Xiaodong He, Li Deng, and Wu Chou

© PHOTODISC & DIGITAL VISION

A unifying review for optimization-oriented speech recognition

Discriminative Learning in Sequential Pattern Recognition

Discriminative learning has become a major theme in recent statistical signal processing and pattern recognition research including practically all areas of speech and language processing, [9], [10], [13], [21], [29], [35], [43], [44], [47], [49]. In particular, much of the striking progress in large-scale automatic speech recognition over the past few years has been attributed to the successful development and applications of discriminative learning [35], [38], [47], [48]. A key to understanding the speech process is the dynamic characterization of its sequential- or variable-length pattern. Two central issues in the development of discriminative learning methods for sequential pattern recognition are construction of the objective function for optimization and actual optimization techniques. There have been a wide variety of methods reported in the literature related to both of these issues [9], [18], [21], [29], [35],

Digital Object Identifier 10.1109/MSP.2008.926652

1053-5888/08/$25.00©2008IEEE

[38], [41], [45], [49], [52], [54], [58]. However, their relationships have not been adequately understood. Because of the practical and theoretical importance of this problem, there is a pressing need for a unified account of the numerous discriminative learning techniques in the literature. This article is aimed at fulfilling this need while providing insight into the discriminative learning framework for sequential pattern classification and recognition. We intend to address the issue of how various discriminative learning techniques are related to and distinguished from each other and what may be a deeper underlying scheme that can unify various ostensibly different techniques. Although the unifying review provided in this article is on a general class of pattern recognition problems associated with sequential characteristics, we will focus most of the discussions on those related to speech recognition and to the hidden Markov model (HMM) [11], [50], [56]. We note that the HMM as well as the various forms of discriminative learning have been used in many signal processing-related areas beyond speech, e.g., in bioinformatics [6], [16], computational genetics [55], text and image classification/recognition [33], [62], [65], video object classification [63], natural language processing [8], [10], and telerobotics [64]. It is our hope that the unifying review and insight provided in this article will foster more principled and successful applications of discriminative learning in a wide range of signal processing disciplines, speech processing or otherwise.

MMI, MCE, AND MPE/MWE ARE THE THREE MOST POPULAR DISCRIMINATIVE LEARNING CRITERIA IN SPEECH AND LANGUAGE PROCESSING.

In addition to presenting an extensive account of the basic ideas behind approaches and methods in discriminative learning, we also wish to position our treatment of related algorithms in a wider context of learning and building statistical classifiers from a more general context of machine learning. Generative and discriminative approaches are two main paradigms for designing and learning statistical classifiers/recognizers. Generative recognizers rely on a learned model of the joint probability distribution of the observed features and the corresponding class membership. They use this joint-probability model to perform the decision-making task based on the posterior probability of the class computed by Bayes rule [12], [50], [66]. In contrast, discriminative classifiers/recognizers directly employ the class posterior probability (or the related discriminant function), exemplified by the argument that "one should solve the (classification/recognition) problem directly and never solve a more general problem as an intermediate step" [57]. This recognizer design philosophy is the basis of a wide range of popular machine learning methods including support vector machine [57], conditional random field [32], [44], and maximum entropy Markov models [19], [34], where the "intermediate step" of estimating the joint distribution has been avoided. For example, in the recently proposed structured classification approach [19], [32], [34], [44] in machine learning and speech recognition, some well-known deficiencies of the HMM are addressed by applying direct discriminative learning, replacing the need for a probabilistic generative model by a set of flexibly selected, overlapping features. Since the conditioning is made on the feature sequence and these features can be designed with long-contextual-span properties, the conditional-independence assumption made in the HMM is conceptually alleviated, provided that proper features can be constructed. How to design such features is a challenging research direction, and it becomes a critical factor for the potential success of the structured discriminative approach, which departs from the generative component or joint distribution. On the other hand, local features can be much more easily designed that are appropriate for the generative approach, and many effective local features have been established (e.g., cepstra, filter-bank outputs, etc. [11], [50] for speech recognition). Despite the complexity of estimating joint distributions when the sole purpose is discrimination, the generative approach has important advantages of facilitating knowledge incorporation and conceptually straightforward analyses of recognizer's components and their interactions.

Analyses of the capabilities and limitations associated with the two general machine-learning paradigms discussed above lead to a practical pattern recognition framework being pursued here. That is, we attempt to establish a simplistic joint-distribution or generative model with the complexity lower than what is required to accurately generate samples from the true distribution. In order to make such low-complexity generative models discriminate well, it requires parameter learning methods that are discriminative in nature to overcome the limitation in the simplistic model structure. This is in contrast to the generative approach of fitting the intraclass data as conventional maximum likelihood (ML)-based methods intend to accomplish. This type of practical framework has been applied to and is guiding much of the recent work in speech recognition research, where HMMs are used as the low-complexity joint distribution for the local acoustic feature sequences of speech and the corresponding underlying linguistic label sequences (sentences, words, or phones). Popular discriminative parameter learning techniques for HMMs are 1) maximum mutual information (MMI) [7], [18], [20], [39], [40], [41], [58],

[61]; 2) minimum classification error (MCE) [1], [9], [23], [28], [29], [35], [36], [38], [49], [52], [54]; and 3) minimum phone error (MPE) and closely related minimum word error (MWE) [13], [45]–[48].

In addition to providing a general overview on the above classes of techniques, this article has a special focus on three key areas in discriminative learning. First, it provides a unifying view of the three major discriminative learning objective functions (MMI, MCE, and MPE/MWE) for classifier parameter optimization, from which insights to the relationships among them are derived. We concentrate on a unified objective function that gives rise to various special cases associated with different levels of performance optimization for pattern recognition tasks, including performance optimization levels of super-string unit, string unit, and substring unit. Second, we describe an efficient approach of parameter estimation in classifier design that unifies the optimization techniques for discriminative learning. This approach for parameter estimation in discriminative learning is based on the optimization framework of growth transformation (GT) (see its detailed introduction in the section "Optimizing Rational Functions by GT"). We show that this approach leads to unified parameter estimation formulas and that it is scalable for large pattern recognition tasks. The third area is the algorithmic properties of the MCE and MPE/MWE-based learning methods under the parameter estimation framework of growth transformation for sequential pattern recognition using HMMs.

DISCRIMINATIVE LEARNING CRITERIA OF MMI, MCE, AND MPE/MWE

MMI, MCE, and MPE/MWE are the three most popular discriminative learning criteria in speech and language processing. Although the discussion of the discriminative classifier design in this article has a focus on speech and language processing, they are equally applicable to other similar sequential pattern recognition problems such as handwriting recognition. References made in this article to words, phones, strings, etc., are for the purpose of showing that the sequential dynamic pattern recognition problem can be based on different levels of recognition units. Moreover, the classifier in sequential pattern recognition can be constructed based on recognizing each pattern (or recognition unit) in isolation. If it can take advantage of the sequential correlation, the classifier can also be constructed based on recognizing a string of patterns (or a string of recognition units), e.g., phrases, word strings, sentences. This flexibility in classifier design for sequential pattern recognition has been a fertile field of research, and many approaches have been developed [22], [29], [47].

To set the stage, we denote by Λ the set of classifier parameters that needs to be estimated during the classifier design. For instance, in speech and language processing, a (generative) joint distribution of observing a data sequence X given the corresponding labeled word sequence S can be written as follows:

$$p(X, S\,|\,\Lambda) = p(X\,|S, \Lambda)\; P(S). \tag{1}$$

In this notation, it is assumed that the parameters in the "language model" $P(S)$ are not subject to optimization. Given a set of training data, we denote by R the total number of training tokens. We focus on supervised learning, where each training token consists of an observation data sequence: $X_r = x_{r,1}, \ldots, x_{r,T_r}$, and its correctly labeled (e.g., word) pattern sequence: $S_r = W_{r,1}, \ldots, W_{r,N_r}$, with $W_{r,i}$ being the ith word in word sequence S_r. We use a lowercase variable s_r to denote all possible pattern sequences that can be used to label the rth token, including the correctly labeled sequence S_r and other sequences.

MMI

In the MMI-based classifier design, the goal of classifier parameter estimation is to maximize the mutual information $I(X, S)$ between data X and their corresponding labels/symbols S. From the information theory perspective, mutual information provides a measure of the amount of information gained, or the amount of uncertainty reduced, regarding S after seeing X. The MMI criterion is well established in information theory. It possesses good theoretical properties, and it is different from the criterion of ML used in generative model-based learning. Quantitatively, mutual information $I(X, S)$ is defined as

$$\begin{aligned} I(X, S) &= \sum_{X,S} p(X, S) \log \frac{p(X, S)}{p(X)p(S)} \\ &= \sum_{X,S} p(X, S) \log \frac{p(S\,|X)}{p(S)} = H(S) - H(S\,|X), \end{aligned} \tag{2}$$

where $H(S) = -\sum_S p(S) \log p(S)$ is the entropy of S and $H(S\,|X)$ is the conditional entropy given data $X : H(S\,|X) = -\sum_{X,S} p(X, S) \log p(S\,|X)$. When $p(S|X)$ is based on model Λ, we have

$$H(S\,|X) = -\sum_{X,S} p(X, S) \log p(S\,|X, \Lambda). \tag{3}$$

Assume that the parameters in $P(S)$ (language model) and hence $H(S)$ is not subject to optimization. Consequently, maximizing mutual information of (2) becomes equivalent to minimizing $H(S\,|X)$ of (3) on the training data. When the tokens in the training data are drawn from an independent and identically-distributed (i.i.d.) distribution, $H(S\,|X)$ is given by

$$\begin{aligned} H(S\,|X) &= -\frac{1}{R} \sum_{r=1}^{R} \log p(S_r\,|X_r, \Lambda) \\ &= -\frac{1}{R} \sum_{r=1}^{R} \log \frac{p(X_r, S_r\,|\Lambda)}{p(X_r|\Lambda)}. \end{aligned}$$

Therefore, parameter optimization of MMI-based discriminative learning is to maximize the following objective function:

$$O_{\mathrm{MMI}}(\Lambda) = \sum_{r=1}^{R} \log \frac{p(X_r, S_r|\Lambda)}{P(X_r|\Lambda)} = \sum_{r=1}^{R} \log \frac{p(X_r, S_r|\Lambda)}{\sum_{s_r} p(X_r, s_r|\Lambda)}, \quad (4)$$

where $P(s_r)$ is the language model probability of pattern sequence s_r.

The objective function O_{MMI} of (4) is a sum of logarithms. For comparisons with other discriminative training criteria in following sections, we construct the monotonically increasing function of exponentiation for (4). This gives

$$\tilde{O}_{\mathrm{MMI}}(\Lambda) = \exp[O_{\mathrm{MMI}}(\Lambda)] = \prod_{r=1}^{R} \frac{p(X_r, S_r|\Lambda)}{\sum_{s_r} p(X_r, s_r|\Lambda)}. \quad (5)$$

It should be noted that $\tilde{O}_{\mathrm{MMI}}$ and O_{MMI} have the same set of maximum points, because maximum points are invariant to monotonically increasing transforms. For comparisons with other discriminative training criteria, we rewrite each factor in (5) as

$$\frac{p(X_r, S_r|\Lambda)}{\sum_{s_r} p(X_r, s_r|\Lambda)} = 1 - \sum_{s_r \neq S_r} p(s_r|X_r, \Lambda) = 1 - \overbrace{\sum_{s_r} \underbrace{(1-\delta(s_r, S_r))}_{\text{0 - 1 loss}} p(s_r|X_r, \Lambda))}^{\text{model based expected loss}}. \quad (6)$$

We define (6) as the model-based expected utility for token X_r, which equals one minus the model-based expected loss for that token.

MCE

The MCE-based classifier design is a discriminant-function-based approach to pattern recognition [1], [28], [29]. The decision rule of the classifier is treated as comparisons among a set of discriminant functions, and parameter estimation involves minimizing the expected loss incurred when these decision rules are applied to the classifier. The loss function in MCE-based discriminative learning is constructed in such a way that the recognition error rate of the classifier is embedded in a smooth functional form, and minimizing the expected loss of the classifier has a direct relation to classifier error rate reduction.

