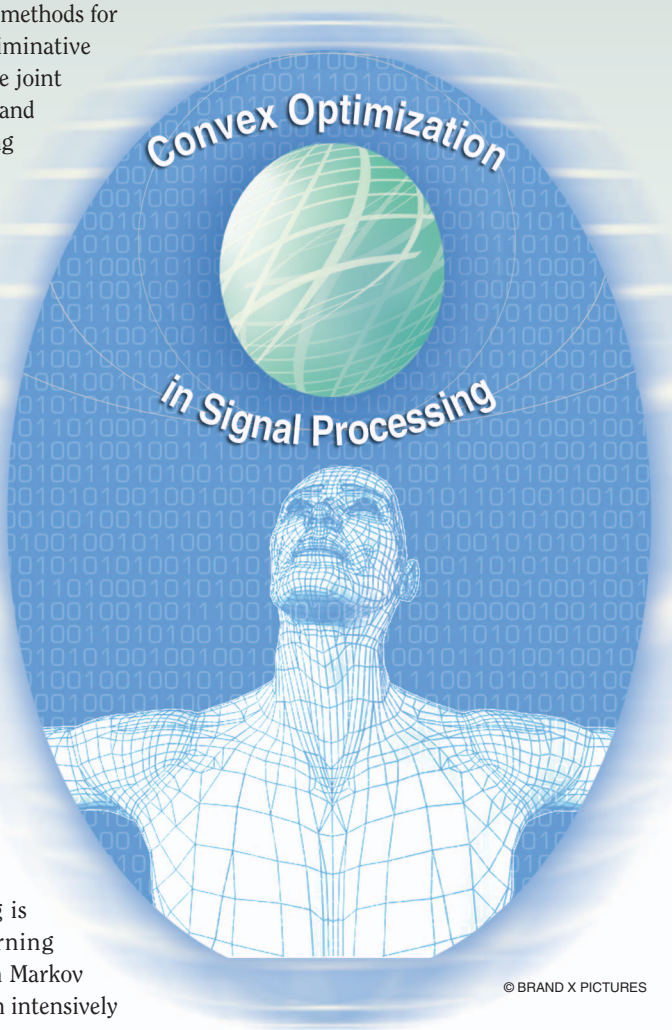
Hui Jiang and Xinwei Li

Parameter Estimation of Statistical Models Using Convex Optimization

An advanced method of discriminative training for speech and language processing

In machine learning, there are two distinct groups of learning methods for building pattern classifiers: generative learning and discriminative learning. The generative learning scheme aims to estimate the joint probability distribution of observation data and class label and then use the estimated distribution for classification according to the well-known Bayes decision rule. In generative learning, it is normal practice to adopt the so-called parametric modeling approach, where it is assumed that unknown probability distributions belong to computationally tractable function families. In this way, the difficult density estimation problem becomes a more tractable parameter estimation problem. The advantage of generative learning is that some inherent dependency structures and/or independency assumptions applicable to the underlying data can be explicitly exploited by choosing an appropriate statistical model, such as mixture models or even highly structured graphical models. On the other hand, the discriminative learning scheme makes no explicit attempt to model the data distribution and instead optimizes a mapping function from inputs to any desired outputs. As a result, only the decision boundary is adjusted without forming a data generator in the entire feature space. The advantage of discriminative learning lies in that the mapping function can be estimated based on criteria that are more relevant to the ultimate classification and regression task. Because of their complementary nature, recent research work in machine learning [1], [9], [10], [12], [32] has shown the potential benefit of combining generative and discriminative learning methods.

One active research topic in speech and language processing is how to learn generative models using discriminative learning approaches. For example, discriminative training (DT) of hidden Markov models (HMMs) for automatic speech recognition (ASR) has been intensively



© BRAND X PICTURES

Digital Object Identifier 10.1109/MSP.2010.936018

studied for several decades. The essential idea of DT is to apply a discriminative criterion to the training procedure of HMMs. Many discriminative criteria have been proposed for ASR, such as maximum mutual information estimation (MMIE) [2], [35], minimum classification error (MCE) [18], minimum phone (word) error (MPE/MWE) [29], and large margin estimation (LME) [13], [20], [31], [40]. In a standard DT process, an objective function is first constructed according to one of these discriminative criteria. Then, optimization methods are used to maximize/minimize the objective function with respect to the model parameters. The major difficulty of DT in ASR lies in the fact that normally the above-mentioned DT criteria result in large-scale nonconvex optimization problems. For example, in a state-of-the-art large vocabulary ASR system, these DT criteria normally lead to complex nonconvex objective functions involving over millions or even tens of millions of free variables. Generally speaking, optimizing nonconvex objective functions of such a large number of variables is extremely difficult since it is very easy to get trapped in a shallow local optimum point in the complicated surface of the objective functions. During the last two decades, a significant amount of research effort in the speech community has been devoted to this problem. Many different optimization methods have been proposed and applied to DT of HMMs in ASR. For example, the extended Baum-Welch (EBW) method was proposed based on a growth function [7], which was first applied to MMIE [35] and then extended to other discriminative criteria including MCE and MPE/MWE [8]. Moreover, a stochastic approximation method called generalized probabilistic descent (GPD) [18] was first proposed for MCE, and then an approximate second-order Quasi-Newton method called quickprop was applied to MCE and other criteria [26]. More recently, constrained line search [21] and trust region [37], [22] methods have also been proposed for DT of HMMs in ASR. Generally speaking, all of these optimization methods are nonconvex in nature and normally lead to better recognition performance only when carefully and skillfully implemented, such as in [35]. These nonconvex optimization methods, especially the EBW approach, remain popular optimization methods for DT of HMMs in ASR. Interested readers may refer to several recent survey articles [8], [17] for details. In this tutorial article, as part of applications of convex optimization in signal processing, we focus on more recent research work that applies convex optimization methods to DT for speech and language processing applications.

ONE ACTIVE RESEARCH TOPIC IN SPEECH AND LANGUAGE PROCESSING IS TO STUDY HOW TO LEARN GENERATIVE MODELS USING DISCRIMINATIVE LEARNING APPROACHES.

INTRODUCTION

To the best of our knowledge, there have been at least two independent research efforts to apply convex optimization methods to the discriminative learning of HMMs for ASR. The advantage of convex optimization is that it does not suffer from the local optimum problem because any local optimum is always globally optimal in convex optimization problems. Both research groups have been motivated by advances of large-margin classifiers in machine learning and attempt to estimate Gaussian mixture HMMs for ASR based on large margin criterion. In [30] and [31], a large margin-based DT method was proposed to estimate Gaussian mixture models for multiclass pattern classification tasks and then extended to Gaussian mixture HMMs for sequential classification such as ASR. In this work, LME of Gaussians is formulated as a regularized optimization problem where the regularization term is calculated based on traces of parameter matrices and a so-called reparamerization method is proposed to relax multivariate Gaussian parameters to formulate LME as a convex optimization problem, which is solved by a customized gradient descent method for efficiency. In [13] and [15], a different strategy is applied to LME of Gaussian mixture HMMs for ASR, where LME of HMMs is directly formulated as a minimax optimization problem based on the principle of maximizing the minimum margin of the HMM-based classifiers. Then, an iterative optimization method, called approximation optimization (AM), is used to solve the original minimax optimization problem through a locality approximation. Convex relaxation methods may be used to convert the original nonconvex optimization into standard convex optimization problems, such as semidefinite programming (SDP) [19], [39] and second-order cone programing (SOCP) [36], [38]. Even though this work was initially proposed only for LME of mean parameters of Gaussian mixture HMMs, it can be easily extended to estimate other parameters of HMMs, such as covariance matrices and transition probabilities, as well as other discriminative criteria, such as MMIE and MCE. As shown in [16], the proposed AM method is general enough for DT of generative models from a wide class of statistical models, namely finite mixtures of exponential family models, while the method in [30], [31] is specially tailored for Gaussians. Over the past few years, we have successfully applied a variety of convex optimization methods under the AM framework to solve some DT problems arising in speech and language processing tasks, initially starting with Gaussian mixture HMMs for speech recognition [19], [36], [38], [39] and more recently extending to multinomial-based models for language processing [25], [28]. Here, convex optimization plays a critical role in solving the underlying large-scale optimization problems in DT of the generative models. In this tutorial article, as an example of convex optimization for signal processing, we summarize recent research efforts that apply convex optimization to DT of various statistical models under the AM framework and highlight the emerging role of convex optimization in these traditional speech and language applications.

GENERATIVE VERSUS DISCRIMINATIVE LEARNING

Pattern classification is the task of classifying an unknown object into one of a set of priori pattern classes based on some noisy observation data (also known as features). The generative learning scheme originates from the Bayes decision rule (also known as maximum a posterior decision rule), which is guaranteed to achieve the minimum classification error rate provided the underlying data distributions are known. For an unknown object, we assume its class label, S, and its observation data, X, are both random variables, if the true joint probability distribution of X and S is known as $p(X, S) = p(S) \cdot p(X|S)$, the optimal pattern classifier can be built based on the MAP (maximum a posterior) decision rule. In other words, class label of an unknown observation X is predicted as follows:

$$\hat{S} = \arg\max_{S} p(S \mid X) = \arg\max_{S} p(S) \cdot p(X|S). \qquad (1)$$

As indicated in (1), constructing the optimal pattern classifier requires to compute two probability distributions, namely the prior probability $p(S)$ and the conditional probability $p(X|S)$, which are usually unknown in practice. The essential problem in generative learning is how to estimate them from some available training data. To simplify the estimation problem, we adopt the so-called parametric modeling approach, where it is assumed that the unknown probability distributions belong to a computationally tractable function family, such as the exponential family [4]. Furthermore, some latent variables may be introduced to derive even more complex models. A rich family of multimodal distributions can be obtained by introducing discrete latent variables. In this way, the difficult density estimation problem turns into a more tractable parameter estimation problem. One major benefit of generative learning is that there exist efficient learning algorithms that can estimate rather complicated models based on the maximum likelihood (ML) criterion, e.g., the expectation-maximization (EM) algorithm [6].

