boldsymbol{u} = (\frac{1}{\sqrt{K}}, \cdots, \frac{1}{\sqrt{K}})'$ is the normal vector of the hyperplane.

In summary, if $\mathcal{O}_{ik}(1) > 0$ and $Q(\hat{\boldsymbol{\omega}}_i - \boldsymbol{\omega}_i^0, \boldsymbol{\Pi}_i^0) < \rho^2$, update as in case 1; If $\mathcal{O}_{ik}(1) > 0$ and $Q(\hat{\boldsymbol{\omega}}_i - \boldsymbol{\omega}_i^0, \boldsymbol{\Pi}_i^0) \geq \rho^2$, update as in case 2; If $\mathcal{O}_{ik}(1) < 0$, update as in case 3; Otherwise, update as in case 5 along the projected gradient in eq.(18). In practice, we also need check the boundary condition of $0 < \omega_{ik} < 1 (1 \leq k \leq K)$ to ensure a valid discrete probability distribution.

5. EXPERIMENTS

In order to verify the effectiveness of the proposed CLS optimization method, we have evaluated it on several benchmark speech recognition tasks, including: connected digit string recognition using the TIDIGITS database, large vocabulary continuous speech recognition using the Switchboard database. In the experiments, the CLS method is compared with the popular EBW method for optimizing the MMI and other DT criteria, such as MPE and MD. In our EBW implementation, following [8], we use kernel dependent smoothing factors which are set to be twice of the corresponding denominator occupancy. When we use EBW for the MPE training, we also use I-smoothing [8] with factor τ set to be 100 during each iteration. In our experiments, the CLS algorithm is operated iteratively. In each iteration, the known models are set as the initial model set, $\mathbf{\Lambda}^0$, in the constraint in eq.(9) and then model parameters are updated according to the CLS formula in section 4. The constant ρ^2 is set to $0.1/n$ in the n^{th} iteration.

When we evaluate the EBW algorithm, we update all the model parameters (including mean, variance and weights). When we test the CLS algorithm, we compare performance of updating Gaussian means only with that of updating all the model parameters altogether.

5.1. TIDIGITS

The TIDIGITS database contains utterances from a total of 326 speakers (111 men, 114 women and 101 children). In our experiments, we have used all data from adults and children, which includes 12,549 training utterances and 12,547 testing utterances. The acoustic features used are 39-dimension MFCCs. The vocabulary is composed of 11 digits of 'zero' to 'nine', plus 'oh'. The length of digit strings varies from 1 to 7 digits. Each digit is modeled by a 10-state, left-to-right, whole-word Gaussian mixture CDHMMs. The best ML-trained model consists of 114 tied states with 6 Gaussians per state. In the experiment, the best ML model is used as the seed model for discriminative training, in either EBW or CLS method.

In Fig. 3, we compare learning curves of CLS and EBW methods in the MMI training. The results clearly show that the proposed CLS method yields much better performance than the EBW method. Firstly, the CLS algorithm shows faster convergence speed than the conventional EBW method. Secondly, the CLS method achieves much lower recognition error rate. For CLS, word error rate decreases from 1.16% to 0.42%, which represents about 63.8% relative error reduction. On the other hand, the EBW method achieves only 44% relative error reduction.

From Fig. 3, we observe that the benefit of updating variances and weights is marginal as long as the means are updated properly.

In addtion, we also compare CLS with EBW in optimizing the MD criterion, which define errors with higher resolution. Here we didn't use MPE because it cannot be directly applied to whole word models. The results are shown in Table 2, from which we can see that the CLS method still outperforms the EBW method. In the MD training, the CLS method achieves 0.4% word error rate which is slightly better than 0.44% in the EBW method.