The objective (loss) function in MCE-based discriminative learning can be constructed from likelihood-based generative models through the following steps. For each training token X_r, the set of discriminant functions $\{g_{s_r}\}$ is given as

$$g_{s_r}(X_r; \Lambda) = \log p(X_r, s_r|\Lambda),$$

which is the log joint probability of data X_r and the pattern sequence (string) s_r given model Λ. The decision rule of the classifier/recognizer is defined as

$$C(X_r) = s_r^* \qquad \text{iff } s_r^* = \arg\max_{s_r} g_{s_r}(X_r; \Lambda).$$

In practice, the N most confusable competing strings, $s_{r,1}, \ldots, s_{r,N}$, against the correct string S_r are considered in MCE-based discriminative learning, where each of these N-best strings can be defined inductively by

$$s_{r,1} = \arg\max_{s_r: s_r \neq S_r} \log p(X_r, s_r|\Lambda)$$
$$s_{r,i} = \underset{s_r: s_r \neq S_r, s_r \neq s_{r,1}, \ldots, s_{r,i-1}}{\arg\max} \log p(X_r, s_r|\Lambda), \quad i = 2, \ldots, N, \quad (7)$$

and Λ is the current model parameter set of the classifier. Then, a misclassification measure $d_r(X_r, \Lambda)$ can be defined to approximate the performance of the decision rule for each training token X_r, i.e., $d_r(X_r, \Lambda) \geq 0$ implies misclassification and $d_r(X_r, \Lambda) < 0$ implies correct classification. In particular, such a misclassification measure can be defined by

$$d_r(X_r, \Lambda) = -g_{S_r}(X_r; \Lambda) + G_{S_r}(X_r; \Lambda), \quad (8)$$

where $G_{S_r}(X_r; \Lambda)$ is a function that represents the scores from the incorrect competing strings and $g_{S_r}(X_r; \Lambda)$ is the discriminant function for the correct string S_r.

In the case of one-best string MCE approach ($N = 1$), only the most confusable incorrect string $s_{r,1}$ is considered as the competitor where $G_{S_r}(X_r; \Lambda)$ becomes

$$G_{S_r}(X_r; \Lambda) = g_{s_{r,1}}(X_r; \Lambda). \quad (9)$$

However, for the general case where $N > 1$, different definitions of $G_{S_r}(X_r; \Lambda)$ can be used. One popular definition takes the following form [29]:

$$G_{S_r}(X_r; \Lambda) = \log \left\{ \frac{1}{N} \sum_{i=1}^{N} p^{\eta}(X_r, s_{r,i}|\Lambda) \right\}^{\frac{1}{\eta}}. \quad (10)$$

Another popular form of $g_{S_r}(X_r; \Lambda)$ and $G_{S_r}(X_r; \Lambda)$ (the latter has similar effects to (10) and was used in [54]) is

$$\begin{cases} g_{S_r}(X_r; \Lambda) = \log p^{\eta}(X_r, S_r|\Lambda) \\ G_{S_r}(X_r; \Lambda) = \log \sum_{i=1}^{N} p^{\eta}(X_r, s_{r,i}|\Lambda), \end{cases} \quad (11)$$

where η is a scaling factor for joint probability $p(X_r, s_r|\Lambda)$. In this article, we adopt $G_{S_r}(X_r; \Lambda)$ with the form of (11) and set $\eta = 1$ for mathematical tractability reasons. (The $\eta \neq 1$ case will be discussed in "Two Empirical Issues in MCE Implementation.")

Given the misclassification measure, the loss function can be defined for each training token r, and it is usually defined through a sigmoid function as originally proposed in [28], [29]:

$$l_r(d_r(X_r, \Lambda)) = \frac{1}{1 + e^{-\alpha d_r(X_r, \Lambda)}}, \tag{12}$$

where $\alpha > 0$ is the slope of the sigmoid function, often determined empirically. As presented in [25, p. 156], we also use $\alpha = 1$ for simplifications in the exposition of this article. (More discussions of α in empirical studies are included in "Two Empirical Issues in MCE Implementation"). It should be noted that the loss function of (12) approximates the zero-one classification error count in a smooth functional form.

Given the set of all possible pattern sequences $\{s_r\} = \{S_r, s_{r,1}, \ldots, s_{r,N}\}$ associated with observation data X_r, and with $\eta = 1$ and $\alpha = 1$, we substitute (11) into (12) and rewrite the loss function for the training token X_r as

$$\begin{aligned} l_r(d_r(X_r, \Lambda)) &= \frac{\sum\limits_{s_r, s_r \neq S_r} p(X_r, s_r|\Lambda)}{\sum\limits_{s_r, s_r \neq S_r} p(X_r, s_r|\Lambda) + p(X_r, S_r|\Lambda)} \\ &= \frac{\sum\limits_{s_r, s_r \neq S_r} p(X_r, s_r|\Lambda)}{\sum\limits_{s_r} p(X_r, s_r|\Lambda)}. \end{aligned} \tag{13}$$

Correspondingly, we can define the utility function as one minus the loss function, i.e.,

$$u_r(d_r(X_r, \Lambda)) = 1 - l_r(d_r(X_r, \Lambda)). \tag{14}$$

The goal in the MCE-based discriminative learning becomes minimization of the expected loss over the entire training data

$$L_{\mathrm{MCE}}(\Lambda) = \frac{1}{R} \sum_{r=1}^{R} l_r(d_r(X_r, \Lambda)). \tag{15}$$

Obviously, minimizing $L_{\mathrm{MCE}}(\Lambda)$ in (15) is equivalent to maximizing the following MCE objective function:

$$\begin{aligned} O_{\mathrm{MCE}}(\Lambda) &= R(1 - L_{\mathrm{MCE}}(\Lambda)) \\ &= \sum_{r=1}^{R} u_r(d_r(X_r, \Lambda)) \\ &= \sum_{r=1}^{R} \frac{p(X_r, S_r|\Lambda)}{\sum_{s_r} p(X_r, s_r|\Lambda)}. \end{aligned} \tag{16}$$

It is noteworthy that the summation in (16) for combining utilities of all string tokens for MCE forms a sharp contrast to the MMI case as in (5) where a multiplication of utility functions is constructed for pooling all string tokens.

MPE/MWE

MPE/MWE is another approach to discriminative learning. It was originally developed in [45] and [47] and has demonstrated quite effective performance improvement in speech recognition. In contrast to MMI and MCE, which are typically aimed at large segments of pattern sequences (e.g., at string or even super-string level obtained by concatenating multiple pattern strings in sequence), MPE aims at the performance optimization at the substring pattern level. In speech recognition, a pattern string usually corresponds to a sentence which consists of a sequence of words, and a substring as a constituent of the sentence can be words or phones (subwords).

The MPE objective function that needs to be maximized is defined as

$$O_{\mathrm{MPE}}(\Lambda) = \sum_{r=1}^{R} \frac{\sum_{s_r} p(X_r, s_r|\Lambda)\ A(s_r, S_r)}{\sum_{s_r} p(X_r, s_r|\Lambda),} \tag{17}$$

where $A(s_r, S_r)$ is the raw phone (substring) accuracy count in the sentence string s_r (proposed originally in [45] and [47]). The raw phone accuracy count $A(s_r, S_r)$ is defined as the total phone (substring) count in the reference string S_r minus the sum of insertion, deletion, and substitution errors of s_r computed based on S_r.

The MPE criterion (17) equals the model-based expectation of the raw phone accuracy count over the entire training set. This relation can be seen more clearly by rewriting (17) as

$$O_{\mathrm{MPE}}(\Lambda) = \sum_{r=1}^{R} \sum_{s_r} P(s_r|X_r, \Lambda) A(s_r, S_r)$$

where

$$p(s_r|X_r, \Lambda) = \frac{p(X_r, s_r|\Lambda)}{p(X_r|\Lambda)} = \frac{p(X_r, s_r|\Lambda)}{\sum_{s_r} p(X_r, s_r|\Lambda)}$$

is the model-based posterior probability.

The concept of raw phone accuracy count $A(s_r, S_r)$ in (17) can be generalized to define raw substring accuracy count. In particular, raw word accuracy count $A_w(s_r, S_r)$ can be defined in the same fashion as the total word (substring) count in the reference string S_r minus the sum of insertion, deletion and substitution errors of s_r computed based on S_r. Based on raw word accuracy count $A_w(s_r, S_r)$, we have the equivalent definition of the MWE criterion

$$O_{\mathrm{MWE}}(\Lambda) = \sum_{r=1}^{R} \frac{\sum_{s_r} p(X_r, s_r|\Lambda)\ A_w(s_r, S_r)}{\sum_{s_r} p(X_r, s_r|\Lambda)}, \tag{18}$$

and therefore, in this article, we merge these two approaches into one MPE/MWE category.

KEY DIFFERENCES

At the single-token level, the MMI criterion uses a model-based expected utility of (6) while the MCE criterion uses an classifier-dependent smoothed empirical utility defined by (8), (12), and (14). The MPE/MWE criterion also uses a model-based expected utility, but the utility is computed at the substring level, e.g., at the phone or word level. In this article, we note that for mathematical tractability reasons, a specific misclassification measure (11) is used for MCE. As a consequence, the smoothed empirical utility (14) takes the same form as (6) (though they are derived from different motivations). This can be directly seen by substituting (13) for (14).

At the multiple-token level, by comparing (5), (16), (17), and (18), it is clear that MMI training maximizes a product of model-based expected utilities of training tokens, while MCE training maximizes a summation of smoothed empirical utilities over all training tokens and MPE/MWE training maximizes a summation of model-based expected utilities (computed on substring units). The difference between the product and the summation forms of the utilities differentiates MMI from MCE/MPE/MWE. This difference causes difficulties in extending the original GT/extended Baum-Welch (EBW) formulas proposed for MMI to other criteria [47 p. 92]. In the following sections, we will show how this difference is reflected in our unified criterion.

GT-BASED PARAMETER OPTIMIZATION REFERS TO A FAMILY OF BATCH-MODE, ITERATIVE OPTIMIZATION SCHEMES THAT GROW THE VALUE OF THE OBJECTIVE FUNCTION UPON EACH ITERATION.

THE COMMON RATIONAL-FUNCTION FORM FOR OBJECTIVE FUNCTIONS OF MMI, MCE, AND MPE/MWE

In this section, we show that the objective functions in discriminative learning based on the MMI, MCE, and MPE/MWE criteria can be mapped to a canonical rational-function form where the denominator function is constrained to be positive valued. This canonical rational-function form has the benefit of offering insights into the relationships among MMI, MCE, and MPE/MWE-based classifiers. In addition, it facilitates the development of a unified classifier parameter optimization framework for applying MMI, MCE, and MPE/MWE objective functions in sequential pattern recognition tasks.

RATIONAL-FUNCTION FORM FOR THE OBJECTIVE FUNCTION OF MMI

Based on (5), the canonical rational-function form for MMI objective function can be constructed as

$$\tilde{O}_{\mathrm{MMI}}(\Lambda) = \frac{p(X_1 \ldots X_R, S_1 \ldots S_R \mid \Lambda)}{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda)} = \frac{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda)\, C_{\mathrm{MMI}}(s_1 \ldots s_R)}{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda)}, \tag{19}$$

where

$$C_{\mathrm{MMI}}(s_1 \ldots s_R) = \prod_{r=1}^{R} \delta(s_r, S_r) \tag{20}$$

is a quantity that depends only on the sentence sequence $s_1, \ldots, s_R$ and $\delta(s_r, S_r)$ is the Kronecker delta function, i.e.,

$$\delta(s_r, S_r) = \begin{cases} 1 & \text{if } s_r = S_r \\ 0 & \text{otherwise.} \end{cases}$$

In (19), the first step uses the common assumption that different training tokens are independent of each other.

The MMI objective function is aimed at improving the conditional likelihood on the entire training data set instead of on each individual string (token). It can be viewed as a discriminative performance measure at the super-string level of all training data $s_1, \ldots, s_R$, where $C_{\mathrm{MMI}}(s_1, \ldots, s_R)$ can be interpreted as the binary function (as accuracy count) of the super-string $s_1, \ldots, s_R$, which takes value one if the super-string $s_1, \ldots, s_R$ is correct and zero otherwise.