Discriminative learning uses a mapping function (from observation X to class label S) to model class boundaries without forming data distribution in the entire feature space. The mapping function is estimated using criteria that are more relevant to the ultimate classification and regression task, such as maximum condition likelihood (MCL) estimation [11], empirical risk minimization (ERM), or LME [41]. Traditionally, only some relatively simple models are considered in the discriminative learning scheme due to computational complexity issues, e.g., linear discriminant functions for support vector machines (SVMs). Other discriminative models include logistic regression and neural networks. It is an interesting topic to extend the discriminative learning scheme to other more complicated models, particularly mixture models and latent graphical models widely adopted in the generative learning scheme. Recent work in both machine learning [1], [9], [10], [12], [32] and speech recognition [2], [18], [35] has shown the benefit of learning generative models discriminatively. Discriminative learning of these generative models imposes a computational challenge since it normally leads to a complicated nonconvex optimization problem. A significant amount of research effort has been devoted to this problem. In this article, we introduce one method proposed for discriminative learning of a wide class of generative models, i.e., mixtures of exponential family (e-family), and particularly underline the role of convex optimization in this method by using convex relaxation techniques.

THE ADVANTAGE OF CONVEX OPTIMIZATION IS THAT IT DOES NOT SUFFER FROM THE LOCAL OPTIMUM PROBLEM BECAUSE ANY LOCAL OPTIMUM IS ALWAYS GLOBALLY OPTIMAL IN CONVEX OPTIMIZATION PROBLEMS.

In the next section, we first introduce notations for a general canonical form of the e-family and its mixtures since many popular generative models fall into this category. Following that, we will present a general framework to learn mixtures of exponential family models in a discriminative way.

STATISTICAL MODELS: THE E-FAMILY AND ITS MIXTURE

In practice, we normally choose generative models from a rather general family of statistical models, namely the e-family, due to its highly computationally tractability in parameter estimation.

THE E-FAMILY

As in [4], all statistical models in the e-family can be represented as the following general canonic form:

$$p(X|\lambda) = \exp\{A(\mathrm{x}) + \mathrm{x}^T\boldsymbol{\lambda} - \mathcal{K}(\boldsymbol{\lambda})\},$$

where $\boldsymbol{\lambda}$ in bold denotes the natural parameter vector of the e-family distribution, and x in bold is called the sufficient statistics. In most cases, the natural parameters $\boldsymbol{\lambda}$ may take a different form from the original model parameter λ and sufficient statistics x can be represented as a function of original data X. Furthermore, $\mathcal{K}(\boldsymbol{\lambda})$ is called cumulant generating function, which is a convex function of the natural parameters $\boldsymbol{\lambda}$ and independent of X. $A(\mathrm{x})$ is a function of sufficient statistics x and independent of λ.

Many common probability distributions belong to the e-family, including Gaussian, multinomial, Bernoulli, exponential, gamma, and poison. For example, one-dimensional Gaussian (with unknown mean and variance) can be represented in the canonic e-family form as

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} = \exp\{A(\mathrm{x}) + \mathrm{x}^T\boldsymbol{\lambda} - \mathcal{K}(\boldsymbol{\lambda})\},$$

where sufficient statistics $\mathrm{x} = [x, -x^2/2]$, the natural parameters $\boldsymbol{\lambda} = [\mu/\sigma^2, 1/\sigma^2]$ and $A(\mathrm{x}) = -(1/2)\ln 2\pi$, and the

cumulant generating function $\mathcal{K}(\boldsymbol{\lambda}) = -(1/2)\ln \boldsymbol{\lambda}_2 + \boldsymbol{\lambda}_1^2/2\boldsymbol{\lambda}_2$ (with the constraint $\boldsymbol{\lambda}_2 > 0$), which is clearly a convex function of $\boldsymbol{\lambda}$. We can represent all other common probability distributions in a similar way. In Table 1, we have summarized the major results for multivariant Gaussian and multinomial distributions since they are mostly relevant to the remainder of this article.

ONE IMPORTANT PROPERTY OF THE E-FAMILY IS THAT PRODUCTS OF E-FAMILY DISTRIBUTIONS REMAIN IN THE E-FAMILY.

One important property of the e-family is that products of e-family distributions remain in the e-family. The natural parameters of the product distribution can be easily constructed by concatenating natural parameters of all member distributions.

MIXTURES OF THE E-FAMILY

All e-family distributions are computationally tractable in parameter estimation but they are normally too simple to model real-word data appropriately. A widely adopted strategy is to introduce latent variables to derive more complicated models from these simple e-family distributions. A rather rich family of multimodal distributions can be obtained by introducing discrete latent variables to derive the so-called finite mixture models.

Mixtures of the e-family are a generalization of the above simple e-family distribution by considering a convex combination of different e-family distributions with different parameters. In a very general form, mixtures of the e-family can be represented as the following form:

$$p(X|\lambda) = \sum_k w_k \cdot \exp\{A_k(\mathbf{x}_k) + \boldsymbol{\lambda}^T \mathbf{x}_k - \mathcal{K}_k(\boldsymbol{\lambda})\},$$

where $\boldsymbol{\lambda}$ denotes the natural parameters of the mixture model that is concatenated from natural parameters of its component distributions, and sufficient statistics $\mathbf{x}_k$ and $\mathcal{K}_k(\boldsymbol{\lambda})$ may vary in different mixture components, and w_k stands for mixture weight and satisfies the sum-to-one constraint $\sum_k w_k = 1$.

Mixtures of the e-family represent a wide class of statistical models and are frequently used in machine learning and pattern classification. Mixtures of the e-family include the Gaussian mixture model (GMM), the multinomial mixture model (MMM), and the HMM. In the HMM, the latent variable is a complete state sequence, $\mathbf{x}_k$ represents sufficient statistics collected along a state sequence, and each mixture component is product of various e-family distributions.

In the generative learning scheme, statistical models are normally estimated from available training data based on the maximum likelihood (ML) criterion. Due to the convexity of the cumulant generating function $\mathcal{K}(\boldsymbol{\lambda})$, ML estimation of an e-family model normally result in a simple solution. In many cases, ML estimation of e-family models can even be solved by a closed-form solution. Furthermore, ML estimation of mixtures of the e-family can also be iteratively derived in a relatively easy manner, relying on the well-known EM algorithm [6].

A NEW FRAMEWORK FOR DT OF GENERATIVE MODELS

In this section, we consider the use of alternative discriminative learning approaches to estimate mixtures of e-family models. Unlike the conventional ML estimation, discriminative learning of these generative models faces significant computational challenges, no longer enjoying simple updating rules and relatively fast convergence of the EM algorithm. Generally speaking, discriminative learning of generative models is an optimization problem, where an objective function must be optimized in an iterative manner. The discriminative objective function is normally formulated according to some popular discriminative criteria, such as maximum mutual information

[TABLE 1] CANONIC FORM OF E-FAMILY DISTRIBUTIONS: A) MULTIVARIATE GAUSSIAN WITH KNOWN PRECISION MATRIX; B) MULTIVARIATE GAUSSIAN WITH UNKNOWN MEAN AND PRECISION MATRIX; C) MULTINOMIAL IN A CONSTRAINED FORM; D) MULTINOMIAL IN AN UNCONSTRAINED FORM.

		$\boldsymbol{\lambda}$	$\mathbf{x}$	$\mathcal{K}(\boldsymbol{\lambda})$	CONSTRAINT
A)	GAUSSIAN $\mathcal{N}(\mu, \Sigma_0)$ (MEAN)	μ	$\Sigma_0^{-1}\mathrm{x}$	$\frac{1}{2}\boldsymbol{\lambda}^T\Sigma_0^{-1}\boldsymbol{\lambda}$	
B)	GAUSSIAN $\mathcal{N}(\mu, \Sigma)$ (MEAN,COV)	$[\Sigma^{-1}\mu, \Sigma^{-1}]$	$\left[x, -\frac{1}{2}x \cdot x^T\right]$	$-\frac{1}{2}\ln\lvert\boldsymbol{\lambda}_2\rvert + \frac{1}{2}\boldsymbol{\lambda}_1^T\boldsymbol{\lambda}_2^{-1}\boldsymbol{\lambda}_1$	$\boldsymbol{\lambda}_2 \succ 0$
C)	MULTINOMIAL (CONSTRAINED) $C \cdot \prod_{d=1}^{D} \mu_d^{x_d}$	$[\ln\mu_1, \ldots, \ln\mu_D]$	$[x_1, \ldots, x_D]$	1	$\boldsymbol{\lambda} < 0$ $\sum_{d=1}^{D} e^{\boldsymbol{\lambda}_d} = 1$
D)	MULTINOMIAL (UNCONSTRAINED) $C \cdot \prod_{d=1}^{D} \mu_d^{x_d}$	$\left[\ln\frac{\mu_1}{1-\sum_{d=1}^{D-1}\mu_d}, \ldots, \ln\frac{\mu_{D-1}}{1-\sum_{d=1}^{D-1}\mu_d}\right]$	$\left[\frac{x_1}{\sum_{d=1}^{D}x_d}, \ldots, \frac{x_{D-1}}{\sum_{d=1}^{D}x_d}\right]$	$\ln\left(1+\sum_{d=1}^{D-1}e^{\boldsymbol{\lambda}_d}\right)$	

(MMI) [2], [35] (maximum conditional likelihood [11]), MCE [18], and LME [13], [31]. Historically, these discriminative criteria have been proposed from different contexts, in the following, we first present a unified view for discriminative criteria, centering on the concept of margin. Following that, we discuss a general method to optimize these discriminative objective functions using locality approximation and convex optimization, named as the AM method.

DISCRIMINATIVE LEARNING OF GENERATIVE MODELS IS A TYPICAL OPTIMIZATION PROBLEM, WHERE EFFICIENT OPTIMIZATION METHODS PLAY A CRITICAL ROLE.

A UNIFIED VIEW OF VARIOUS DT CRITERIA

In supervised learning, given a set of training samples, denoted as $\mathcal{D} = \{X_1, X_2, \ldots, X_T\}$, we usually know the true class labels for all training samples in $\mathcal{D}$, denoted as $\mathcal{L} = \{S_1, S_2, \ldots, S_T\}$. For notational convenience, we use the uppercase letter S_t to represent the true transcription of each training sample X_t, and use lowercase s_t to denote a variable that may take all possible labels in a hypothesis space.