5.2. Switchboard

In the Switchboard task, we have used two different training sets: the *mini-train* and the full *h5train00* set, consisting of 18 and 265 hours of speech data, respectively. The acoustic features used are 39-dimension PLPs. *Eval2000* set, which contains 1,831 utterances, was used as the evaluation set. Context dependent tri-phone HMMs

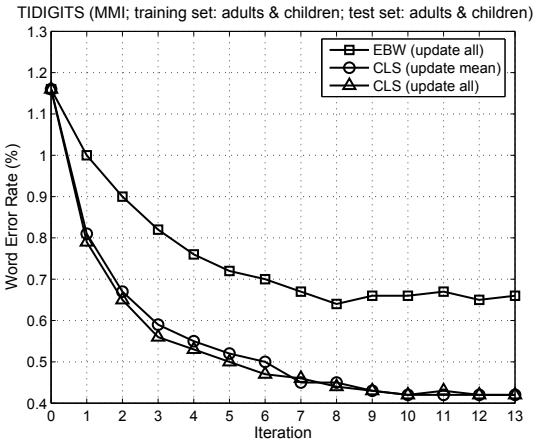

Fig. 3. WER Comparison of different optimization methods in MMI training on the TIDIGITS task.

Table 2. Summary of recognition performance in TIDIGITS by using EBW or CLS optimization method for MMI and MD criteria.

Criterion	Optimization	WER (in %)
ML	BW	1.16
MMI	EBW	0.65
	CLS	0.42
MD	EBW	0.44
	CLS	0.40

are used in this experiment. Tri-gram language model was used in testing, and uni-gram language model was used in training. The NIST scoring software was used to evaluate word error rates.

On mini-train task, the baseline ML models consists of 1500 physical states with 12 Gaussian kernels per state, while on h5train00 task, the baseline ML models consists of 6000 physical states with 16 Gaussian kernels per state.

We first compare the CLS method with the EBW method in MMI training, as shown in 4 and 5. The results show again that the proposed method achieves better word accuracy and more stable convergence than EBW method on both *mini-train* and *h5train00* training set. Compared with ML baselines, the word error rate decreases from 40.8% to 37.9%, or a 7.1% relative error reduction for the *mini-train* set, and from 31.7% to 28.9%, or a 8.8% relative error reduction for the *h5train00 set*, respectively, from the best MLE-trained models.

It is remarkable that on the Switchboard task, the benefit of updating variances and weights is significant. By using the proposed CLS algorithm, we can effectively adjust all the model parameters in the sense of discriminative training.

At last, we also compare CLS with EBW in optimizing the MPE criterion on the full set of switchboard task. The results are summarized in Table 3. Again, the CLS is demonstrated to be advantageous over the EBW in optimizing MPE criterion as well.

6. CONCLUSIONS

In this paper, *constrained line search (CLS)*, has been proposed as discriminative training algorithm in speech recognition. The pro-

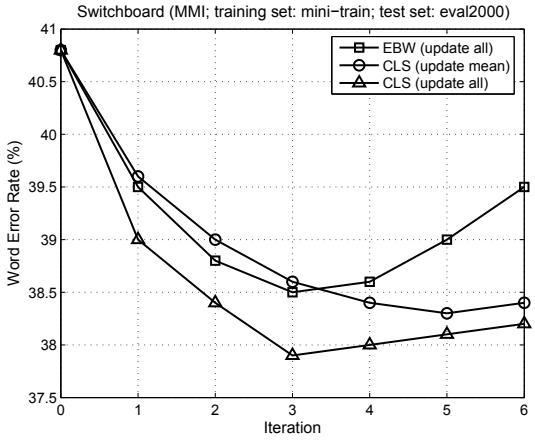

Fig. 4. Comparison of word error rates of different optimization methods on the Switchboard *eval2000* test set, using *mini-train* training set, based on MMI criterion.