RATIONAL-FUNCTION FORM FOR THE OBJECTIVE FUNCTION OF MCE

Unlike the MMI case where the rational-function form can be obtained through a simple exponential transformation, the objective function of MCE as given in (16) is a sum of rational functions rather than a rational function in itself (i.e., a ratio of two polynomials). This creates the problem of making the objective function of MCE amenable to the parameter optimization framework of GT. Consequently, the objective function of MCE is usually optimized using the generalized probabilistic descent (GPD) [9], [28], [29] algorithm or other gradient-based methods [37], [38]. Despite the popularity and many successful applications, the gradient descent based sequential learning using GPD has two main drawbacks. First, it is a sample-by-sample learning algorithm. Algorithmically, it is difficult for GPD to parallelize the parameter learning process, which is critical for large scale tasks. Second, it is not a monotone learning algorithm and it does not have a monotone learning function to determine the stopping point of the discriminative learning. Recently, applying other batch-mode gradient-based optimization methods, including batch and semibatch probabilistic descent, Quickprop, and resilient back-propagation (Rprop), to MCE training have been proposed, and improved recognition results are reported [37], [38]. However, monotone convergence of these methods has not been established.

In this article, we take a different approach that makes the objective function for MCE-based discriminative learning directly suitable for GT-based parameter optimization. The scalability and monotone convergence learning properties of GT have the advantage of being fast and stable. In order to realize this advantage, we need to reformulate the MCE objective function and derive a canonical rational-function form for the objective function of MCE. The canonical rational-function form of MCE derived in this process has an additional benefit of unifying the MCE objective function with MMI and MPE/MWE ones, upon which their differences and similarities can be studied.

The derivation of the rational-function form for the objective function of MCE is as follows:

$$
\begin{aligned}
O_{\mathrm{MCE}}(\Lambda) &= \sum_{r=1}^{R} \frac{\sum_{s_r} p(X_r, s_r \mid \Lambda)\delta(s_r, S_r)}{\sum_{s_r} p(X_r, s_r \mid \Lambda)} \qquad (21) \\
&= \underbrace{\frac{\sum_{s_1} p(X_1, s_1 \mid \Lambda)\delta(s_1, S_1)}{\sum_{s_1} p(X_1, s_1 \mid \Lambda)}}_{:=O_1} \\
&\quad + \underbrace{\frac{\sum_{s_2} p(X_2, s_2 \mid \Lambda)\delta(s_2, S_2)}{\sum_{s_2} p(X_2, s_2 \mid \Lambda)}}_{:=O_2} \\
&\quad + \underbrace{\frac{\sum_{s_3} p(X_3, s_3 \mid \Lambda)\delta(s_3, S_3)}{\sum_{s_3} p(X_3, s_3 \mid \Lambda)}}_{:=O_3} \\
&\quad + \cdots + \underbrace{\frac{\sum_{s_R} p(X_R, s_R \mid \Lambda)\delta(s_R, S_R)}{\sum_{s_R} p(X_R, s_R \mid \Lambda)}}_{:=O_R} \\
&= \frac{\sum_{s_1}\sum_{s_2} p(X_1, s_1|\Lambda)p(X_2,s_2|\Lambda)[\delta(s_1,S_1) + \delta(s_2,S_2)]}{\sum_{s_1}\sum_{s_2} p(X_1, s_1|\Lambda)p(X_2, s_2|\Lambda)} \\
&\quad + O_3 + \cdots + O_R \\
&= \frac{\sum_{s_1 s_2} p(X_1, X_2, s_1, s_2 \mid \Lambda)[C_{\mathrm{MCE}}(s_1 s_2)]}{\sum_{s_1 s_2} p(X_1, X_2, s_1, s_2 \mid \Lambda)} \\
&\quad + O_3 + \cdots + O_R \\
&= \frac{\sum_{s_1 s_2 s_3} p(X_1, X_2, X_3, s_1, s_2, s_3 \mid \Lambda)[C_{\mathrm{MCE}}(s_1 s_2 s_3)]}{\sum_{s_1 s_2 s_3} p(X_1, X_2, X_3, s_1, s_2, s_3 \mid \Lambda)} \\
&\quad + O_4 + \cdots + O_R \\
&= \frac{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda)\ C_{\mathrm{MCE}}(s_1 \ldots s_R)}{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda)}, \qquad (22)
\end{aligned}
$$

where $C_{\mathrm{MCE}}(s_1 \ldots s_R) = \sum_{r=1}^{R} \delta(s_r, S_r)$. $C_{\mathrm{MCE}}(s_1, \ldots, s_R)$ can be interpreted as the string accuracy count for $s_1, \ldots, s_R$, which takes an integer value between zero and R as the number of correct strings in $s_1, \ldots, s_R$. The rational-function form (22) for the MCE objective function will play a pivotal role in our study of MCE-based discriminative learning.

RATIONAL-FUNCTION FORM FOR THE OBJECTIVE FUNCTIONS OF MPE/MWE

Similar to MCE, the MPE/MWE objective function is also a sum of multiple (instead of a single) rational functions, and hence it is difficult to derive GT formulas as discussed in [47, p. 92]. In order to bypass this issue, a method of optimizing MPE/MWE objective functions based on a heuristic weak-sense auxiliary function (WSAF) was developed in [45] and [47]. We reformulate the MPE/MWE objective function to its equivalent, canonical rational-function form, making the parameter optimization in MPE/MWE-based discriminative learning directly amendable to the GT-based parameter estimation framework. It provides a unified parameter estimation framework with guaranteed monotone convergence properties which are lacking in other alternative methods such as gradient-based and WSAF-based approaches.

An important finding is that the same method used to derive the rational-function form (22) for the MCE objective function can be applied directly to derive the rational-function form for MPE/MWE objective functions as defined in (17) and (18). Note that (17) and (18) are in the same form as (21), except that $\delta(s_r, S_r)$ is replaced by $A(s_r, S_r)$ or $A_w(s_r, S_r)$. The same derivation steps for the objective function of MCE can be applied here and rational-function forms for MPE/MWE are given as follows:

$$
O_{\mathrm{MPE}}(\Lambda) = \frac{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda) C_{\mathrm{MPE}}(s_1 \ldots s_R)}{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda)}, \qquad (23)
$$

where

$$
C_{\mathrm{MPE}}(s_1 \ldots s_R) = \sum_{r=1}^{R} A(s_r, S_r),
$$

and

$$
O_{\mathrm{MWE}}(\Lambda) = \frac{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda) C_{\mathrm{MWE}}(s_1 \ldots s_R)}{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda)}, \qquad (24)
$$

where

$$
C_{\mathrm{MWE}}(s_1 \ldots s_R) = \sum_{r=1}^{R} A_w(s_r, S_r).
$$

$C_{\mathrm{MPE}}(s_1, \ldots, s_R)$ or $C_{\mathrm{MWE}}(s_1, \ldots, s_R)$ can be interpreted as the raw phone or word (substring unit) accuracy count within the super string $s_1, \ldots, s_R$. Its upper-limit value is the total number of phones or words in the full training data (i.e., the correct super-string $S_1, \ldots, S_R$). However, the actual value can become negative, e.g., if there are too many insertion errors. Correspondingly, $O_{\mathrm{MPE}}(\Lambda)$ and $O_{\mathrm{MWE}}(\Lambda)$ can be interpreted as the model-based average raw phone or word accuracy count of the full training data set, respectively.

DISCUSSIONS

The main result in this section is that all three discriminative learning objective functions, MMI, MCE, and MPE/MWE, can be formulated in a unified canonical rational-function form as follows:

$$
O(\Lambda) = \frac{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda) \cdot C_{\mathrm{DT}}(s_1 \ldots s_R)}{\sum_{s_1 \ldots s_R} p(X_1 \ldots X_R, s_1 \ldots s_R \mid \Lambda)}, \qquad (25)
$$

where the summation over $s = s_1 \ldots s_R$ in (25) denotes all possible labeled sequences (both correct and incorrect ones) for all R training tokens. As it will be further elaborated, this huge number of possible strings can be drastically reduced in practical implementations.

In (25), $X_1 \ldots X_R$ denotes the collection of all observation data sequences (strings) in all R training tokens, which we also call a super string after concatenating them into one single string. $p_\Lambda(X_1 \ldots X_R,)(s_1 \ldots s_R)$ is the joint distribution of the super-string data $X_1 \ldots X_R$ and its possible label sequence $s_1 \ldots s_R$. MMI, MCE, and MPE/MWE are differentiated in (25) through the criterion-dependent weighting factors $C_{\mathrm{MMI}}(s_1 \ldots s_R)$, $C_{\mathrm{MCE}}(s_1 \ldots s_R)$, and $C_{\mathrm{MPE}}(s_1 \ldots s_R)$, respectively. An important property is that: $C_{\mathrm{DT}}(s_1 \ldots s_R)$ is dependent only on the labeled sequence $s_1 \ldots s_R$, and it is independent of the parameter set Λ to be optimized.

[TABLE 1] $C_{DT}(s_1 \ldots s_R)$ **IN THE UNIFIED RATIONAL-FUNCTION FORM FOR MMI, MCE, AND MPE/MWE OBJECTIVE FUNCTIONS. THE SET OF "COMPETING TOKEN CANDIDATES" DISTINGUISHES** N**-BEST AND ONE-BEST VERSIONS OF THE MCE. NOTE THAT THE OVERALL** $C_{DT}(s_1 \ldots s_R)$ **IS CONSTRUCTED FROM ITS CONSTITUENTS** $C_{DT}(s_r)$**'S IN INDIVIDUAL STRING TOKENS BY EITHER SUMMATION (FOR MCE, MPE/MWE) OR PRODUCT (FOR MMI).**

OBJECTIVE FUNCTIONS	$C_{DT}(s_r)$	$C_{DT}(s_1 \ldots s_R)$	LABEL SEQUENCE SET USED IN DT
MCE (N-BEST)	$\delta(s_r, S_r)$	$\sum_{r=1}^{R} C_{DT}(s_r)$	$\{S_r, s_{r,1}, \ldots, s_{r,N}\}$
MCE (ONE-BEST)	$\delta(s_r, S_r)$	$\sum_{r=1}^{R} C_{DT}(s_r)$	$\{S_r, s_{r,1}\}$
MPE	$A(s_r, S_r)$	$\sum_{r=1}^{R} C_{DT}(s_r)$	ALL POSSIBLE LABEL SEQUENCES
MWE	$A_w(s_r, S_r)$	$\sum_{r=1}^{R} C_{DT}(s_r)$	ALL POSSIBLE LABEL SEQUENCES
MMI	$\delta(s_r, S_r)$	$\prod_{r=1}^{R} C_{DT}(s_r)$	ALL POSSIBLE LABEL SEQUENCES

The rational-function formulation (25) for MMI, MCE, and MPE/MWE objective functions serves two main purposes. First, it unifies the objective functions for MMI, MCE, and MPE/MWE in a canonical rational-function form upon which the relations among different discriminative learning criteria can be studied and their properties be compared. This provides insights into the various approaches in discriminative learning. Second, the unified objective function (25) overcomes the main obstacle for applying the GT-based parameter optimization framework in discriminative learning. It leads to a scalable and common parameter estimation framework for discriminative learning, which is highly efficient and has well-founded algorithmic convergence properties. All these properties have been among the major concerns in the past when applying discriminative learning to sequential pattern recognition.