Following [5] and [34], a multiclass separation margin for each training sample X_t is defined as follows:

$$d(X_t \mid \Lambda) = \ln p(S_t, X_t|\Lambda) - \max_{s_t \in \mathcal{M}_t} \ln p(s_t, X_t|\Lambda), \qquad (2)$$

where Λ denotes the whole set of model parameters from all classes and max is taken over a particular hypothesis space, denoted as $\mathcal{M}_t$. In many static pattern classification tasks, such as text categorization, $\mathcal{M}_t$ represents a set of all possible class labels including or excluding the correct label S_t. However, in sequential pattern classification tasks, such as continuous speech recognition, the size of $\mathcal{M}_t$ is prohibitively large, such as all possible word sequences consisting of N or fewer words. In this case, $\mathcal{M}_t$ may be represented by either an N-best list or a word graph (or word lattice). An N-best list is simply a linear list of the top N-best label sequences s_t, as measured by probability score $p(s_t, X_t|\Lambda)$. Word graph (lattice) is another more compact way to represent most likely label sequences as a graph, where each complete path in word graph constitutes a competing hypothesis for the given sentence. N-best lists and word graphs can be efficiently generated as a by product from the Viterbi decoding process (e.g., see [27] for details).

For computational convenience, a variant margin can be defined based on soft-max using log-sum as follows:

$$d'(X_t \mid \Lambda) = \ln p(S_t, X_t|\Lambda) - \ln \left[\sum_{s_t \in \mathcal{M}_t} p(s_t, X_t|\Lambda) \right]. \qquad (3)$$

Obviously, according to the maximum a posteriori (MAP) decision rule in (1), $d(X_t \mid \Lambda) > 0$ if and only if X_t is correctly recognized by the model set Λ. Moreover, margin $d(X_t \mid \Lambda)$ or $d'(X_t \mid \Lambda)$ can be intuitively viewed as a measure of distance from X_t to the current decision boundary.

Now let us define a discriminative objective function based on the above margins so that model Λ can be estimated through optimizing the discriminative objective function. Generally speaking, a discriminative objective function can be represented as a function of margins of all training samples in $\mathcal{D}$

$$\mathrm{F}_{\mathrm{DT}}(\Lambda) = \mathrm{f}\Big((d(X_1 \mid \Lambda), d(X_2 \mid \Lambda), \ldots, d(X_T \mid \Lambda)\Big). \qquad (4)$$

For the LME in [13], the function $\mathrm{f}(\cdot)$ is a min function. The LME objective function can be written as

$$\mathrm{F}_{\mathrm{LME}}(\Lambda) = \min_{X_t \in \mathcal{D}} d(X_t \mid \Lambda), \qquad (5)$$

where margin $d(X_t \mid \Lambda)$ is calculated as in (2) with s_t summed only over all competing hypotheses (excluding the correct label S_t from $\mathcal{M}_t$).

For the MCL estimation [11] or MMI [2], [35], the function $\mathrm{f}(\cdot)$ is a sum function. The objective function for MCL or MMI estimation can be represented as

$$\mathrm{F}_{\mathrm{MMI}}(\Lambda) = \sum_{X_t \in \mathcal{D}} d'(X_t \mid \Lambda), \qquad (6)$$

where margin $d'(X_t \mid \Lambda)$ is calculated as in (3) with s_t summed over all possible hypotheses (including the correct label S_t in $\mathcal{M}_t$).

For MCE [18], the function $\mathrm{f}(\cdot)$ is a sum-exp function (see [17] for details), the MCE objective function can be expressed as

$$\mathrm{F}_{\mathrm{MCE}}(\Lambda) = \sum_{X_t \in \mathcal{D}} \exp\left[d'(X_t \mid \Lambda)\right], \qquad (7)$$

where margin $d'(X_t \mid \Lambda)$ is calculated as in (3) with s_t summed over all possible hypotheses (including the correct label S_t in $\mathcal{M}_t$).

In summary, DT of statistical models can be formulated as an optimization problem to optimize the above discriminative objective function $\mathrm{F}_{\mathrm{DT}}(\Lambda)$ with respect to model parameters Λ, where $\mathrm{F}_{\mathrm{DT}}(\Lambda)$ is a function of margins of all training samples. In the following, we consider how to solve this optimization problem using efficient algorithms.

DT AS CONSTRAINED OPTIMIZATION

As shown in [21], when optimizing $\mathrm{F}_{\mathrm{DT}}(\Lambda)$ it is beneficial to impose a locality constraint on model parameters Λ to ensure that parameters do not deviate too much from their initial or current values. The locality constraint can be quantitatively computed based on Kullback-Leibler divergence (KLD). Therefore, DT of model parameters, Λ, can be formulated as the following iterative constrained maximization problem:

$$\Lambda^{(n+1)} = \arg\max_{\Lambda} \ \mathrm{F}_{\mathrm{DT}}(\Lambda) \tag{8}$$

$$\text{subject to } \ \mathcal{D}(\Lambda||\Lambda^{(n)}) \le \rho^2, \tag{9}$$

where $\mathcal{D}(\Lambda||\Lambda^{(n)})$ is the KLD between probability density functions specified by Λ and $\Lambda^{(n)}$, and $\rho > 0$ is a preset constant to control the search range. The constraint in (9) intuitively specifies a trust region for optimization in each iteration. As shown in [21], for some models, such as Gaussians, the KLD-based constraint in (9) can be further relaxed as quadratic form

$$||\Lambda - \Lambda^{(n)}||_2 \le \rho^2. \tag{10}$$

THE AM APPROACH

As shown above, DT of statistical models is an iterative optimization process. In each iteration, we need to solve a constrained maximization problem shown in (8) and (9). Following [15]–[17], in this section we introduce a general framework to solve this constrained maximization problem for one iteration. The key idea here is to find a simpler auxiliary function to approximate the original DT function in a close proximity of current model parameters if the original objective function is too complicated to optimize directly. Then, the auxiliary function is optimized by using some efficient optimization algorithms. Because of the locality constraint in (9) or (10), we can apply a variety of approximation strategies to construct the auxiliary function with a simpler function form. Based on the proximity approximation, the optimal solution found for the approximate auxiliary functions is expected to improve the original objective function as well. Then, in the next iteration, the original objective function can be approximated in another close proximity of this newly found solution based on the same approximation principle. This process repeats until convergence conditions are met for the original objective function. If the locality approximation is good enough, convergence of the above process is guaranteed provided the trust region ρ in the locality constraint is sufficiently small. Analogous to the popular EM algorithm [6], each iteration consists of two separate steps: i) Approximation step (A-step): the original objective function is approximated by an auxiliary function in a close proximity of current model parameters; ii) optimization step (M-step): the approximate auxiliary function is optimized under the locality constraints in either (9) or (10). Analogously, we call this method the AM algorithm. As explained below, the AM algorithm is more general than the EM algorithm since the expectation (E-step) in EM can be viewed as a proximity approximation method. More importantly, the AM algorithm can also deal with some more complicated objective functions arising from DT of statistical models with latent variables, e.g, mixtures of the e-family.

A-STEP

There are many different methods available to construct auxiliary functions to approximate an objective function in a close proximity. As shown in the section "A Unified View of Various DT Criteria," various DT objective functions can be viewed as a function of margins, which are defined as difference of two log likelihood functions, e.g., $\ln p(S_t, X_t|\Lambda)$ and $\ln p(s_t, X_t|\Lambda)$ as in (2) or (3), each of which is a log-sum term if model Λ belongs to mixtures of the e-family. As a result, the above DT objective functions, arising from DT of mixtures of e-family models, normally involves the difference of two log-sum terms so that they are nonconvex in nature and difficult to handle mathematically. In the A-step, we need to construct an auxiliary function to approximate the original DT objective function in a close proximity.

First of all, let us consider two strategies to approximate a single log-sum term arising from log likelihood function of mixtures of e-family distributions. The first method is to use the dominant term to approximate sum by replacing sum with max, which is named as M-approx. The other one is to use the well-known Jensen's inequality to approximate log-sum, named as E-approx. As shown below, both methods can approximate log-sum of e-family distributions by a lower-bounded concave function.

Assume model Λ belongs to mixtures of the e-family, its log-likelihood function, $\ln p(X, s|\Lambda)$, can be represented as log-sum of a finite number of e-family distributions, i.e., $\ln p(X, s|\Lambda) = \ln \sum_k f_k(\Lambda) \quad \text{with} \quad f_k(\Lambda) = w_k \exp\{A_k(\mathrm{x}_k) + \boldsymbol{\lambda} \cdot \mathrm{x}_k - \mathcal{K}_k(\boldsymbol{\lambda})\}$, where $\mathrm{x}_k \ (k = 1, \ldots, K)$ stands for sufficient statistics collected based on data X and label s for kth mixture component.

M-Approx (Max-Based Approximation)

In M-approx, the above log-sum term is approximated by its dominant component identified by an initial model Λ_0 as follows:

$$\ln p(X, s|\Lambda) = \ln\left[\sum_{k=1}^{K} f_k(\Lambda)\right] > \ln f_{\hat{k}}(\Lambda)$$

$$\equiv V(\Lambda|\Lambda_0) = A_{\hat{k}}(\mathrm{x}_{\hat{k}}) + \boldsymbol{\lambda}^T \mathrm{x}_{\hat{k}} - \mathcal{K}_{\hat{k}}(\boldsymbol{\lambda}), \tag{11}$$

where the dominant term $\hat{k}$ is selected based on an initial model Λ_0 as $\hat{k} = \arg\max_k f_k(\Lambda_0)$, and the auxiliary function $V(\Lambda|\Lambda_0)$ is a lower bound of the original log-sum.