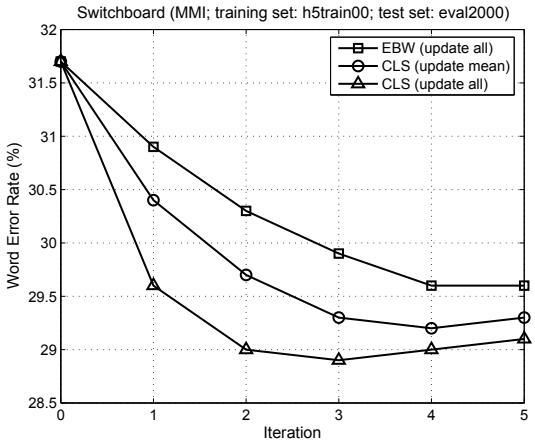

Fig. 5. WER Comparison of different optimization methods in MMI training on the Switchboard h5train00 task.

posed CLS method is general enough to optimize various popular objective functions in discriminative training. In this work, discriminative training of CDHMMs is first formulated as a constrained optimization problem, where a constraint is imposed on the KLD between models, which guarantees an equalized updating process across all the parameters in the model set. Based upon some approximations on the KLD constraint, closed-form solutions can be easily derived for updating all CDHMM parameters. We examined the proposed CLS methods on several standard speech recognition tasks, from small vocabulary digit string recognition to large vocabulary continuous speech recognition. Experimental results clearly show that our method can effectively update all model parameters of Gaussian mixture CDHMM, and it consistently outperforms the popular EBW method.

Table 3. Summary of recognition performance (WER in %) in Switchboard by using EBW or CLS optimization method for MMI and MPE training criteria.

Criterion	Optimization	mini-train	full *h5train00*
ML	BW	40.8	31.7
MMI	EBW	38.5	29.6
	CLS	37.9	28.9
MPE	EBW w/ I-smooting	38.0	28.7
	CLS	37.7	28.4

7. REFERENCES

[1] P. C. Woodland and D. Povey, "Large Scale Discriminative Training of hidden Markov models for speech recognition," *Computer Speech & Language*, pp.25-47, Vol. 16, No. 1, January 2002.

[2] B.-H. Juang, W. Chou and C.-H. Lee, "Minimum Classification Error Rate Methods for Speech Recognition," *IEEE Trans. SAP*, pp.257-265, Vol.5, No.3, May 1997.

[3] H. Jiang, F. Soong and C.-H. Lee, "A dynamic in-search data selection method with its applications to acoustic modeling and utterance verification," *IEEE Trans. SAP,* pp.945-955, Vol. 13, No.5, September 2005.

[4] B. Liu, H. Jiang, J.-L. Zhou, R.-H. Wang, "Discriminative Training Based on The Criterion of Least Phone Competing Tokens for Large vocabulary Speech Recognition," *Proc. of ICASSP'05,* Philadelphia, Mar. 2005.

[5] D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," *Proc. of ICASSP'02,* Orlando, 2002.

[6] J. Du, P. Liu, F.K. Soong, J.-L. Zhou, R.-H Wang, "Minimum Divergence based Discriminative Training," *Proc of ICSLP'06*, pp.2410-2413, 2006.

[7] C. Liu, P. Liu, H. Jiang, F. Soong and R.-H. Wang, "A Constrained Line Search Optimization For Discriminative Training in Speech Recognition," *Proc. of ICASSP'2007*, Hawaii, USA, April 2007.

[8] D. Povey, "Discriminative Training for Large Vocabulary Speech Recognition," *Ph.D. thesis*, Cambridge University, 2004.

[9] X. He and W. Chou, "Minimum Classification Error linear regression for acoustic model adaptation of continuous density HMMs," *Proc. of ICASSP'2003*, pp.556-559, 2003.

[10] P. S. Gopalakrishnan, D. Kanevsky, A. Nadas, D. Nahamoo, "Generalization of the Baum Algorithm to Rational Objective Functions," pp. 631-634, *Proc. ICASSP'89.*

[11] R. Schluter, *Investigations on Discriminative Training Criteria*, Ph.D.thesis, Aachen University, 2000.

[12] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, New York: Wiley Interscience, 1991.