As presented in this section, the key difference in the rational-function form of MMI, MCE, and MPE/MWE criteria is the weighting factor in the numerator of (25), where $C_{\mathrm{DT}}(s_1 \ldots s_R)$ as a generic weighting factor depends on what discriminative training (DT) criterion is being applied. For example, for MMI

$$C_{\mathrm{DT}}(s_1 \ldots s_R) = \prod_{r=1}^{R} \delta(s_r, S_r)\,,$$

and for MPE

$$C_{\mathrm{DT}}(s_1 \ldots s_R) = \sum_{r=1}^{R} A(s_r, S_r)\,.$$

In the case of MCE with general N-best competitors where $N > 1$

$$C_{\mathrm{DT}}(s_1 \ldots s_R) = \sum_{r=1}^{R} \delta(s_r, S_r)\,,$$

and for one-best MCE ($N = 1$), s_r belongs to only the subset $\{S_r, s_{r,1}\}$. From the canonical rational-function form (25), direct comparisons can be made on the objective functions of MMI, MCE, and MPE/MWE. Table 1 tabulates the relation among these discriminative objective functions. As discussed in [47], MPE/MWE has an important difference from MCE and MMI in that the weighting given by the MPE/MWE criterion to an incorrect string (sentence token) depends on the number of wrong substrings (e.g., wrong phones or words) within the string. MCE and MMI make a binary distinction based on whether the entire sentence string is correct or not, which may not be a good fit if the goal is to reduce the substring errors (e.g., word errors in speech recognition). This distinction can be clearly seen by comparing the sum of the binary function

$$C_{\mathrm{DT}}(s_1 \ldots s_R) = \sum_{r=1}^{R} \delta(s_r, S_r)$$

for MCE and the sum of nonbinary functions

$$C_{\mathrm{DT}}(s_1 \ldots s_R) = \sum_{r=1}^{R} A(s_r, S_r)$$

for MPE/MWE. This key difference gives rise to the distinction of the substring level versus the string level recognition performance optimization in MPE/MWE and MCE. Further, the product instead of summation form of the binary function associated with MMI, i.e.,

$$C_{\mathrm{DT}}(s_1 \ldots s_R) = \prod_{r=1}^{R} \delta(s_r, S_r)$$

makes it clear that MMI achieves performance optimization at the super-string level, e.g., the joint product of Kronecker delta functions becomes zero if any sentence token is incorrect. Therefore, all summation terms in the numerator of (25) are zero except for the one corresponding to the correct label/transcription sequence. This criterion is apparently less desirable than MCE or MPE/MWE, as has been observed extensively in speech recognition experiments [35], [45]–[47].

Another insight from the unified form of objective function (25) is that in the special case of having only one sentence token (i.e., $R = 1$) in the training data and when the sentence contains only one phone, then all three MMI, MCE, and MPE/MWE criteria become identical. This is obvious because in this case $C_{\mathrm{DT}}(s_1 \ldots s_R)$ becomes identical. The difference surfaces only when the training set consists of multiple sentence tokens. With multiple training tokens, the difference lies mainly in the Λ-independent weighing factor $C_{\mathrm{DT}}(s_1 \ldots s_R)$ (as well as in the set of competitor strings) while the general rational-function form (25) for the three criteria remains unchanged.

Although we intend to derive the GT-based parameter optimization framework for the three types of objective functions of MMI, MCE, and MPE/MWE in sequential pattern recognition, it should be noted that the unified objective function (25) can provide a critical foundation to derive other parameter optimization methods in discriminative learning. For example, recently Jebara [26], [27] proposed a parameter optimization method for rational functions as an alternative to the GT method. This method is based on the reverse Jensen inequality, upon which an elegant solution for HMMs with exponential-family densities is constructed [26].

OPTIMIZING RATIONAL FUNCTIONS BY GT

GT-based parameter optimization refers to a family of batch-mode, iterative optimization schemes that “grow” the value of the objective function upon each iteration. That is, the new set of model parameter Λ is estimated from the current model parameter set Λ' through a transformation $\Lambda = T(\Lambda')$ with the property that the target objective function “grows” in its value $O(\Lambda) > O(\Lambda')$ unless $\Lambda = \Lambda'$. One particular algorithm of this type of optimization techniques is EBW algorithm when HMM parameters are estimated. The GT/EBW algorithm was initially developed for the homogeneous polynomial by Baum and his colleagues [3], [4]. It was later extended to optimizing nonhomogeneous rational functions as reported in [18]. The EBW algorithm became popular for its successful application in MMI-based discriminative training of discrete HMMs [18]. It was later extended and applied to MMI-based discriminative training of continuous-density HMMs (CDHMMs) [2], [20], [41], [59], [61].

The importance of the GT/EBW algorithm lies in its monotone convergence properties, its algorithmic effectiveness and scalability for parallel execution, and its closed-form parameter updating formulas for large-scale optimization problems. The unified parameter optimization framework of GT also alleviates the need for other heuristics, e.g., tuning the parameter-dependent learning rate as in some other methods [29], [52].

Let $G(\Lambda)$ and $H(\Lambda)$ be two real-valued functions on the parameter set Λ, and the denominator function $H(\Lambda)$ is positive valued. The goal of GT-based parameter optimization is to find an optimal Λ that maximizes the objective function $O(\Lambda)$ which is a rational-function of the following form:

$$O(\Lambda) = \frac{G(\Lambda)}{H(\Lambda)}. \tag{26}$$

For example, $O(\Lambda)$ can be one of the rational-functions of (19), (22), (23) and (24) for the MMI, MCE, and MPE/MWE objective functions, respectively, or the general rational-function (25). In the general case of (25), we have

$$\begin{aligned} G(\Lambda) &= \sum_s p(X, s \mid \Lambda)\, C(s), \quad \text{and} \\ H(\Lambda) &= \sum_s p(X, s \mid \Lambda), \end{aligned} \tag{27}$$

where we use short-hand notation $s = s_1 \ldots s_R$ to denote the labeled sequences of all R training tokens/sentences, and $X = X_1 \ldots X_R$, to denote the observation data sequences for all R training tokens.

PRIMARY AUXILIARY FUNCTION

As originally proposed in [18], for the objective function (26), the GT-based optimization algorithm will construct an auxiliary function of the following form:

$$F(\Lambda; \Lambda') = G(\Lambda) - O(\Lambda')H(\Lambda) + D, \tag{28}$$

where D is a quantity independent of the parameter set, and Λ is the model parameter set to be estimated by applying GT to another model parameter set Λ'. The GT algorithm starts from the (initial) parameter set Λ' (e.g., obtained using ML training). Then, it updates the parameter set from Λ' to Λ by maximizing the auxiliary function $F(\Lambda; \Lambda')$, and the process iterates until convergence is reached. Maximizing the auxiliary function $F(\Lambda; \Lambda')$ can often be more feasible than directly maximizing the original rational-function $O(\Lambda)$. The important property of GT-based parameter optimization is that as long as D is a quantity not relevant to the parameter set Λ, an increase of $F(\Lambda; \Lambda')$ guarantees an increase of $O(\Lambda)$. This can be seen clearly from the following derivation.

Substituting $\Lambda = \Lambda'$ into (27), we have

$$F(\Lambda'; \Lambda') = \underbrace{G(\Lambda') - O(\Lambda')H(\Lambda')}_{=0} + D = D.$$

Hence,

$$\begin{aligned} F(\Lambda; \Lambda') - F(\Lambda'; \Lambda') &= F(\Lambda; \Lambda') - D \\ &= G(\Lambda) - O(\Lambda')H(\Lambda) \\ &= H(\Lambda)\left(\frac{G(\Lambda)}{H(\Lambda)} - O(\Lambda')\right) \\ &= H(\Lambda)(O(\Lambda) - O(\Lambda')). \end{aligned}$$

Since $H(\Lambda)$ is positive, we have $O(\Lambda) - O(\Lambda') > 0$ on the right-hand side, as long as $F(\Lambda; \Lambda') - F(\Lambda'; \Lambda') > 0$ is on the left-hand side.

SECONDARY AUXILIARY FUNCTION

However, $F(\Lambda; \Lambda')$ may still be too difficult to optimize directly, and a secondary auxiliary function can be constructed and optimized based on the previous auxiliary function $F(\Lambda; \Lambda')$. As proposed in [20], this secondary auxiliary function in GT-based parameter estimation can have the following form:

$$V(\Lambda; \Lambda') = \sum_s \sum_q \sum_\chi f(\chi, q, s, \Lambda') \log f(\chi, q, s, \Lambda), \quad (29)$$

where $f(\chi, q, s, \Lambda)$ is a positive valued function which is constructed with discrete arguments of χ, q, s and which is related to the primary auxiliary function $F(\Lambda; \Lambda')$ according to

$$F(\Lambda; \Lambda') = \sum_s \sum_q \sum_\chi f(\chi, q, s, \Lambda). \quad (30)$$

Examples of the arguments χ, q, s are the discrete acoustic observation, the HMM state sequence, and the label sequence, respectively, in a discrete-HMM-based sequential classifier.

By applying the Jensen's inequality to the concave log function, it is easy to prove (proof omitted here) that an increase in the auxiliary function $V(\Lambda; \Lambda')$ guarantees an increase in $\log F(\Lambda; \Lambda')$. Since logarithm is a monotonically increasing function, this implies an increase of $F(\Lambda; \Lambda')$ and hence an increase of the original objective function $O(\Lambda)$.

DISCRIMINATIVE LEARNING FOR DISCRETE HMMs BASED ON THE GT FRAMEWORK

The GT/EBW-based discriminative learning for discrete HMMs needs to estimate the model parameters $\Lambda = \{\{a_{i,j}\}, \{b_i(k)\}\}$ consisting of the state transition and emitting probabilities. We derive the parameter optimization formula that "grows" the generic discriminative objective function $O(\Lambda)$ in the form of (25) which covers MMI, MCE, and MPE/MWE as special cases. The discriminative function $O(\Lambda)$ is difficult to optimize directly. However, since it is a rational function, it is amenable to the GT/EBW-based parameter estimation framework. We can construct the auxiliary function F and then construct the secondary auxiliary function V based on F. We describe how to optimize $V(\Lambda; \Lambda')$, leading to the GT-based parameter estimation formulas for all three types of discriminative criteria: MMI, MCE, and MPE/MWE. This approach is applicable to any other discriminative criteria as long as the objective functions can be represented in a rational-function form of (25).

For the discrete HMM, the observation space is quantized by some discrete codebook. In this case, $X = X_1 \dots X_R$ is a concatenation of all training tokens, and each training token X_r consists of a sequence of discrete indices obtained by mapping the time sequence of observations for rth token to a discrete index sequence with each element $x_{r,t} \in [1, 2, \dots, K]$, where K is the size of the codebook index set and $x_{r,t}$ is the index of the cell that the observation of the tth frame in rth token is quantized to.

CONSTRUCTING THE PRIMARY AUXILIARY FUNCTION $F(\Lambda; \Lambda')$

Substituting (27) into (28), we obtain the following auxiliary function:

$$\begin{aligned} F(\Lambda; \Lambda') &= \sum_s p(X, s|\Lambda)C(s) - O(\Lambda') \sum_s p(X, s|\Lambda) + D \\ &= \sum_s p(X, s|\Lambda)[C(s) - O(\Lambda')] + D \\ &= \sum_s \sum_q p(X, q, s|\Lambda)[C(s) - O(\Lambda')] + D, \end{aligned} \quad (31)$$

where q is an HMM state sequence, and $s = s_1 \dots s_R$ is the "super" label sequence for all R training tokens (including correct or incorrect sentences). The main terms in the auxiliary function $F(\Lambda; \Lambda')$ above can be interpreted as the average deviation of the accuracy count.

CONSTRUCTING THE SECONDARY AUXILIARY FUNCTION $V(\Lambda; \Lambda')$

Since $p(s)$ depends on the language model and is irrelevant for optimizing Λ, we have $p(X, q, s|\Lambda) = p(s,) \cdot p(X, q|s, \Lambda)$, and

$$\begin{aligned} F(\Lambda; \Lambda') &= \sum_s \sum_q [C(s) - O(\Lambda')]p(s)p(X, q|s, \Lambda) + D \\ &= \sum_s \sum_q \sum_\chi [\Gamma(\Lambda') + d(s)]p(\chi, q|s, \Lambda), \end{aligned} \quad (32)$$

where

$$\Gamma(\Lambda') = \delta(\chi, X)p(s)[C(s) - O(\Lambda')] \quad (33)$$

and

$$D = \sum_s d(s)$$

is a quantity independent of parameter set Λ. In (33), $\delta(\chi, X)$ is the Kronecker delta function, where χ represents the entire discrete data space where X belongs. Using ideas in [20], the summation over this data space is introduced here for satisfying the requirement in (28) and (32) that constant D be parameter independent. That is, in (32)

$$\sum_s \sum_q \sum_\chi d(s)p(\chi, q|s, \Lambda) = \sum_s d(s) = D$$

is a Λ-independent constant. While the full sum is Λ-independent, each constituent $d(s)p(\chi, q|s, \Lambda)$ is a Λ-dependent quantity in order to account for the possibility that the corresponding term $\Gamma(\Lambda')p(\chi, q|s, \Lambda)$ may be negative. We elaborate this point below.