E-Approx (Expectation-Based Approximation)

In E-approx, we first calculate the so-called posterior probability of kth function based on an initial model Λ_0 as: $\xi_k(\Lambda_0) = f_k(\Lambda_0)/\sum_{k=1}^{K} f_k(\Lambda_0)$. Obviously, they satisfy the sum-to-one constraint that $\sum_{k=1}^{K} \xi_k(\Lambda_0) = 1$. According to the Jensen's inequality, we have the following inequality held for $\ln p(X, s|\Lambda)$:

$$\ln p(X, s|\Lambda) = \ln\left[\sum_{k=1}^{K} f_k(\Lambda)\right]$$

$$= \ln\left[\sum_{k=1}^{K} \xi_k(\Lambda_0) \frac{f_k(\Lambda)}{\xi_k(\Lambda_0)}\right]$$

$$
\begin{aligned}
&\geq \sum_{k=1}^{K} \xi_k(\Lambda_0) \ln \frac{f_k(\Lambda)}{\xi_k(\Lambda_0)} \\
&= \sum_{k=1}^{K} \xi_k(\Lambda_0) \ln f_k(\Lambda) + H(\Lambda_0) \\
&\equiv Q(\Lambda|\Lambda_0) \\
&= \sum_{k=1}^{K} \xi_k(\Lambda_0) \cdot \{A_k(\mathbf{x}_k) + \boldsymbol{\lambda}^T \mathbf{x}_k - \mathcal{K}_k(\boldsymbol{\lambda})\} \\
&\quad + H(\Lambda_0), \qquad (12)
\end{aligned}
$$

where $H(\Lambda_0) = -\sum_{k=1}^{K} \xi_k(\Lambda_0) \ln \xi_k(\Lambda_0)$ denotes entropy calculated based on the posterior probabilities of $\xi_k(\Lambda_0)$. Furthermore, it is easy to verify that $Q(\Lambda|\Lambda_0)$ serves as a tangential lower bound of $\ln p(X, s|\Lambda)$: $\ln p(X, s|\Lambda)\,|_{\Lambda=\Lambda_0} = Q(\Lambda|\Lambda_0)|_{\Lambda=\Lambda_0}$ and $\partial \ln p(X, s|\Lambda)/\partial\Lambda\,|_{\Lambda=\Lambda_0} = \partial Q(\Lambda|\Lambda_0)/\partial\Lambda\,|_{\Lambda=\Lambda_0}$.

If model Λ belongs to mixture of the e-family, either M-approx or E-approx leads to a concave approximation function to each log-sum term.

Second, we apply the similar locality approximation (M-approx or E-approx) to margin since margin is defined as difference of two log-sum terms. However, one difficulty arises in approximating margins. For example, if we use the above E-approx to approximate each of the terms, the resultant E-approx margin involves difference of two Q functions, denoted as $d(X|\Lambda) = \ln p(X, s|\Lambda) - \ln p(X, s'|\Lambda) \approx Q^+(\Lambda|\Lambda_0) - Q^-(\Lambda|\Lambda_0) \equiv \tilde{d}(X|\Lambda)$, where the approximate margin, $\tilde{d}(X|\Lambda)$, remains tangential to the real margin at Λ_0 but the lower bound property in (12) does not hold anymore. Moreover, the approximate margin $\tilde{d}(X|\Lambda)$ is neither convex nor concave because it is a difference of two concave functions. However, the E-approx margin can still be viewed as a close proximity approximation of $d(X|\Lambda)$ at Λ_0 with accuracy up to the first order, as shown in Figure 1. It is clear that the E-approx margin $\tilde{d}(X|\Lambda)$ remains as a good approximation to the true margin as long as trust region ρ is sufficiently small.

Finally, since the discriminative objective function $\mathrm{F}_{\mathrm{DT}}(\Lambda)$ is a function of margins as shown in (4), an auxiliary function of $\mathrm{F}_{\mathrm{DT}}(\Lambda)$ can be constructed by substituting the approximate margins $\tilde{d}(X|\Lambda)$ into (4) in place of the original margins. The auxiliary function is denoted as $\widetilde{\mathrm{F}}_{\mathrm{DT}}(\Lambda)$. Similarly to the approximate margins $\tilde{d}(X|\Lambda)$, the auxiliary function $\widetilde{\mathrm{F}}_{\mathrm{DT}}(\Lambda)$ is neither concave nor convex. Moreover, $\widetilde{\mathrm{F}}_{\mathrm{DT}}(\Lambda)$ does not serve as a strict lower bound of the original discriminative function $\mathrm{F}_{\mathrm{DT}}(\Lambda)$. If E-approx is used, it only remains tangential to $\mathrm{F}_{\mathrm{DT}}(\Lambda)$ at Λ_0, which is similar to margins in Figure 1.

M-STEP

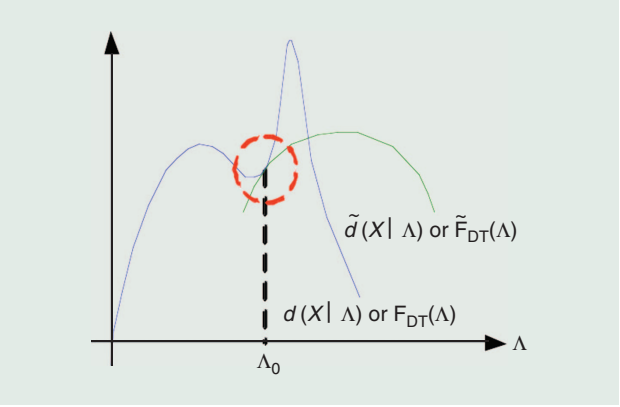
In the M-step, we need to consider two critical issues: how to optimize the nonconvex auxiliary function and how to ensure that the original objective function $\mathrm{F}_{\mathrm{DT}}(\Lambda)$ also improves when the auxiliary function $\widetilde{\mathrm{F}}_{\mathrm{DT}}(\Lambda)$ is optimized. Unlike in the EM algorithm, both issues are not straightforward to address in the AM algorithm.

As for the second issue, if we adopt the locality constraint in optimization as in (9) or (10), the found optimal point of $\widetilde{\mathrm{F}}_{\mathrm{DT}}(\Lambda)$ is guaranteed to improve $\mathrm{F}_{\mathrm{DT}}(\Lambda)$ when the trust region ρ is sufficiently small. In practice, this has also been observed in experiments on speech recognition and text categorization when a proper ρ is selected (see [19], [28], and [38] for experimental details).

Let us return to the first issue regarding optimization of the nonconvex auxiliary function. This is a major difference between the AM method and the conventional EM algorithm, where the auxiliary function is always concave for mixtures of e-family models. In fact, many different approaches have been proposed to optimize this nonconvex and nonconcave auxiliary function in the literature, such as the constrained line search [21] and trust-region methods [22], [37]. In this article, we focus on a group of methods to optimize the auxiliary function using convex optimization methods. More specifically, we apply convex relaxation techniques to convert the underlying problem into a convex optimization problem so that the global optimal point of the relaxed convex optimization problem can be efficiently found.

Depending on the underlying models, a variety of convex relaxation methods may be applied to convert the nonconvex optimization problem in the M-step into a standard convex optimization problem as in [19], [38], and [28], such as linear programming (LP), quadratic programming (QP), SOCP, and semidefinite programming (SDP) [23], so that some standard convex optimization algorithms can be applied to optimize the relaxed auxiliary functions under the proximity constraint in (9) or (10). Any local optimal point is always globally optimal in a convex optimization problem. Therefore, the advantage of using convex optimization in M-step is that optimization can be efficiently and reliably solved even for very large-scale models.

In the remainder of this article, we introduce three different convex relaxation techniques that have been successfully applied to speech recognition and text categorization. As shown in [15] and [19], the DT objective function of many Gaussian-derived statistical models, such as GMM and Gaussian mixture HMMs, can be approximated by either



[FIG1] Illustration of E-approx in the AM method.

M-approx or E-approx as indefinite quadratic auxiliary functions. As in [19], optimization of the quadratic auxiliary functions can be represented as an equivalent matrix form. If the self-constrained matrix variables can be relaxed as positive semidefinite matrices, the original nonconvex maximization problem in M-step can be converted into an SDP problem. We first discuss this work for large-margin-based DT in the section "Case Study 1: LME of CDHMMs Using SDP Based on *N*-Best Lists." In [38] and [39], a different approach is taken to convert maximization of indefinite quadratic form to convex optimization, where the indefinite Hessian matrix is decomposed based on eigenvectors with positive and negative eigenvalues. All quadratic terms related to negative eigenvalues are replaced by a linear term along with some relaxed convex constraints. In this way, the original maximization of indefinite quadratic form can be relaxed into another convex optimization problem, namely SOCP. As our second case study, we discuss this work for MMI-based DT in the section "Case Study 2: MMI Training of HMMs Using SOCP Based on Word Graphs." As in [25] and [28], the DT objective functions of various discrete statistical models based on multinomial distributions, such as MMM, Markov chain model, discrete density HMMs and etc., can be approximated by either M-approx or E-approx as linear auxiliary functions. Then, in the M-step, optimization of these linear auxiliary functions can be converted into a standard linear programming (LP) problem if the sum-to-one constraints are relaxed. We discuss this work in our third case study in the section "Case Study 3: Large Margin MMMs Using LP."

A UNIFIED VIEW FOR VARIOUS DISCRIMINATIVE CRITERIA IS PRESENTED, CENTERING ON THE CONCEPT OF MARGIN.

CASE STUDY 1: LME OF CDHMMs USING SDP BASED ON *N*-BEST LISTS

As our first case study we use the above AM algorithm to perform LME of Gaussian mixture HMMs in ASR. We assume all competing hypotheses of each sentence are linearly encoded in an N-best list, which is applicable to small vocabulary ASR tasks. In the A-step, we use the M-approx method to construct the auxiliary function. Then, in the M-step, we showcase the use of a well-known convex relaxation technique to convert the LME problem into an SDP problem.

Assume all speech units are modeled by some Gaussian mixture HMMs, denoted as Λ. For a complete utterance, X, consisting of a sequence of feature vectors, i.e., $X = \{x_1, x_2, \cdots, x_R\}$, let us examine the log-likelihood function of HMMs, i.e., $\ln p(X|\Lambda)$. Gaussian mixture HMMs have hidden variables $\mathbf{s}$, the unobserved state sequence, and $\mathbf{l}$, the unobserved Gaussian mixture labels. We have

$$\ln p(X|\Lambda) = \ln \left[\sum_{s,l} p(X, \mathbf{s}, \mathbf{l} \mid \Lambda) \right], \tag{13}$$

where $p(X, \mathbf{s}, \mathbf{l}|\Lambda)$ denotes the probability of X calculated along a single path $\mathbf{s}$ and $\mathbf{l}$. In this case, $p(X, \mathbf{s}, \mathbf{l}|\Lambda)$ is the product of Gaussians and multinomials. Therefore, it can be represented in the standard form of e-family distribution, $\exp\{\mathcal{K}_{\mathbf{sl}}(\boldsymbol{\lambda}) + \boldsymbol{\lambda} \cdot \mathbf{x}_{\mathbf{sl}} + A_{\mathbf{sl}}(\mathbf{x}_{\mathbf{sl}})\}$, where $\boldsymbol{\lambda}$ denotes natural parameter of HMMs (to be explained later) and $\mathbf{x}_{\mathbf{s,l}}$ stands for sufficient statistics collected along a single path $\mathbf{s}, \mathbf{l}$. Therefore, HMMs belong to mixtures of the e-family.

Given a set of training data $\mathcal{T} = \{X_1, X_2, \ldots, X_T\}$, we know the true word transcriptions for all utterances in $\mathcal{T}$, denoted as $\mathcal{L} = \{S_1, S_2, \ldots, S_T\}$. For each X_t, its competing hypotheses are encoded as a linear N-Best list, denoted as $\mathcal{M}_t$, which is generated from an N-best Viterbi decoding process. If we accept the margin definition in (2), LME in (5) can be converted into the following maximin optimization problem subject to the locality constraint:

$$\tilde{\Lambda} = \arg\max_{\Lambda} \min_{X_t \in \mathcal{T},\, s_t \in \mathcal{M}_t,\, s_t \neq S_t} [\ln p(X_t|S_t, \Lambda) - \ln p(X_t|s_t, \Lambda)]. \tag{14}$$

Conceptually speaking, LME attempts to reestimate the HMM parameters to make the decision boundary stay as far from training samples as possible. In case all of the training samples are not separable, as in [13], we may consider only a subset of positive tokens in (14) instead of the whole training set $\mathcal{T}$ for each iteration. By doing so, we can avoid infeasibility caused by training errors and meanwhile significantly reduce optimization complexity. Another formal treatment is to follow the soft margin concept in SVM and extend the above LME formulation to maximize a linear combination of minimum margin and average training error rate, which results in the so-called soft LME method [14]. Since the soft LME has a similar mathematical formulation, we only introduce LME in this article and interested readers may refer to [14] for details on the soft LME.

As shown in [19], the above maximin optimization problem can be equivalently converted into the following constrained maximization problem by introducing a new variable θ ($\theta > 0$) as a common lower bound to represent min part of all terms in (14) along with the constraints that every item must be larger than or equal to θ.

PROBLEM 1

$$\max_{\Lambda, \theta} \ \theta \tag{15}$$

subject to:

$$\ln p(X_t|S_t, \Lambda) - \ln p(X_t|s_t, \Lambda) \geq \theta \tag{16}$$
$$(\forall X_t \in \mathcal{T} \text{ and } \mathrm{s_t} \in M_\mathrm{t} \text{ and } \mathrm{s_t} \neq \mathrm{S_t})$$

$$\|\Lambda - \Lambda^{(\eta)}\|_2 \leq \rho^2 \text{ and } \theta \geq 0. \tag{17}$$

In the following, we consider how to convert the above optimization into an SDP problem. If we consider to estimate acoustic models only, we focus on acoustic score $p(X_t|S_t, \Lambda)$ in (13). Based on the concept of M-approx in (11), we use the best Viterbi path, $\hat{\mathbf{s}}$ and $\hat{\mathbf{l}}$, to approximate the above summation instead of summing over all possible paths. In each iteration, the best Viterbi path can be derived based on the current model, $\Lambda^{(n)}$, by the following max operation using the well-known Viterbi algorithm as $\{\hat{\mathbf{s}}, \hat{\mathbf{l}}\} = \arg\max_{\mathbf{s},\mathbf{l}} \; p(X, \mathbf{s}, \mathbf{l}|\Lambda^{(n)})$ Thus, the log-likelihood function can be approximated as follows:

THE AM ALGORITHM CAN ALSO DEAL WITH MORE COMPLICATED OBJECTIVE FUNCTIONS ARISING FROM DISCRIMINATIVE LEARNING OF STATISTICAL MODELS WITH LATENT VARIABLES, E.G., MIXTURES OF E-FAMILY.

$$\ln p(X|S, \Lambda) > \ln p(X, \hat{\mathbf{s}}, \hat{\mathbf{l}}|S, \lambda) = A_{\hat{s}\hat{l}}(\mathbf{x}_{\hat{s}\hat{l}}) + \boldsymbol{\lambda}^T \mathbf{x}_{\hat{s}\hat{l}} - \mathcal{K}^+_{\hat{s}\hat{l}}(\boldsymbol{\lambda}) \equiv \mathcal{V}^+(\boldsymbol{\lambda}), \quad (18)$$

where $\boldsymbol{\lambda}$ denotes the natural parameters of HMM, and $\mathbf{x}_{\hat{s}\hat{l}}$ the sufficient statistics collected along the path $\{\hat{\mathbf{s}}, \hat{\mathbf{l}}\}$.

For simplicity, we only estimate Gaussian mean vectors of HMMs based on the large margin principle while keeping all other HMM parameters constant. It is possible to extend the method to estimate other parameters in CDHMMs, such as covariance matrices and mixtures weights. See [16] for details.

Furthermore, we assume there are M Gaussians in total in the whole HMM set Λ, denoted as $\{1, 2, \ldots, M\}$. We denote each Gaussian as $\mathcal{N}(\mu_k, \Sigma_k)$ with $k \in [1, M]$. For notational convenience, the optimal Viterbi path $\hat{\mathbf{s}}$ and $\hat{\mathbf{l}}$ can be equivalently represented as a sequence of Gaussian indices, i.e., $\mathbf{j} = \{j_1, j_2, \ldots, j_R\}$, where $j_t \in [1, M]$ is the index of Gaussians along the optimal Viterbi path $\{\hat{\mathbf{s}}, \hat{\mathbf{l}}\}$. As in case A) of Table 1, the natural parameters of HMM, $\boldsymbol{\lambda}$, is a concatenated vector of all Gaussian mean vectors, i.e., $\boldsymbol{\lambda} = [\mu_1, \ldots, \mu_M]$, after some mathematical manipulation, $\mathcal{V}^+(\lambda)$ in (18) can be represented as

$$\mathcal{V}^+(\boldsymbol{\lambda}) = \sum_{k=1}^{M} \left[\mu_k^T \mathbf{x}_k^+ - \frac{1}{2}(\mathbf{x}_k^+)^T \mathbf{x}_k^+ - K_k^+(\mu_k) \right] + c_1, \quad (19)$$

where c_1 is a constant independent of all Gaussian means and $\mathbf{x}_k^+$ denotes sufficient statistics collected along the best path $\mathbf{j}$ for kth Gaussian ($k \in [1, M]$) as $\mathbf{x}_k^+ = \sum_{r=1}^{R} \Sigma_k^{-1} x_r \delta(j_r - k)$, with $\delta(\cdot)$ for Kronecker delta function, and the cumulant generating function $\mathcal{K}_k^+(\mu_k) = 1/2 \mu_k^T \Phi_k^+ \mu_k$ with $\Phi_k^+ = \sum_{r=1}^{R} \Sigma_k^{-1} \delta(j_r - k)$.

As in [39], assume we construct two matrices as follows:

$$\mathbf{Z}_k := \begin{pmatrix} 1 & \mu_k^T \\ \mu_k & \mu_k \mu_k^T \end{pmatrix} \quad (20)$$

$$\mathbf{A}_k^+ := \frac{1}{2} \begin{pmatrix} \mathbf{x}_k^{+T} \mathbf{x}_k^+ & -\mathbf{x}_k^{+T} \\ -\mathbf{x}_k^+ & \Phi_k^+ \end{pmatrix}. \quad (21)$$

It is easy to verify that $\mathcal{V}^+(\boldsymbol{\lambda}) = c_1 - \sum_k \mathrm{Tr}(\mathbf{A}_k^+ \mathbf{Z}_k)$. Based on the same idea of M-approx, $\ln p(X_t, s_t|\Lambda)$ in (16) can be approximated as $\mathcal{V}^-(\boldsymbol{\lambda}) = c_2 - \sum_k \mathrm{Tr}(\mathbf{A}_k^- \mathbf{Z}_k)$. Therefore, the constraint in (16) can be approximated as

$$\ln p(X|S, \Lambda) - \ln p(X|s, \Lambda) \approx \mathcal{V}^+(\boldsymbol{\lambda}) - V^-(\boldsymbol{\lambda}) = c - \sum_k \mathrm{Tr}(\mathbf{A}_k \mathbf{Z}_k), \quad (22)$$

where $c = c_1 - c_2$ and $\mathbf{A}_k = \mathbf{A}_k^+ - \mathbf{A}_k^-$.

For the locality constraint in (17), we can rewrite it using $\mathbf{Z}_k$ as: $R(\Lambda) = \sum_{k=1}^{\mathcal{K}} (\mu_k - \mu_k^{(n)})^T (\mu_k - \mu_k^{(n)}) = \sum_{k=1}^{M} \mathrm{Tr}(\mathbf{R}_k \mathbf{Z}_k) \le \rho^2$, where matrix $\mathbf{R}_k$ is built as in (21) with $\mathbf{x}_k^+$ and Φ_k^+ replaced by $\mu_k^{(n)}$ and identity matrix I, respectively.

In this way, we have a number of small variable matrices $\mathbf{Z}_k$ ($1 \le k \le M$), each of which is constructed from a Gaussian mean vector μ_k as in (20). Obviously, the rank of $\mathbf{Z}_k$ is equal to one. Following [3] and [24], to transform this into an SDP problem, we relax the rank-one condition to a positive semidefinite condition for all $\mathbf{Z}_k$ as follows:

$$\mathrm{rank}(\mathbf{Z}_k) = 1 \;\Rightarrow\; \mathbf{Z}_k \succcurlyeq 0 \quad \forall k \in (1, 2, \ldots, M). \quad (23)$$

Moreover, we must impose another constraint that the top-left element of $\mathbf{Z}_k$ is equal to unity, i.e., $\{\mathbf{Z}_k\}_{1,1} = 1$. This constraint can be easily cast as a linear constraint in matrix form.

Finally, we can formulate the original LME problem as the following SDP problem after the relaxation in (23).