To construct the secondary auxiliary function for (29) based on function (32), we first identify from (32) that

TWO CENTRAL ISSUES IN THE DEVELOPMENT OF DISCRIMINATIVE LEARNING METHODS FOR SEQUENTIAL PATTERN RECOGNITION ARE CONSTRUCTION OF THE OBJECTIVE FUNCTION FOR OPTIMIZATION AND ACTUAL OPTIMIZATION TECHNIQUES.

$$f(\chi, q, s, \Lambda) = [\Gamma(\Lambda') + d(s)]p(\chi, q|s, \Lambda)$$

according to (30). To ensure that $f(\chi, q, s, \Lambda)$ above is positive, $d(s)$ should be selected to be sufficiently large so that $\Gamma(\Lambda') + d(s) > 0$ (note $p(\chi, q|s, \Lambda)$ in (32) is nonnegative). Then, using (29), we have

$$\begin{aligned}
V(\Lambda; \Lambda') &= \sum_q \sum_s \sum_\chi [\Gamma(\Lambda') + d(s)]p(\chi, q|s, \Lambda') \\
&\quad \times \log \left\{ \underbrace{[\Gamma(\Lambda') + d(s)]}_{\text{optimization-independent}} p(\chi, q|s, \Lambda) \right\} \\
&= \sum_q \sum_s \sum_\chi [\Gamma(\Lambda') + d(s)]\, p(\chi, q|s, \Lambda') \\
&\quad \times \log p(\chi, q|s, \Lambda) + \text{Const.} \\
&= \sum_q \sum_s p(X, q, s|\Lambda')(C(s) - O(\Lambda')) \log\, p(X, q|s, \Lambda) \\
&\quad + \sum_q \sum_s \sum_\chi d(s)p(\chi, q|s, \Lambda') \\
&\quad \times \log\, p(\chi, q|s, \Lambda) + \text{Const.}
\end{aligned} \tag{34}$$

The auxiliary function (34) is easier to optimize than (32), because the new logarithm $\log p(X, q|s, \Lambda)$ introduced in (34) [which is absent in (32)] can lead to significant simplification of $V(\Lambda; \Lambda')$ which we outline below.

SIMPLIFYING THE SECONDARY AUXILIARY FUNCTION $V(\Lambda; \Lambda')$

We first ignore optimization-independent constant in (34), and divide $V(\Lambda; \Lambda')$ by another optimization-independent quantity, $p(X|\Lambda')$, in order to convert the joint probability $p(X, q, s|\Lambda')$ to the posterior probability $p(q, s|X, \Lambda') = p(s|X, \Lambda')p(q|X, s, \Lambda')$. We then obtain an equivalent auxiliary function of

$$\begin{aligned}
U(\Lambda; \Lambda') &= \sum_q \sum_s p(s|X, \Lambda')p(q|X, s, \Lambda') \\
&\quad \times (C(s) - O(\Lambda')) \log\, p(X, q|s, \Lambda) \\
&\quad + \sum_q \sum_s \sum_\chi d'(s)p(\chi, q|s, \Lambda') \log p(\chi, q|s, \Lambda),
\end{aligned} \tag{35}$$

where

$$d'(s) = d(s)/p(X|\Lambda'). \tag{36}$$

Since X depends only on the HMM state sequence q, we have $p(X, q|s, \Lambda) = p(q|s, \Lambda)\cdot p(X|q, \Lambda)$. Therefore, $U(\Lambda; \Lambda')$ can be further decomposed to four terms as follows:

$$\begin{aligned}
U(\Lambda; \Lambda') = &\overbrace{\sum_q \sum_s p(s|X, \Lambda')p(q|X, s, \Lambda')(C(s) - O(\Lambda')) \log p(X|q, \Lambda)}^{\text{term-I}} \\
&\underbrace{+ \sum_q \sum_s \sum_\chi d'(s)p(\chi, q|s, \Lambda') \log p(\chi|q, \Lambda)}_{\text{term-II}} \\
&+ \overbrace{\sum_q \sum_s p(s|X, \Lambda')p(q|X, s, \Lambda')(C(s) - O(\Lambda')) \log p(q|s, \Lambda)}^{\text{term-III}} \\
&\underbrace{+ \sum_q \sum_s \sum_\chi d'(s)p(\chi, q|s, \Lambda') \log p(q|s, \Lambda)}_{\text{term-IV}}.
\end{aligned} \tag{37}$$

In this case, $X = X_1 \ldots X_R$, aggregates all training data with R independent sentence tokens. For each token $X_r = x_{r,1}, \ldots, x_{r,T_r}$, the observation vector $x_{r,t}$ is independent of each other and it depends only on the HMM state at time t. Hence, $\log p(X|q, \Lambda)$ can be decomposed, enabling simplification of both term-I and term-II in (37). To simplify term-III and term-IV in (37), we decompose $\log p(q|s, \Lambda)$ based on the property of the first-order HMM that state at time t depends only on state at time $t-1$. We now elaborate on the simplification of each of these four terms.

For term-I, we first define

$$\begin{aligned}
\gamma_{i,r,s_r}(t) &= \sum_{q, q_{r,t}=i} p(q|X, s, \Lambda') \\
&= p(q_{r,t} = i|X, s, \Lambda') \\
&= p(q_{r,t} = i|X_r, s_r, \Lambda').
\end{aligned} \tag{38}$$

The last equality comes from the fact that sentence tokens in the training set are independent of each other. $\gamma_{i,r,s_r}(t)$ is the occupation probability of state i at time t, given the label sequence s_r and observation sequence X_r, which can be obtained through an efficient forward-backward algorithm [50]. Using the definition of (38) and assuming the HMM state index is from 1 to I, we have

$$\text{term-I} = \sum_s p(s|X,\Lambda')(C(s)-O(\Lambda'))$$
$$\times \sum_q p(q|X,s,\Lambda') \sum_{r=1}^{R}\sum_{t=1}^{T_r} \log p(x_{r,t}|q_{r,t},\Lambda)$$
$$= \sum_s p(s|X,\Lambda')(C(s)-O(\Lambda'))$$
$$\times \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I}\sum_{q,q_{r,t}=i} p(q|X,s,\Lambda')$$
$$\times \log p(x_{r,t}|q_{r,t}=i,\Lambda)$$
$$= \sum_s p(s|X,\Lambda')(C(s)-O(\Lambda'))$$
$$\times \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I} \gamma_{i,r,s_r}(t)\,\log p(x_{r,t}|q_{r,t}=i,\Lambda). \quad (39)$$

The simplification process for the second term in (37) is can be found in [24], which gives the final result of

$$\text{term-II} = \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I} d(r,t,i)\sum_{\chi_{r,t}} p(\chi_{r,t}|q_{r,t}=i;\Lambda')$$
$$\times \log\, p(\chi_{r,t}|q_{r,t}=i;\Lambda), \quad (40)$$

where

$$d(r,t,i) = \sum_s d'(s)p(q_{r,t}=i|s,\Lambda'). \quad (41)$$

To simplify term-III in (37), we first define

$$\xi_{i,j,r,s_r}(t) = \sum_{q:q_{r,t-1}=i,q_{r,t}=j} p(q|X,s,\Lambda')$$
$$= p(q_{r,t-1}=i,q_{r,t}=j|X,s,\Lambda')$$
$$= p(q_{r,t-1}=i,q_{r,t}=j|X_r,s_r,\Lambda'), \quad (42)$$

which is the posterior probability of staying at state i at time $t-1$ and staying at state j at time t, given the labeled sequence s_r and the observation sequence X_r. This posterior probability can be computed using an efficient forward-backward algorithm [50]. Further, $p(q|s,\Lambda)$ can be decomposed as follows:

$$p(q|s,\Lambda) = \prod_{r=1}^{R} p(q_{r,1},\ldots,q_{r,T_r}|s_r,\Lambda) = \prod_{r=1}^{R}\prod_{t=1}^{T_r} a_{q_{r,t-1},q_{r,t}}.$$

This leads to the following results (see technical detail in [24]):

$$\text{term-III} = \sum_s p(s|X,\Lambda')(C(s)-O(\Lambda'))$$
$$\times \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I}\sum_{j=1}^{I} \xi_{i,j,r,s_r}(t)\,\log a_{i,j} \quad (43)$$

and

$$\text{term-IV} = \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I} d(r,t-1,i)\sum_{j=1}^{I} a'_{i,j}\log a_{i,j} \quad (44)$$

where $a'_{i,j} = p(q_{r,t}=j|q_{r,t-1}=i,s,\Lambda')$ is the transition probability from the previous GT iteration.

Substituting (39), (40), (43) and (44) into (37), and denoting the emitting probability by $b_i(x_{r,t}) = p(x_{r,t}|q_{r,t}=i,\Lambda)$ and $b'_i(x_{r,t}) = p(x_{r,t}|q_{r,t}=i,\Lambda')$, we obtain the decomposed and simplified objective function

$$U(\Lambda;\Lambda') = U_1(\Lambda;\Lambda') + U_2(\Lambda;\Lambda'), \quad (45)$$

where

$$U_1(\Lambda;\Lambda') = \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I}\sum_s p(s|X,\Lambda')(C(s)-O(\Lambda'))$$
$$\times \gamma_{i,r,s_r}(t)\,\log b_i(x_{r,t}) + \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I} d(r,t,i)$$
$$\times \sum_{\chi_{r,t}} b'_i(\chi_{r,t})\,\log\, b_i(\chi_{r,t}) \quad (46)$$

$$U_2(\Lambda;\Lambda') = \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I}\sum_{j=1}^{I}\sum_s p(s|X,\Lambda')(C(s)-O(\Lambda'))$$
$$\times \xi_{i,j,r,s_r}(t)\,\log\, a_{i,j} + \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I} d(r,t-1,i)$$
$$\times \sum_{j=1}^{I} a'_{i,j}\,\log\, a_{i,j}. \quad (47)$$

In (45), $U_1(\Lambda;\Lambda')$ is relevant only to optimizing the emitting probability $b_i(k)$, and $U_2(\Lambda;\Lambda')$ is relevant only to optimizing the transition probability $a_{i,j}$.

ESTABLISHING GT BY OPTIMIZING THE AUXILIARY FUNCTION $U(\Lambda;\Lambda')$

In order to optimize the discrete distribution $b_i(k) = p(x_{r,t}=k|q_{r,t}=i,\Lambda)$, $k=1,2,\ldots,K$, where the constraint $\sum_{k=1}^{K} b_i(k)=1$ is imposed, we apply the Lagrange multiplier method by constructing

$$W_1(\Lambda;\Lambda') = U_1(\Lambda;\Lambda') + \sum_{i=1}^{I}\lambda_i\left(\sum_{k=1}^{K} b_i(k)-1\right). \quad (48)$$

Setting $\partial W_1(\Lambda;\Lambda')/\partial\lambda_i = 0$ and $\partial W_1(\Lambda;\Lambda')/\partial b_i(k) = 0$, $k=1,\ldots,K$, we have the following $K+1$ equations

$$\sum_{k=1}^{K} b_i(k) - 1 = 0$$

$$0 = \lambda_i b_i(k) + \sum_{r=1}^{R} \sum_{\substack{t=1 \\ s.t.\ x_{r,t}=k}}^{T_r} \times \overbrace{\sum_s p(s|X, \Lambda')(C(s) - O(\Lambda'))\gamma_{i,r,s_r}(t)}^{\Delta\gamma(i,r,t)} + \sum_{r=1}^{R}\sum_{t=1}^{T_r} d(r,t,i) b_i'(k), \quad k = 1, \ldots, K,$$

where $b_i(k)$ is multiplied on both sides. Solving for $b_i(k)$, we obtain the re-estimation formula shown in (49) (shown at the bottom of the page). We now define

$$D_i = \sum_{r=1}^{R}\sum_{t=1}^{T_r} d(r,t,i) \tag{50}$$

$$\Delta\gamma(i,r,t) = \sum_s p(s|X, \Lambda')(C(s) - O(\Lambda'))\gamma_{i,r,s_r}(t) \tag{51}$$

and rewrite (49) as

$$b_i(k) = \frac{\sum_{r=1}^{R} \sum_{\substack{t=1 \\ s.t.\ x_{r,t}=k}}^{T_r} \Delta\gamma(i,r,t) + b_i'(k) D_i}{\sum_{r=1}^{R}\sum_{t=1}^{T_r} \Delta\gamma(i,r,t) + D_i}. \tag{52}$$

In order to optimize transition probability $a_{i,j}$, with constraint $\sum_{j=1}^{I} a_{i,j} = 1$, we apply the Lagrange multiplier method by constructing

$$W_2(\Lambda; \Lambda') = U_2(\Lambda; \Lambda') + \sum_{i=1}^{I} \lambda_i \left(\sum_{j=1}^{I} a_{i,j} - 1 \right). \tag{53}$$