PROBLEM 2

$$\max_{\Lambda, \theta} \; \theta \quad (24)$$

subject to

$$\sum_{k=1}^{M} \mathrm{Tr}(\mathbf{A}_k^{(t)} \mathbf{Z}_k) + \theta \le c_t \quad (25)$$
$$(\forall \mathrm{X_t} \in \mathcal{T} \text{ and } \mathrm{s_t} \in \mathcal{M}_\mathrm{t} \text{ with } \mathrm{s_t} \neq \mathrm{S_t})$$

$$\sum_{k=1}^{M} \mathrm{Tr}(\mathbf{R}_k \mathbf{Z}_k) \le \rho^2 \quad (26)$$

$$\theta \ge 0 \;\text{ and }\; \mathbf{Z}_k \succcurlyeq 0 \;\text{ for all }\; k \in (1, 2, \ldots, M) \quad (27)$$

$$\{\mathbf{Z}_k\}_{1,1} = 1 \;\text{ for all }\; k \in (1, 2, \ldots, M). \quad (28)$$

Problem 2 is a standard SDP problem, where optimization is performed with respect to all variable matrices $\mathbf{Z}_k$

and θ using standard convex optimization tools. After that, Gaussian mean vector μ_k can be updated based on the found SDP solution $\mathbf{Z}_k^*$.

In [19] and [39], the above SDP-based LME method has been applied to a speaker-independent connected digit string recognition task using the standard TIDIGITS database and it has achieved one of the best recognition performance ever reported in this task. See [19] and [39] for experimental details.

UNDER SOME MINOR APPROXIMATION AND RELAXATION CONDITIONS, DISCRIMINATIVE LEARNING OF MMMs CAN BE FORMULATED AS LP PROBLEMS, WHICH CAN BE SOLVED FAIRLY EFFICIENTLY FOR VERY LARGE-SCALE TASKS.

CASE STUDY 2: MMI TRAINING OF HMMs USING SOCP BASED ON WORD GRAPHS

In this section, as another case study, we consider to use the MMI criterion in (6) to estimate Gaussian mean vectors of HMMs for large vocabulary continuous speech recognition (LVCSR), where competing hypotheses for each training sentence are encoded in a word graph. We introduce a different convex relaxation method in [38] to convert DT problem into a different convex optimization problem, i.e., SOCP, for better computational efficiency.

For each training utterance $X_t = \{x_{t1}, \ldots, x_{tR}\}$ in a training set $\mathcal{T}$, assume its true transcription is S_t and its competing hypotheses are represented in a word graph, denoted as $\mathcal{M}_t$. The MMI objective function can be viewed as a sum of margins, as defined in (3). In this work, we assume language model scores $p(S_t)$ and $p(s_t)$ are constant. As a result, we focus on acoustic model scores $p(X_t|S_t, \Lambda)$ and $p(X_t|s_t, \Lambda)$, and consider how to approximate them using the E-approx method. As in the previous section, we only consider to estimate Gaussian mean vectors and assume there are in total M Gaussians in Λ.

Based on E-approx in (12), we have

$$\begin{aligned}\ln p(X_t|S_t, \Lambda) &= \ln \sum_{\mathrm{s,l}} p(X_t, \mathrm{s}, \mathrm{l}|S_t, \Lambda) \\ &\geq \sum_{\mathrm{s,l}} \ln p(X_t, \mathrm{s}, \mathrm{l}|S_t, \Lambda) \cdot \Pr(\mathrm{s}, \mathrm{l}|X_t, S_t, \Lambda^{(n)}) \\ &= \sum_{k=1}^{M} \left[\mu_k^T \mathbf{x}_{tk}^+ - \frac{1}{2}(\mathbf{x}_{tk}^+)^T \mathbf{x}_{tk}^+ - \frac{1}{2}\mu_k^T Q_{tk}^+ \mu_k \right] + b_1 \\ &\equiv \mathcal{Q}_t^+(\Lambda|\Lambda^{(n)}), \end{aligned} \tag{29}$$

where $\mathbf{x}_{tk}^+ = \sum_{r=1}^{R} \Sigma_k^{-1} x_{tr} \cdot \xi_k^+(t,r)$ and $Q_{tk}^+ = \sum_{r=1}^{R} \Sigma_k^{-1} \cdot \xi_k^+(t,r)$ with $\xi_k^+(t,r)$ denotes posterior probability of residing in kth Gaussian given rth feature vector, x_{tr} of X_t, which can be calculated efficiently by running the forward-backward algorithm against HMMs of reference S_t.

Similarly, we can apply E-approx to approximate $\ln \sum_{s_t \in \mathcal{M}_t} p(X_t, s_t|\Lambda)$ in (3) as follows:

$$\begin{aligned}\ln \sum_{s_t \in \mathcal{M}_t} p(X_t, s_t|\Lambda) &\geq \mathcal{Q}_t^-(\Lambda|\Lambda^{(n)}) \\ &= \sum_{k=1}^{M} \left[\mu_k^T \mathbf{x}_{tk}^- - \frac{1}{2}(\mathbf{x}_{tk}^-)^T \mathbf{x}_{tk}^- - \frac{1}{2}\mu_{tk}^T Q_{tk}^- \mu_k \right] + b_2, \end{aligned} \tag{30}$$

where $\mathbf{x}_{tk}^- = \sum_{r=1}^{R} \Sigma_k^{-1} x_{tr} \cdot \xi_k^-(t,r)$ and $Q_{tk}^- = \sum_{r=1}^{R} \Sigma_k^{-1} \cdot \xi_k^-(t,r)$ with $\xi_k^-(t,r)$ denotes posterior probability of residing in kth Gaussian given rth feature vector, x_{tr}, of X_t, which can be calculated efficiently by running the forward-backward algorithm in word graph $\mathcal{M}_t$ for all relevant HMMs. Refer to [33] for details on running the forward-backward algorithm on a word graph.

Based on (6), given a training set $\mathcal{D}$, the MMI objective function $\mathrm{F}_{\mathrm{MMI}}(\Lambda)$ can be approximated by an auxiliary function that involves difference of $\mathcal{Q}_t^+$ and $\mathcal{Q}_t^-$ as follows:

$$\begin{aligned}\mathrm{F}_{\mathrm{MMI}}(\Lambda) &\approx \sum_{X_t \in \mathcal{D}} [\mathcal{Q}_t^+(\Lambda|\Lambda^{(n)}) - \mathcal{Q}_t^-(\Lambda|\Lambda^{(n)})] + h_r \\ &= \boldsymbol{\lambda}^T \mathbf{Q} \boldsymbol{\lambda} + \mathbf{q}^T \boldsymbol{\lambda} + \mathrm{g}, \end{aligned} \tag{31}$$

where $\boldsymbol{\lambda}$ denotes natural parameters of models, which is a super-vector constructed by concatenating all Gaussian mean vectors $\boldsymbol{\lambda} = [\mu_1, \ldots, \mu_M]$, $\mathbf{Q}$ is a block diagonal matrix with all matrices, $Q_k = -1/2 \sum_{X_t \in \mathcal{D}} [Q_{tk}^+ - Q_{tk}^-]$ for all $k = 1, \ldots, M$, aligned diagonally, and $\mathbf{q}$ is another super-vector constructed by concatenating vectors $q_k = \sum_{X_t \in \mathcal{D}} [\mathbf{x}_{tk}^+ - \mathbf{x}_{tk}^-]$ for $k = 1, \ldots, M$, and g is a constant.

The locality constraint in (9) can also be represented as a spheral constraint of $\boldsymbol{\lambda}$ as $\|\Lambda - \Lambda^{(n)}\|_2 = \|\boldsymbol{\lambda} - \boldsymbol{\lambda}^{(n)}\|_2$.

Therefore, the MMI training of HMMs can be converted into the following iterative constrained maximization problem.

PROBLEM 3

$$\max_{\boldsymbol{\lambda}} \boldsymbol{\lambda}^T \mathbf{Q} \boldsymbol{\lambda} + \mathbf{q}^T \boldsymbol{\lambda} + \mathrm{g} \tag{32}$$

subject to:

$$R(\boldsymbol{\lambda}) = \|\boldsymbol{\lambda} - \boldsymbol{\lambda}^{(n)}\|_2 \leq \rho^2. \tag{33}$$

The constraint in (33) is a convex constraint. The objective function is in standard quadratic form but it is not convex since we cannot guarantee matrix $\mathbf{Q}$ to be positive semidefinite. To deal with this problem, using the convex relaxation technique in [38], we first decompose the matrix $\mathbf{Q}$ according to its eigenvalues and eigenvectors, λ_m and v_m $(m \in \mathcal{M})$, as: $\mathbf{Q} = \mathbf{Q}^+ + \sum_{m \in \mathcal{M}, \lambda_m < 0} \lambda_{\mathrm{m}} \cdot \mathrm{v}_{\mathrm{m}} (\mathrm{v}_{\mathrm{m}})^{\mathrm{T}}$, where $Q^+ = \sum_{m \in \mathcal{M}, \lambda_m > 0} \lambda_{\mathrm{m}} \cdot \mathrm{v}_{\mathrm{m}} (\mathrm{v}_{\mathrm{m}})^{\mathrm{T}}$ is a positive semidefinite matrix since it is constructed only from all eigenvectors with positive eigenvalues. As in [38], the quadratic term in (32) can be decomposed as two terms: $\boldsymbol{\lambda}^T \mathbf{Q} \boldsymbol{\lambda} = \boldsymbol{\lambda}^T \mathbf{Q}^+ \boldsymbol{\lambda} + \sum_{m \in \mathcal{M}, \lambda_m < 0} \lambda_m z_m$, where

all eigenvectors with negative eigenvalues are replaced by some new variables, z_m, subject to some nonconvex equality constraints: $z_m = \boldsymbol{\lambda}^T \mathrm{v_m} (\mathrm{v_m})^T \boldsymbol{\lambda}$.

Following [38], the above nonconvex equality constraints are relaxed as the following convex inequality constraints:

$$\boldsymbol{\lambda}^T \mathrm{v_m} (\mathrm{v_m})^T \boldsymbol{\lambda} \le \mathrm{z_m} \le 2\mu_{\mathrm{m}}^{(\mathrm{n})} \mu_{\mathrm{m}} - \mu_{\mathrm{m}}^{(\mathrm{n})} \mu_{\mathrm{m}}^{(\mathrm{n})} + \rho^2, \qquad (34)$$

for all $m \in \mathcal{M}$ and $\lambda_m < 0$.

The above SOCP relaxation can be intuitively illustrated in Figure 2. The original MMIE problem can be viewed as optimizing the objective function along the solid quadratic curve segment, which is not a convex set. After relaxation, optimization is performed within the shaded area under the linear upper bound, which becomes a convex set.

Finally, Problem 3 can be relaxed and converted into the following convex optimization problem.

PROBLEM 4

$$\max_{\boldsymbol{\lambda},\, z_m} \left[\boldsymbol{\lambda}^{\mathrm{T}} \mathrm{Q}^{+} \boldsymbol{\lambda} + \sum_{\mathrm{m} \in \mathcal{M},\, \lambda_{\mathrm{m}} < 0} \lambda_{\mathrm{m}} z_{\mathrm{m}} + \mathrm{q}^{\mathrm{T}} \boldsymbol{\lambda} + \mathrm{g} \right] \qquad (35)$$

subject to:

$$R(\Lambda) = \|\boldsymbol{\lambda} - \boldsymbol{\lambda}^{(n)}\| \le \rho^2 \qquad (36)$$

$$\boldsymbol{\lambda}^T \mathrm{v_m} (\mathrm{v_m})^T \boldsymbol{\lambda} \le z_{\mathrm{m}} \le 2\mu_{\mathrm{m}}^{(\mathrm{n})} \mu_{\mathrm{m}} - \tilde{\mu}_{\mathrm{m}}^{(\mathrm{n})} \mu_{\mathrm{m}}^{(\mathrm{n})} + \rho^2 \qquad (37)$$

$$\text{for all } m \in \mathcal{M} \quad \text{and} \quad \lambda_{\mathrm{m}} < 0.$$

As shown in [38], Problem 4 is a convex quadratic programming problem and it can be easily converted into an SOCP problem so that it can be solved by any SOCP solver.

In [36], this SOCP-based MMI training has been successfully applied to a large vocabulary continuous speech recognition task using the WSJ-5k data set and it has been shown to outperform some conventional DT methods.

CASE STUDY 3: LARGE-MARGIN MMMs USING LP

In this section, we apply the AM-based DT to a different type of generative models, derived from multinomial distributions. This type of statistical models is normally used to model discrete data, such as text and symbolic data. As our third case study, we consider to apply LME to MMMs for text categorization using the AM algorithm. Under some minor approximation and relaxation conditions, discriminative learning of MMMs can be formulated as LP problems, which can be solved fairly efficiently for very large-scale tasks.

The goal of text categorization is the automatic classification of text documents into one of the predefined categories. In text processing, we normally select a set of features to represent each text document. Some widely used features include occurrence frequency of a particular word or an n-gram or a phrase or even a given syntax structure in a text document. Each of these is called a feature. In this way, each text document can be represented as a feature vector, $X = (x_1, \ldots, x_D)$, where D stands for the total number of selected features and each x_d represents frequency of dth feature in the document.

THE GOAL OF TEXT CATEGORIZATION IS THE AUTOMATIC CLASSIFICATION OF TEXT DOCUMENTS INTO ONE OF THE PREDEFINED CATEGORIES.

In MMM, each class is modeled by several multinomial models. The contributions from these multinomial models are linearly combined as in other finite mixture models. In an MMM, denoted as $\lambda_i = \{\mu_{ikd}, w_{ik} \mid \forall k, d\}$, given a document with its feature vector $X_t = (x_{t1}, \ldots, x_{tD})$, the probability of observing the document from this class is computed as

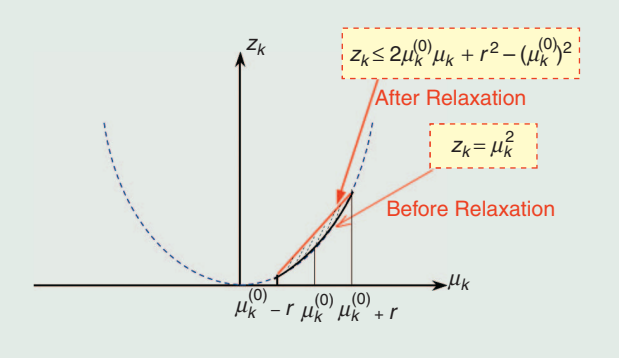
$$\Pr(X_t \mid \lambda_i) = \sum_{k=1}^{K} \left[w_{ik} \cdot C_t \cdot \prod_{d=1}^{D} \mu_{ikd}^{x_{td}} \right], \qquad (38)$$

where μ_{ikd} denotes the conditional probability of the dth feature in kth mixture of model λ_i. Obviously, the conditional probabilities and mixture weights satisfy the sum-to-one constraint as follows:

$$\sum_{d} \mu_{ikd} = 1 \ (\forall i, k) \ \text{ and } \ \sum_{k} w_{ik} = 1 \ (\forall i). \qquad (39)$$

For simplicity, we use Λ to denote all MMMs representing all classes. MMM also belongs to mixtures of the e-family. Each component can be represented as the canonic form in either case C) or D) in Table 1. In the generative learning framework, the MMM parameters can be estimated from available training based on ML estimation using the EM algorithm [6]. In the following we use the LME method in (5) to estimate MMMs for document classification based on the principle of maximizing the minimum separation margin.

Assume the entire training set for all classes is given as $\{X_t | t = 1, \ldots, T\}$ along with class labels for all documents, denoted as $\{S_t | t = 1, \ldots, T\}$, where each document is



[FIG2] Conceptual illustration (one-dimensional case) of the proposed SOCP relaxation in DT of HMMs.

represented by its feature vector X_t and its class label is known as S_t. Assume we accept the margin definition in (2). We have

DISCRIMINATIVE LEARNING METHODS HAVE ACHIEVED MANY SUCCESSES IN SPEECH AND LANGUAGE PROCESSING DURING THE PAST DECADES.

$$d(X_t|\Lambda) = \min_{s_t \neq S_t} \left[\ln \Pr(X_t|\lambda_{S_t}) - \ln \Pr(X_t|\lambda_{s_t}) \right] = \min_{s_t \neq S_t} d_{S_t s_t}(X_t), \quad (40)$$

where $d_{S_t s_t}(X_t) = \ln \Pr(X_t|\lambda_{S_t}) - \ln \Pr(X_t|\lambda_{s_t})$ is called decision margin of X_t between models λ_{S_t} and λ_{s_t}.

As in the previous sections, LME in (5) leads to the following constrained maximization problem.

PROBLEM 5

$$\max_{\Lambda, \theta} \ \theta \ \ (\theta > 0)$$

subject to

$$\ln \Pr(X_t|\lambda_{S_t}) - \ln \Pr(X_t|\lambda_{s_t}) \geq \theta, \ \forall X_t, S_t \text{ and } \forall s_t \ (s_t \neq S_t).$$

As in case C) of Table 1, the natural parameters of MMMs are represented in the logarithm scale as follows: $\psi_{ik} \equiv \ln w_{ik}$ and $\varphi_{ikd} \equiv \ln \mu_{ikd}$. If we represent each multinomial as the constrained e-family form in case C) of Table 1, the E-approx in (12) can be applied to approximate $\ln \Pr(X_t|\lambda_i)$ as

$$\begin{aligned} \ln \Pr(X_t|\lambda_i) &= \ln \sum_{k=1}^{K} \left[w_{ik} \cdot C_t \cdot \prod_{d=1}^{D} \mu_{ikd}^{x_{td}} \right] \\ &\geq \sum_{k=1}^{K} \sum_{d=1}^{D} [\gamma_{tik} x_{td}] \cdot \varphi_{ikd} + \sum_{k=1}^{K} \gamma_{tik} \cdot \psi_{ik} + h_{ti} \\ &\equiv Q_t(\lambda_i|\lambda_i^{(n)}), \end{aligned} \quad (41)$$

where γ_{tik} is the so-called responsibility of kth component in ith model given X_t, and h_{ti} is a constant independent of MMM parameters. In the same way, the decision margin $d_{\mathrm{i}\mathrm{j}_2}(X_t|\Lambda)$ can be approximated based on E-approx as a linear function of all ψ_{ik} and ϕ_{ikd} as follows:

$$\begin{aligned} d_{\mathrm{i}\mathrm{j}_2}(X_t|\Lambda) &= \ln \Pr(X_t|\lambda_{\mathrm{j}_1}) - \ln \Pr(X_t|\lambda_{\mathrm{j}_2}) \\ &\approx Q_t(\lambda_{\mathrm{j}_1}|\lambda_{\mathrm{j}_1}, X_t) - Q_t(\lambda_{\mathrm{j}_2}|\lambda_{\mathrm{j}_2}, X_t) \\ &= \sum_i \sum_{k=1}^{K} e_{ik}^{\mathrm{j}\mathrm{j}_2} \cdot \psi_{ik} + \sum_i \sum_{k=1}^{K} \sum_{d=1}^{D} f_{ikd}^{\mathrm{j}\mathrm{j}_2} \cdot \phi_{ikd} + g_{t\mathrm{j}_1\mathrm{j}_2}, \end{aligned} \quad (42)$$

where all coefficients are computed as: $e_{ik}^{\mathrm{j}\mathrm{j}_2} = \gamma_{t\mathrm{j}_1 k} \cdot \delta(i - \mathrm{j}_1) - \gamma_{t\mathrm{j}_2 k} \cdot \delta(i - \mathrm{j}_2)$, and $f_{ikd}^{\mathrm{j}\mathrm{j}_2} = \gamma_{t\mathrm{j}_1 k} x_{td} \cdot \delta(i - \mathrm{j}_1) - \gamma_{t\mathrm{j}_2 k} x_{td} \cdot \delta(i - \mathrm{j}_2)$, and $g_{t\mathrm{j}_1\mathrm{j}_2} = h_{t\mathrm{j}_1} - h_{t\mathrm{j}_2}$.

We need to impose a locality constraint for all parameters during the LME optimization. One simple choice is to use the following box constraints: $\phi_{ikd}^{(n)} - \tau_1 \leq \phi_{ikd} \leq \phi_{ikd}^{(n)} + \tau_1$, $\forall i, k, d$, and $\psi_{ik}^{(n)} - \tau_2 \leq \psi_{ik} \leq \psi_{ik}^{(n)} + \tau_2$, $\forall i, k$, where τ_1 and τ_2 are two constants to control box size.

After relaxing all sum-to-one constraints in (39), under E-approx, LME of MMM can be formulated as the following LP problem.