Setting $\partial W_2(\Lambda;\Lambda')/\partial\lambda_i = 0$ and $\partial W_2(\Lambda;\Lambda')/\partial a_{i,j} = 0$, $j = 1, \ldots, I$, we have the following $I + 1$ equations:

$$\sum_{j=1}^{I} a_{i,j} - 1 = 0$$

$$0 = \lambda_i a_{i,j} + \sum_{r=1}^{R}\sum_{t=1}^{T_r} \overbrace{\sum_s p(s|X, \Lambda')(C(s) - O(\Lambda'))\xi_{i,j,r,s_r}(t)}^{\Delta\xi(i,j,r,t)} + \sum_{r=1}^{R}\sum_{t=1}^{T_r} d(r,t-1,i) a_{i,j}', \quad j = 1, \ldots, I.$$

Note that $\sum_{j=1}^{I} \xi_{i,j,r,s_r}(t) = \gamma_{i,r,s_r}(t)$. By solving $a_{i,j}$, we obtain the re-estimation formula shown in (54) (shown at the bottom of the page) with a standard procedure (used for deriving the EM estimate of transition probabilities [11]). Now we define

$$\tilde{D}_i = \sum_{r=1}^{R}\sum_{t=1}^{T_r} d(r,t-1,i) \tag{55}$$

$$\Delta\xi(i,j,r,t) = \sum_s p(s|X, \Lambda')(C(s) - O(\Lambda'))\xi_{i,j,r,s_r}(t) \tag{56}$$

and together with (51), we rewrite (54) as

$$a_{i,j} = \frac{\sum_{r=1}^{R}\sum_{t=1}^{T_r} \Delta\xi(i,j,r,t) + a_{i,j}' \tilde{D}_i}{\sum_{r=1}^{R}\sum_{t=1}^{T_r} \Delta\gamma(i,r,t) + \tilde{D}_i}. \tag{57}$$

The parameter re-estimation formulas (52) and (57) are unified across MMI, MCE, and MPE/MWE. What distinguishes among MMI, MCE, and MPE/MWE is the different weighing term $\Delta\gamma(i,r,t)$ in (51) and $\Delta\xi(i,j,r,t)$ in (56) due to the different $C(s)$ contained in the unified objective function. Details for computing $\Delta\gamma(i,r,t)$ for MMI, and MCE, and MPE/MWE are included in "Computing $\Delta\gamma(i,r,t)$ in the GT Formulas."

SETTING CONSTANT D_i

Values of constant D_i in (52) and $\tilde{D}_i$ in (60) determine the stability and convergence speed of the above GT/EBW algorithm. From (50), (41), and (36), we have

$$b_i(k) = \frac{\sum_{r=1}^{R} \sum_{\substack{t=1 \\ s.t.\ x_{r,t}=k}}^{T_r} \sum_s p(s|X, \Lambda')(C(s) - O(\Lambda'))\gamma_{i,r,s_r}(t) + b_i'(k) \sum_{r=1}^{R}\sum_{t=1}^{T_r} d(r,t,i)}{\sum_{r=1}^{R}\sum_{t=1}^{T_r} \sum_s p(s|X, \Lambda')(C(s) - O(\Lambda'))\gamma_{i,r,s_r}(t) + \sum_{r=1}^{R}\sum_{t=1}^{T_r} d(r,t,i)} \tag{49}$$

$$a_{i,j} = \frac{\sum_{r=1}^{R}\sum_{t=1}^{T_r} \sum_s p(s|X, \Lambda')(C(s) - O(\Lambda'))\xi_{i,j,r,s_r}(t) + a_{i,j}' \sum_{r=1}^{R}\sum_{t=1}^{T_r} d(r,t-1,i)}{\sum_{r=1}^{R}\sum_{t=1}^{T_r} \sum_s p(s|X, \Lambda')(C(s) - O(\Lambda'))\gamma_{i,r,s_r}(t) + \sum_{r=1}^{R}\sum_{t=1}^{T_r} d(r,t-1,i)} \tag{54}$$

$$D_i = \sum_{r=1}^{R}\sum_{t=1}^{T_r} d(r,t,i)$$
$$= \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{s} d'(s)p(q_{r,t}=i\,|s,\Lambda')$$
$$= \frac{1}{p(X\,|\Lambda')}\sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{s} d(s)p(q_{r,t}=i\,|s,\Lambda'). \quad (58)$$

The theoretical basis for setting D_i to ensure that (52) and (57) are growth transformations is the requirement described in (32) that $d(s)$ of (58) be sufficiently large so that $\Gamma(\Lambda') + d(s) > 0$. From (33),

$$\Gamma(\Lambda') = \delta(\chi, X)p(s)[C(s) - O(\Lambda')]$$
$$= \begin{cases} p(s)[C(s) - O(\Lambda')] & \text{if } \chi = X \\ 0 & \text{otherwise.} \end{cases}$$

Therefore, $d(s) > \max\{0, -p(s)[C(s) - O(\Lambda')]\}$. This gives

$$D_i > \frac{1}{p(X\,|\Lambda')}\sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{s} \max\{0,\ p(s)[O(\Lambda') - C(s)]\} p(q_{r,t}=i\,|s,\Lambda'). \quad (59)$$

Similarly, we can derive that

$$\tilde{D}_i > \frac{1}{p(X\,|\Lambda')}\sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{s} \max\{0,\ p(s)[O(\Lambda') - C(s)]\}\, p(q_{r,t-1}=i\,|s,\Lambda'). \quad (60)$$

In practice, D_i and $\tilde{D}_i$ given by (59) and (60) have often been found to be over conservative and unnecessarily large, causing slower convergence than those obtained through some empirical methods. We will not discuss such heuristics in this review, but would like to point out that this is still an interesting research problem and to refer the readers to the studies and discussions in [18], [41], [42], [47], [54], [59] and [61].

DISCRIMINATIVE LEARNING FOR CDHMMs

For CDHMMs, the observation space is not quantized. In this case, $X = X_1 \ldots X_R$, is a concatenation of all training tokens, and each training token X_r consists of a sequence of continuous random variables. The formulation (25) applies to discriminative learning for CDHMMs. In particular, χ in previous equations (29) and (30) is a continuous variable and hence the summation over domain χ is changed to integration over χ. That is, (29) is modified to

$$V(\Lambda;\Lambda') = \sum_{s}\sum_{q}\int_{\chi} f(\chi,q,s,\Lambda')\log f(\chi,q,s,\Lambda)d\chi, \quad (61)$$

where the integrand $f(\chi, q, s, \Lambda)$ is defined by

$$F(\Lambda;\Lambda') = \sum_{s}\sum_{q}\int_{\chi} f(\chi,q,s,\Lambda)d\chi. \quad (62)$$

Correspondingly,

$$F(\Lambda;\Lambda') = \sum_{s}\sum_{q}[C(s) - O(\Lambda')]\,p(s)p(X,q\,|s,\Lambda) + D$$
$$= \sum_{s}\sum_{q}\int_{\chi}[\Gamma(\Lambda') + d(s)]\,p(\chi,q\,|s,\Lambda)d\chi, \quad (63)$$

where

$$f(\chi,q,s,\Lambda) = [\Gamma(\Lambda') + d(s)]p(\chi,q\,|s,\Lambda) \quad (64)$$

and

$$\Gamma(\Lambda') = \delta(\chi, X)p(s)[C(s) - O(\Lambda')] \quad (65)$$

with $\delta(\chi, X)$ in (65) being the Dirac delta function. Jensen's inequality no longer applies to the secondary auxiliary function, due to the Dirac delta function. However, in the section "Setting Constant D_i for CDHMM," we will show that (61) is still a valid auxiliary function. After a similar derivation as in the preceding section, it can be shown that the transition probability estimation formula (57) stays the same as the discrete HMM case. But for the emitting probability, (46) is changed to

$$U_1(\Lambda;\Lambda') = \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I}\sum_{s} p(s\,|X,\Lambda')(C(s) - O(\Lambda'))$$
$$\times\, \gamma_{i,r,s_r}(t)\log b_i(x_{r,t}) + \sum_{r=1}^{R}\sum_{t=1}^{T_r}\sum_{i=1}^{I} d(r,t,i)$$
$$\times \int_{\chi_{r,t}} b_i'(\chi_{r,t})\,\log\, b_i(\chi_{r,t})\,d\chi_{r,t}. \quad (66)$$

GT-BASED PARAMETER ESTIMATION FOR GAUSSIAN DENSITY CDHMM

We first derive the GT-based parameter estimation formulas for the CDHMM with Gaussian distributions and then generalize them to the case of mixture-Gaussian distributions in the subsequent subsection. For the CDHMM with Gaussian distributions, the observation probability density function $b_i(x_{r,t})$ in (66) becomes a Gaussian distribution taking the following form:

$$b_i(x_{r,t}) \propto \frac{1}{|\Sigma_i|^{1/2}}\exp\left[-\frac{1}{2}(x_{r,t}-\mu_i)^T\Sigma_i^{-1}(x_{r,t}-\mu_i)\right], \quad (67)$$

where (μ_i, Σ_i), $i = 1, 2, \ldots, I$ are the mean vector and covariance matrix of the Gaussian component at state i.

To solve for μ_i and Σ_i, based on (66), we set

$$\frac{\partial U_1(\Lambda;\Lambda')}{\partial \mu_i}=0; \quad \text{and} \quad \frac{\partial U_1(\Lambda;\Lambda')}{\partial \Sigma_i}=0.$$

This gives

$$\begin{aligned}0=&\sum_{r=1}^{R}\sum_{t=1}^{T_r}\overbrace{\sum_{s}p(s|X,\Lambda')(C(s)-O(\Lambda'))\gamma_{i,r,s_r}(t)}^{\Delta\gamma(i,r,t)}\\&\times\Sigma_i^{-1}(x_{r,t}-\mu_i)+\sum_{r=1}^{R}\sum_{t=1}^{T_r}d(r,t,i)\Sigma_i^{-1}\\&\times\int_{\chi_{r,t}}b_i'(\chi_{r,t})(\chi_{r,t}-\mu_i)d\chi_{r,t}\end{aligned} \tag{68}$$

$$\begin{aligned}0=&\sum_{r=1}^{R}\sum_{t=1}^{T_r}\overbrace{\sum_{s}p(s|X,\Lambda')(C(s)-O(\Lambda'))\,\gamma_{i,r,s_r}(t)}^{\Delta\gamma(i,r,t)}\\&\times\left[\Sigma_i^{-1}-\Sigma_i^{-1}(x_{r,t}-\mu_i)(x_{r,t}-\mu_i)^T\Sigma_i^{-1}\right]\\&+\sum_{r=1}^{R}\sum_{t=1}^{T_r}d(r,t,i)\int_{\chi_{r,t}}b_i'(\chi_{r,t})\\&\times\left[\Sigma_i^{-1}-\Sigma_i^{-1}(\chi_{r,t}-\mu_i)(\chi_{r,t}-\mu_i)^T\Sigma_i^{-1}\right]d\chi_{r,t}.\end{aligned} \tag{69}$$

For a Gaussian distribution $b_i'(\chi_{r,t})=p(\chi_{r,t}|q_{r,t}=i;\Lambda')$, we have

$$\int_{\chi_{r,t}}b_i'(\chi_{r,t})d\chi_{r,t}=1,$$

$$\int_{\chi_{r,t}}\chi_{r,t}\cdot b_i'(\chi_{r,t})\,d\chi_{r,t}=\mu_i',$$

$$\int_{\chi_{r,t}}(\chi_{r,t}-\mu_i')(\chi_{r,t}-\mu_i')^T\cdot b_i'(\chi_{r,t})d\chi_{r,t}=\Sigma_i'.$$

Hence integrals in (68) and (69) have closed-form results. Next, we left-multiply both sides of (68) by Σ_i, and left- and right-multiply both sides of (69) by Σ_i. Finally, solving μ_i and Σ_i gives the GT formulas of (70) and (71) (shown at the bottom of the page), where $\Delta\gamma(i,r,t)$ is defined in (51) and D_i defined in (50).

Just as in the discrete HMM case, (70) and (71) are based on the generic discriminative objective function $O(\Lambda)$ in the form of (25), which covers MMI, MCE, and MPE/MWE as special cases. This leads to unified, GT-based parameter estimation formulas for MMI, MCE, and MPE/MWE as well as for any other discriminative objective functions that can be mapped into the rational-function form (25). Moreover, $\Delta\gamma(i,r,t)$ in (70) and (71) is defined in the same way as (51) in the discrete-HMM case—differing only in $C(s)$ for MMI, MCE, and MPE/MWE, respectively, as will be illustrated further in "Computing $\Delta\gamma(i,r,t)$ in the GT Formulas."

SETTING CONSTANT D_i FOR CDHMM

Based on Jensen's inequality, the theoretical basis for setting an appropriate constant D_i to ensure that (70) and (71) are growth transformation is the requirement specified in (32), where $d(s)$ in (58) needs to be sufficiently large to ensure that for any string s and any observation sequence χ, $\Gamma(\Lambda')+d(s)>0$, where $\Gamma(\Lambda')=\delta(\chi,X)p(s)[C(s)-O(\Lambda')]$ is defined in (33). However, for CDHMM, $\delta(\chi,X)$ is the Dirac delta function, which is a distribution with its density function value unbounded at the center point, i.e., $\delta(\chi,X)=+\infty$ when $\chi=X$. Therefore, for a string s such that $C(s)-O(\Lambda')<0$, $\Gamma(\Lambda')|_{\chi=X}=-\infty$. Under this condition, it is impossible to find a bounded $d(s)$ that ensures $\Gamma(\Lambda')+d(s)>0$ and hence Jensen's inequality may not apply. Note that this problem does not occur in the discrete HMM case, because in that case $\delta(\chi,X)$ is a Kronecker delta function taking only a finite value of either zero or one.

The above-mentioned difficulty for CDHMMs can be overcome and the same derivation can still be used, if it can be shown that there exists a sufficiently large but still bounded constant D so that $V(\Lambda;\Lambda')$ of (61), with the integrand defined by (64), is still a valid auxiliary function of $F(\Lambda;\Lambda')$; i.e., an increase of the value of $V(\Lambda;\Lambda')$ can guarantee an increase of the value of $F(\Lambda;\Lambda')$. Such a proof was developed in the recent work of [2] for GT-based MMI training for CDHMMs, and it holds for our common rational-function discriminative training criterion as well. Therefore, a bounded D_i exists according to (58) (see technical detail in [24]).

Although a sufficiently large D_i guarantees monotone convergence of the GT-based iterative estimation formulas, i.e., (52), (57) for the discrete HMM and (70), (71) for the CDHMM,

$$\mu_i=\frac{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\Delta\gamma(i,r,t)x_t+D_i\mu_i'}{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\Delta\gamma(i,r,t)+D_i}, \tag{70}$$

$$\Sigma_i=\frac{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\left[\Delta\gamma(i,r,t)(x_t-\mu_i)(x_t-\mu_i)^T\right]+D_i\Sigma_i'+D_i(\mu_i-\mu_i')(\mu_i-\mu_i')^T}{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\Delta\gamma(i,r,t)+D_i} \tag{71}$$

the value of D_i from the monotone convergence proof is a very loose upper bound and it can be too large for a reasonable convergence speed. In practice, D_i is often empirically set to achieve compromised training performance.

The empirical setting of D_i has been extensively studied from the day when EBW was proposed. In early days, only one global constant D was used for all parameters [18], [41]. Later research discovered on the empirical basis that for CDHMM, a useful lower bound on (nonglobal) D_i is the value satisfying the constraint that newly estimated variances remain positive [42]. In [59] and [60], this constraint was further explored, leading to some quadratic inequalities upon which the lower bound of D_i can be solved. Most recently, in [54], constant D_i was further bounded by an extra condition that the denominators in re-estimation formulas remain nonsingular.

In [61], the use of Gaussian-specific D_i was reported to give further improved convergence speed. For MMI, the Gaussian-specific constant D_i was set empirically to be the maximum of i) two times of the value necessary to ensure positive variances, i.e., $2 \cdot D_{\min}$, and ii) a global constant E multiplied by the denominator occupancy; e.g., $E \cdot \gamma_i^{\text{den}}$. Specifically, for MMI in the work of [61],

$$\gamma_i^{\text{den}} = \sum_{r=1}^{R} \sum_{t=1}^{T_r} \gamma_{i,r}^{\text{den}}(t) = \sum_{r=1}^{R} \sum_{t=1}^{T_r} \sum_{s_r} p(s_r | X_r, \Lambda') \gamma_{i,r,s_r}(t).$$

However, the $\Delta\Gamma(i, r, t)$ in the unified reestimation formulas (52), (57), (70), and (71) is different from the classical form in [61] by a constant factor and therefore the setting of D_i should be adjusted accordingly. This issue is discussed in details in "Computing $\Delta\Gamma(i, r, t)$ in the GT Formulas." For MPE reported in [45]–[47], the empirical setting of D_i was the same as MMI, i.e., $D_i = \max\{2 \cdot D_{\min}, E \cdot \gamma_i^{\text{den}}\}$ except that the computation of the denominator occupancy became $\gamma_i^{\text{den}} = \sum_{r=1}^{R} \sum_{t=1}^{T_r} \max(0, -\Delta\gamma(i, r, t))$. In addition, these new parameters were further smoothed with the ML estimate of parameters (which was called I-smoothing).

For MCE, the empirical setting of γ_i^{den} as

$$\sum_{r=1}^{R} \sum_{t=1}^{T_r} p(S_r | X_r, \Lambda') \sum_{s_r} p(s_r | X_r, \Lambda') \gamma_{i,r,s_r}(t)$$

was developed in the recent work of [23] and [66]. It was based on the consideration that MCE and MMI are equivalent in the special case of having only one utterance in the training set. This setting was experimentally verified with strong recognition results as reported in [23] and [66]. Further discussions and comparisons of different settings of empirical D_i can be found in [18], [23], [41], [42], [47], [54], [60] and [61].

PARAMETER ESTIMATION FOR GAUSSIAN MIXTURE CDHMM

The model parameter estimation formulas for a Gaussian-mixture HMM are similar to those for a Gaussian HMM discussed earlier. For a Gaussian-mixture HMM, the continuous observation density function $b_i(x_{r,t})$ for state i has the following form:

$$b_i(x_{r,t}) = \sum_{l=1}^{L} w_{i,l} N(x_{r,t} | \mu_{i,l}, \Sigma_{i,l}), \tag{72}$$

where $b_i(x_{r,t})$ is a mixture of L Gaussian components, $N(x_{r,t} | \mu_{i,l}, \Sigma_{i,l})$ is the lth Gaussian mixture component that takes the same form as (67), $w_{i,l}$ is a positive weight of the lth Gaussian component, and $\sum_{l=1,\ldots,L} w_{i,l} = 1$. Compared with a Gaussian HMM, there is an additional hidden component, the Gaussian component index sequence l. The hidden sequence l can be accommodated in (61) by the same way that we exploited to treat the hidden state sequence q. Then after similar derivation steps, we can obtain the parameter estimation formulas (73) and (74) (shown at the bottom of the page), where $D_{i,l}$ and $\Delta\gamma(i, l, r, t)$ are defined in a similar way to (50) and (51), i.e.,

$$D_{i,l} = \sum_{r=1}^{R} \sum_{t=1}^{T_r} d(r, t, i, l), \tag{75}$$

$$\Delta\gamma(i, l, r, t) = \sum_{s} p(s | X, \Lambda')(C(s) - O(\Lambda')) \gamma_{i,l,r,s_r}(t), \tag{76}$$

and $\gamma_{i,l,r,s_r}(t) = p(q_{r,t} = i, l | X_r, s_r, \Lambda')$ is the occupation probability of Gaussian mixture component l of state i, at time t in the rth utterance. Accordingly, the empirical setting of $D_{i,l}$ takes similar forms as discussed in the previous section, except that $\Delta\gamma(i, l, r, t)$ and $\gamma_{i,l,r,s_r}(t)$ will be used instead. Estimation of the mixture component weights $w_{i,l}$ is similar to the discrete HMM estimation case and will not be described here.

$$\mu_{i,l} = \frac{\sum_{r=1}^{R} \sum_{t=1}^{T_r}, \Delta\gamma(i, l, r, t) x_t + D_{i,l} \mu'_{i,l}}{\sum_{r=1}^{R} \sum_{t=1}^{T_r} \Delta\gamma(i, l, r, t) + D_{i,l}}, \tag{73}$$

$$\Sigma_{i,l} = \frac{\sum_{r=1}^{R} \sum_{t=1}^{T_r} \left[\Delta\gamma(i, l, r, t)(x_t - \mu_{i,l})(x_t - \mu_{i,l})^T\right] + D_{i,l} \Sigma'_{i,l} + D_{i,l}(\mu_{i,l} - \mu'_{i,l})(\mu_{i,l} - \mu'_{i,l})^T}{\sum_{r=1}^{R} \sum_{t=1}^{T_r} \Delta\gamma(i, l, r, t) + D_{i,l}} \tag{74}$$

RELATED WORK AND DISCUSSIONS

RELATION TO OTHER APPROACHES

In recent articles [35] and [54], an approach was proposed to unify a number of discriminative learning methods including MMI, MPE, and MPE/MWE (the earlier article [54] did not include MPE/MWE). Functional similarities and differences among MMI, MCE, and MPE/MWE criteria were noted and discussed in [35], [54]. In this article, the proposed framework takes an additional step of unifying these criteria in a canonical rational-function form (25), and GT-based discriminative learning is applied to this rational-function that includes MMI, MCE, and MPE/MWE criteria as special cases. This is significant from two perspectives. First, it provides a more precise and direct insight into the fundamental relations among MMI, MCE, and MPE/MWE criteria at the objective function level based on the common rational-function form (25). Second, it enables a unified GT-based parameter optimization framework that applies to MMI, MCE, MPE/MWE, and other discriminative criteria, as long as their objective functions can be represented by (25).

The proposed framework in [35] was based on the objective function of the following form (rewritten using the mathematical notations adopted in this article for easy comparisons):

$$O(\Lambda) = \frac{1}{R}\sum_{r=1}^{R} f\left(\frac{1}{\eta}\log\frac{\sum_{s_r} p^{\eta}(X_r, s_r|\Lambda)C_{DT}(s_r)}{\sum_{s_r\in M_r} p^{\eta}(X_r, s_r|\Lambda)}\right), \quad (77)$$

where $C_{\mathrm{DT}}(s_r)$ takes the same value as in Table 1. The choices of the smoothing function $f(z)$, the competing word sequences M_r, and the weight value η in (77) are provided in Table 2 for the different types of DT criteria. In Table 2, q is the slope of a sigmoid smoothing function.

Equation (77) indicates that different discriminative criteria can have a similar form of kernel and differ by the criterion-dependent smoothing function $f(z)$ that modulates the kernel, where the objective function is a sum of smoothing functions. Equation (77) is a generic description of the objective functions of MMI, MCE, and MPE/MWE. However, it is not in a general form of a rational function (defined as a ratio of two polynomial functions) due to the presence of the nonlinear function $f(z)$. The important distinction of product versus summation of utility functions among these criteria is not explicitly addressed. In the approach presented in this article, we address this issue directly and show that the objective functions from MMI, MCE, and MPE/MWE criteria can have a definitive rational-function form (25), and for each discriminative criterion, the objective function differs only by a model-independent quantity $C_{\mathrm{DT}}(s_1 \ldots s_R)$.

Furthermore, as shown in Table 2, since $f(z)$ is a nonlinear function for the MPE/MWE and MCE criteria, the original GT solution [18], while directly applicable to MMI with $f(z)$ being an identity function and z being the logarithm of a rational function (since sum of log becomes log of product), is not directly applicable to the objective functions of the MPE/MWE and MCE criteria (since the sum stays when $f(z)$ is nonlinear). In order to circumvent this difficulty, the theorem described in [30] is applied. In [30], the original objective function is approximated by a Taylor series expansion. Then, via a similar approach to that of [18], the GT-based parameter optimization may be applied to the partial sum of the Taylor series expansion, which is a polynomial with a finite degree. This forms the theoretical basis of the earlier GT-based methods for MCE and MPE/MWE [35], [54]. However, the positive growth of the partial sum depends on the degree of that partial sum (see more detailed discussions on this point in [18]), and it vanishes when the degree goes to infinity. It may vanish even faster than the error of Taylor series approximation does. Therefore, it has not been definitively shown that the re-estimation formula ensures true growth of the value of the objective function with iteration.