PROBLEM 6

$$\max_{\Lambda, \theta} \ \theta \ \ (\theta > 0) \quad (43)$$

subject to

$$\phi_{ikd}^{(n)} - \tau_1 \leq \phi_{ikd} \leq \phi_{ikd}^{(n)} + \tau_1 \quad \forall i, k, d, \quad (44)$$

$$\psi_{ik}^{(n)} - \tau_2 \leq \psi_{ik} \leq \psi_{ik}^{(n)} + \tau_2 \quad \forall i, k. \quad (45)$$

$$\sum_i \sum_{k=1}^{K} e_{ik}^{l,\mathrm{j}} \cdot \psi_{ik} + \sum_i \sum_{k=1}^{K} \sum_{d=1}^{D} f_{ikd}^{l,\mathrm{j}} \cdot \phi_{ikd} + g_{ijt}^{l,\mathrm{j}} \geq \theta \quad (46)$$

for all $X_t \in \mathcal{S}$ (with correct model λ_{l_t}) and other models λ_{j} $(\mathrm{j} \neq l_t)$.

Problem 6 can be solved using many general optimization tools. Then, all MMM parameters can be updated with the found solution with the sum-to-one constraints relaxed. If training data are not separable, following the same soft LME method, the above LME formulation can be extended to consider training errors [28]. In [28], the above LME method has been applied to a text categorization task using the standard RCV1 text database and it has achieved much lower classification error rates than other conventional methods, such as SVM and the EM-trained MMMs.

FINAL REMARKS

Discriminative learning methods have achieved many successes in speech and language processing during the past decades. Discriminative learning of generative models is a typical optimization problem, where efficient optimization methods play a critical role. For many widely used statistical models, discriminative learning normally leads to nonconvex optimization problems. In this article we used three representative examples to showcase how to use a proper convex relaxation method to convert discriminative learning of HMMs and MMMs into standard convex optimization problem so that it can be solved effectively and efficiently even for large-scale statistical models. We believe convex optimization will continue to play important role in discriminative learning of other statistical models in other application domains, such as statistical machine translation, computer vision, biometrics, and informatics.

AUTHORS

Hui Jiang (hj@cse.yorku.ca) received the B.Eng. and M.Eng. degrees from the University of Science and Technology of China and his Ph.D. degree from the University of Tokyo, Japan, all in electrical engineering. From 1999 to 2000, he was with the Department of Electrical and Computer Engineering, University of Waterloo, Canada, as a postdoctoral fellow. From 2000 to 2002, he worked at Lucent Technologies Inc., Murray Hill, New Jersey. He joined the Department of Computer Science and Engineering, York University, Toronto, Canada, as assistant professor in 2002 and was promoted to associate professor in 2007. His current research interests include speech/audio/language processing, machine learning, and statistical data modeling. He is a Member of the IEEE and has served as an associate editor for *IEEE Transactions on Audio, Speech, and Language Processing* since 2009.

Xinwei Li (xwli@cse.yorku.ca) received his M.S. degree in computer science from York University, Canada, and his B.S. degree in electronics from Beijing University, China. He is a speech scientist with Nuance Inc. His major research interest focuses on ASR, with a focus on DT.

REFERENCES

[1] Y. Altun, I. Tsochantaridis, and T. Hofmann, "Hidden Markov support vector machines," in *Proc. Int. Conf. Machine Learning (ICML'03)*, Washington, DC, 2003.

[2] L. R. Bahl, P. F. Brown, P. V. De Souza, and R. L. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'86)*, Tokyo, Japan, 1986, pp. 49–52.

[3] P. Biswas and Y. Ye, "Semidefinite programming for ad hoc wireless sensor network localization," in *Proc. Information Processing in Sensor Networks (IPSN'04)*, Berkeley, Apr. 2004, pp. 46-54.

[4] L. D. Brown, *Fundamentals of Statistical Exponential Families: With Applications in Statistical Decision Theory*. Hayward, CA: Institute of Mathematical Statistics, 1986.

[5] K. Crammer and Y. Singer, "On the algorithmic implementation of multi-class kernel-based vector machines," *J. Mach. Learn. Res.* (Special Issue on Kernel Methods), vol. 2, pp. 265–292, 2001.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm (with discussion)," *J. Royal Stat. Soc. B*, vol. 39, no. 1, pp. 1–38, 1977.

[7] P. S. Gopalakrishnan, D. Kanevsky, A. Nadas, and D. Nahamoo, "An inequality for rational functions with applications to some statistical estimation problems," *IEEE Trans. Information Theory*, vol. 37, no. 1, pp. 107–113, Jan. 1991.

[8] X. He, L. Deng, and W. Chou, "Discriminative learning in sequential pattern recognition: A unifying review for optimization-based speech recognition," *IEEE Signal Processing Mag.*, Sept. 2008, vol. 25, no. 5, pp. 14–36.

[9] T. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," in *Proc. Neural Information Processing Systems (NIPS'98)*, no. 11, 1998, pp. 487–493.

[10] T. Jaakkola, M. Meila, and T. Jebara, "Maximum entropy discrimination," in *Proc. Neural Information Processing Systems (NIPS'99)*, no. 12, 1999, pp. 470–476.

[11] T. Jebara and A. Pentland, "Maximum conditional likelihood via bound maximization and the CEM algorithm," in *Proc. Neural Information Processing Systems (NIPS'98)*, 1998, pp. 494–500.

[12] T. Jebara, "Discriminative, generative and imitative learning," Ph.D. dissertation, MIT Press, Cambridge, MA, Feb. 2002.

[13] H. Jiang, X. Li, and C.-J. Liu, "Large margin hidden Markov models for speech recognition," *IEEE Trans. Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1584–1595, 2006.

[14] H. Jiang and X. Li, "Incorporating training errors for large margin HMMs under semi-definite programming framework," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'07)*, Hawaii, USA, 2007, vol. 4, pp. 629–632.

[15] H. Jiang and X. Li, "A general approximation-optimization approach to large margin estimation of HMMs," in *Speech Recognition and Synthesis*, V. Kodic, Ed., Vienna, Austria: I-Tech Education and Publishing, May 2007, pp 103–120.

[16] H. Jiang, "A general formulation for discriminative learning of graphical models," Dept. of Computer Science and Engineering, York Univ., Toronto, Canada, Tech. Rep., Mar. 2007.

[17] H. Jiang, "Discriminative training of HMMs for automatic speech recognition: A survey," *Comput. Speech Lang.*, to be published.

[18] B.-H. Juang, W. Chou, and C.-H. Lee, "Minimum classification error rate methods for speech recognition," *IEEE Trans. Speech and Audio Processing*, vol. 5, no. 3, pp.257–265, 1997.

[19] X. Li and H. Jiang, "Solving large margin hidden Markov model estimation via semidefinite programming," *IEEE Trans. Audio, Speech and Language Processing*, vol. 15, no. 8, pp. 2383–2392, Nov. 2007.

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

[21] P. Liu, C. Liu, H. Jiang, F. Soong, and R.-H. Wang, "A constrained line search optimization method for discriminative training of HMMs," *IEEE Trans. Audio, Speech and Language Processing*, vol. 16, no. 5, pp. 900–909, July 2008.

[22] C. Liu, Y. Hu, H. Jiang, and L.-R. Dai, "A bounded trust region optimization for discriminative training of HMMs in speech recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'10)*, Dallas, TX, Mar. 2010.

[23] Z.-Q. Luo and W. Yu, "An introduction to convex optimization for communications and signal processing," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1426–1438, Aug. 2006.

[24] Z.-Q. Luo, W.-K. Ma, A. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization probllems," *IEEE Signal Processing Mag.* (Special Issue on Convex Optimization in Signal Processing), vol. 27, no. 3, pp. 20–34, May 2010.

[25] V. Magdin and H. Jiang, "Discriminative training of n-gram language models for speech recognition via linear programming," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU'09)*, Italy, Dec. 2009, pp. 305–310.

[26] E. McDermott, T. Hazen, J. Le Roux, A. Nakamura, and S. Katagiri, "Discriminative training for large-vocabulary speech recognition using minimum classification error," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 203–223, 2007.

[27] S. Ortmanns, H. Ney, and X. Aubert, "A word graph algorithm for large vocabulary continuous speech recognition," *Comput. Speech Lang.*, vol. 11, no. 1, pp. 43–72, 1997.

[28] Z. Pan and H. Jiang, "Large margin multinomial mixture model for text categorization," in *Proc. Interspeech*, Brisbane, Australia, Sept. 2008, pp. 1566–1569.

[29] D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'02)*, Orlando, FL, 2002, vol. 1, pp. 105–108.

[30] F. Sha and L. K. Saul, "Large margin Gaussian mixture modeling for phonetic classification and recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'06)*, Toulouse, France, 2006, pp. 265–268.

[31] F. Sha and L. K. Saul, "Large margin hidden Markov models for automatic speech recognition," in *Proc. Neural Information Processing Systems (NIPS'07)*, 2007, pp. 1249–1256.

[32] B. Taskar, C. Guestrin, and D. Koller, "Max-margin Markov networks," in *Proc. Neural Information Processing Systems (NIPS'03)*, Vancouver, Canada, 2003.

[33] F. Wessel, R. Schluter, K. Macherey, and H. Ney, "Confidence measures for large vocabulary continuous speech recognition," *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 3, pp. 288–298, 2001.

[34] J. Weston and C. Watkins, "Support vector machines for multi-class pattern recognition," in *Proc. of European Symp. on Artificial Neural Networks*, 1999.

[35] P. C. Woodland and D. Povey, "Large scale discriminative training of hidden Markov models for speech recognition," *Comput. Speech Lang.*, vol. 16, no. 1, pp.25–47, Jan. 2002.

[36] D. Wu, B. Li, and H. Jiang, "Maximum mutual information estimation via second order cone programming for large vocabulary continuous speech recognition," in *Proc. Interspeech*, Brighton, U.K., Apr. 2009, pp. 672–675.

[37] Z. Yan, C. Liu, Y. Hu, and H. Jiang, "A trust region based optimization for maximum mutual information estimation of HMMs in speech recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'09)*, Taipei, Taiwan, Apr. 2009, pp. 3757–3760.

[38] Y. Yin and H. Jiang, "A fast optimization method for large margin estimation of HMMs based on second order cone programming," in *Proc. Interspeech*, Sept. 2007, pp. 34–37.

[39] Y. Yin and H. Jiang, "A compact semidefinite programming (SDP) formulation for large margin estimation of HMMs in speech recognition," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU'07)*, 2007, pp. 312–317.

[40] D. Yu, L. Deng, X. He, and A. Acero, "Large-margin minimum classification error training for large-scale speech recognition tasks," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'07)*, Hawaii, USA, 200, vol. 4, pp. 1137–1140.

[41] V. N. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998. [SP]