In contrast, the unified rational-function approach described in this article departs from the work of [35] and [54]. It is free from the Taylor series approximation and it shows that the objective functions for the MMI, MCE, and MPE/MWE criteria have a common definitive rational-function form (25). Therefore, the GT-based parameter optimization framework can be directly applied to (25) in a constructive way. The approach taken in this article is based on the work in [2] and [20] rather than on the work of [3] and [18]. Moreover, the unified representation of the discriminative objective functions developed in this article opens a way to apply other rational-function based optimization methods (e.g., the method based on the reverse Jensen inequality [26]) to MMI, MCE, and MPE/MWE-based classifier design. Using the structure of the rational function, we expect that all desirable algorithmic properties of the parameter optimization procedures presented in this article can be established and justified.

RELATION TO GRADIENT-BASED OPTIMIZATION

The relation between the GT/EBW methods and gradient-based methods has been studied in the literature (e.g., [2], [53], [54]). In addition to the critical difference in the convergence properties, the learning speed of GT/EBW-based updating formula (70) is comparable to a quadratic Newton

[TABLE 2] CHOICES OF THE SMOOTHING FUNCTION $f(z)$, ALTERNATIVE WORD SEQUENCES M_r, AND EXPONENT WEIGHT η IN (77) FOR VARIOUS TYPES OF DT CRITERIA. THIS IS MODIFIED FROM THE ORIGINAL TABLE IN [54].

CRITERIA	SMOOTHING FUNCTION $f(z)$	ALTERNATIVE WORD SEQUENCES M_r	η
MCE (*N*-BEST)	$-1/[1+\exp(2qz)]$	$\{s_r\}$ EXCLUDING S_r	≥ 1
MCE (ONE-BEST)	$-1/[1+\exp(2qz)]$	$\{s_{r,1}\}$	N/A
MPE/MWE	$\exp(z)$	ALL POSSIBLE LABEL SEQUENCE $\{s_r\}$	1
MMI	z	ALL POSSIBLE LABEL SEQUENCE $\{s_r\}$	1

update; i.e., it can be formulated as a gradient ascent with the step size that approximates inverse Hessian H of the objective function. Let us take the mean vector estimation as an example for the objective function of the form (25) in the case of CDHMM. The gradient of $O(\Lambda)$ w.r.t. μ_i can be shown to be

$$\nabla_{\mu_i} O(\Lambda)|_{\Lambda=\Lambda'} = \Sigma_i'^{-1} \sum_{r=1}^{R} \sum_{t=1}^{T_r} \Delta\gamma(i,r,t)(x_t - \mu_i'). \quad (78)$$

On the other hand, we can rewrite the GT formula of (70) into the following equivalent form:

$$\begin{aligned} \mu_i &= \mu_i' + \frac{1}{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\Delta\gamma(i,r,t) + D_i} \\ &\quad \times \sum_{r=1}^{R}\sum_{t=1}^{T_r}\Delta\gamma(i,r,t)(x_t - \mu_i') \\ &= \mu_i' + \frac{1}{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\Delta\gamma(i,r,t) + D_i}\Sigma_i' \cdot \nabla_{\mu_i} O(\Lambda)|_{\Lambda=\Lambda'}. \end{aligned} \quad (79)$$

Consider the quadratic Newton update, where the Hessian H_i for μ_i can be approximated by the following equation after dropping the dependency of μ_i with $\Delta\gamma(i,r,t)$:

$$H_i = \nabla^2_{\mu_i} O(\Lambda)|_{\Lambda=\Lambda'} \approx -\Sigma_i'^{-1} \sum_{r=1}^{R}\sum_{t=1}^{T_r}\Delta\gamma(i,r,t).$$

Therefore, the updating formula of GT in (70) can be further rewritten to

$$\mu_i \approx \mu_i' \underbrace{\frac{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\Delta\gamma(i,r,t)}{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\Delta\gamma(i,r,t) + D_i}}_{\varepsilon_i} H_i^{-1}\nabla_{\mu_i} O(\Lambda)|_{\Lambda=\Lambda'}, \quad (80)$$

which approximates the quadratic Newton update $\mu_i = \mu_i' - \alpha \cdot H_i^{-1}\nabla_{\mu_i} O(\Lambda)|_{\Lambda=\Lambda'}$ and usually gives a faster learning speed than the simple gradient-based search.

Other popular and effective gradient-based methods exist for optimizing discriminative training criteria [15], [21], [31], [37], [38]. For instance, Quickprop [17] is a batch-mode, second-order optimization method that approximates Newton's optimization, with the help of heuristics to determine the proper update step size. Rprop [51] is another batch-mode optimization method, which performs dynamic scaling of the update step size for each parameter based on different kinds of heuristics. In [38], a comprehensive study of gradient-based optimization methods for MCE training, including batch and semibatch probabilistic descent (PD), Quickprop, and Rprop, is given for large vocabulary speech recognition tasks. It was shown that the MCE criterion can be optimized by using these gradient-based methods, and improved recognition accuracies were reported. Furthermore, there exist other gradient-based methods such as Broyden-Fletcher-Goldfarb-Shanno (BFGS) and conjugate gradient search [5], [14]. Although both of these methods are more complicated to implement for large scale discriminative training tasks, they are superior to other gradient-descent techniques in terms of the convergence properties. Readers are referred to [5] and [52] for further discussions.

THE DERIVED RATIONAL-FUNCTION FORM FOR MMI, MCE, AND MPE/MWE ALLOWS THE GT/EBW-BASED PARAMETER OPTIMIZATION FRAMEWORK TO BE APPLIED DIRECTLY IN DISCRIMINATIVE LEARNING.

In contrast to the popular gradient-based methods discussed above, we can view a class of optimization methods with a re-estimation style, including expectation-maximization (EM) algorithm and EBW algorithm, as GT-based methods in a broad sense. The GT-based methods are designed for the objective functions with special, rational-functional forms, and the GT algorithm can ensure rigorous monotone growth of the value of the objective functions iteratively. From this perspective, on the one hand, GT-based methods are less general than gradient-based ones. On the other hand, they give desirable monotone convergence in training. Further, although GT-based parameter re-estimation formulas may be rewritten into gradient-based forms, the step sizes are specifically derived so that monotone convergence is guaranteed. This critical property differentiates them from general gradient-based methods.

The advanced gradient-based methods discussed above, such as batch and semibatch PD, Quickprop, Rprop, BFGS, and conjugate gradient, are alternatives to the GT/EBW-method for optimizing discriminative training criteria. Although theoretically the GT/EBW method has the desirable monotone convergence property, empirical setting of D is used in practice to speed up training with the trade-off for monotone convergence. This makes rigorous comparisons between GT/EBW-based and advanced gradient-based methods difficult. In the literature, experimental results of both types of methods have been reported on various speech recognition tasks [21], [23], [31], [54], [42].

Algorithmic convergence of parameter estimation is a central issue for classifier design using discriminative training criteria. Search for more powerful discriminative criteria and optimization methods in classifier design remains an area of active and ongoing research. It is our hope that the unified rational-function based objective function representation reviewed in this paper can provide additional structural formulation and can motivate the development of new learning algorithms to improve the discriminative power of sequential pattern classifiers and recognizers.

SUMMARY

In this article, we studied the objective functions of MMI, MCE, and MPE/MWE for discriminative learning in sequential pattern recognition. We presented an approach that unifies the objective functions of MMI, MCE, and MPE/MWE in a common rational-function form of (25). The exact structure of the rational-function form for each discriminative criterion was derived and studied. While the rational-function form of MMI has been known in the past, we provided the theoretical proof that the similar rational-function form exists for the objective functions of MCE and MPE/MWE. Moreover, we showed that the rational function forms for objective functions of MMI, MCE, and MPE/MWE differ in the constant weighting factors $C_{DT}(s_1 \ldots s_R)$ and these weighting factors depend only on the

COMPUTING $\Delta\gamma(i, r, t)$ IN THE GT FORMULAS

In (51), computing $\Delta\gamma(i, r, t)$ involves summation over all possible super-string label sequences $s = s_1 \ldots s_R$. The number of training tokens (sentence strings), R, is usually very large. Hence, the summation over s needs to be decomposed and simplified. To proceed, we use the notations of $s' = s_1 \ldots s_{r-1}$, $s'' = s_{r+1} \ldots s_R$, $X' = X_1 \ldots X_{r-1}$, and $X'' = X_{r+1} \ldots X_R$. Then, from (51), we have,

$$\Delta\gamma(i, r, t) = \sum_{s_r} p(s_r|X_r, \Lambda') \times \underbrace{\left[\sum_{s'}\sum_{s''} p(s', s''|X', X''; \Lambda')(C(s', s_r, s'') - O(\Lambda'))\right]}_{\Psi} \times \gamma_{i,r,s_r}(t), \tag{81}$$

where factor Ψ is the average deviation of the accuracy count for the given string s_r. The remaining steps in simplifying the computation of $\Delta\gamma(i, r, t)$ will be separate for MMI and MCE/MPE/MWE because the parameter-independent accuracy count function $C(s)$ for them takes the product and summation form, respectively (as shown in Table 1).

Product form of $C(s)$ (for MMI)

For MMI, we have

$$C(s) = C(s_1, \ldots, s_R) = \prod_{r=1}^{R} C(s_r) = \prod_{r=1}^{R} \delta(s_r, S_r)$$

in a product form. Using $C(s', s_r, s'') = C(s_r) \cdot C(s', s'')$, we simplify factor Ψ in (81) to

$$\begin{aligned}\Psi &= C(s_r) \cdot \sum_{s'}\sum_{s''} p(s', s''|X', X''; \Lambda')C(s', s'') - O(\Lambda') \\ &= O(\Lambda')\left(\frac{C(s_r) \cdot \sum_{s'}\sum_{s''} p(s', s''|X', X''; \Lambda')C(s', s'')}{O(\Lambda')} - 1\right).\end{aligned} \tag{82}$$

The idea behind the above steps is to make use of the product form of the $C(s)$ function for canceling out common factors in both $O(\Lambda')$ and $C(s)$ functions. To proceed, we now factorize $O(\Lambda')$ as follows:

$$\begin{aligned}O(\Lambda') &= \frac{\left[\sum_{s_r} p(s_r, X_r|\Lambda')C(s_r)\right]\left[\sum_{s'}\sum_{s''} p(s', s'', X', X''|\Lambda)C(s', s'')\right]}{[p(X_r|\Lambda')]\,[p(X', X''|\Lambda\prime)]} \\ &= p(S_r|X_r, \Lambda') \cdot \sum_{s'}\sum_{s''} p(s', s''|X', X''; \Lambda')C(s', s''),\end{aligned}$$

where the last step uses $C(s_r) = \delta(s_r, S_r)$. Substituting this to (82) then gives the simplification of

$$\Psi = O(\Lambda')\left(\frac{C(s_r)}{p(S_r|X_r, \Lambda')} - 1\right). \tag{83}$$

Substituting (83) to (81) and using $C(s_r) = \delta(s_r, S_r)$ again, we obtain

$$\Delta\gamma(i, r, t) = O(\Lambda')\left[\gamma_{i,r,S_r}(t) - \sum_{s_r} p(s_r|X_r, \Lambda')\gamma_{i,r,s_r}(t)\right]. \tag{84}$$

In the re-estimation formulas (70) and (71), if we divide both the numerator and denominator by $O(\Lambda')$, $\Delta\gamma(i, r, t)$ in (84) can take a simplified form of

$$\begin{aligned}\Delta\tilde{\gamma}(i, r, t) &= \left[\gamma_{i,r,S_r}(t) - \sum_{s_r} p(s_r|X_r, \Lambda')\gamma_{i,r,s_r}(t)\right] \\ &= \gamma_{i,r}^{\text{num}}(t) - \gamma_{i,r}^{\text{den}}(t).\end{aligned} \tag{85}$$

The corresponding constant D_i in the re-estimation formulas (70) and (71) then becomes

$$\tilde{D}_i = D_i / O(\Lambda'). \tag{86}$$

Substituting the above into (70) and (71), we have the GT/EBW formulas for MMI, shown in (87) and (88) at the bottom of the next page. This gives the classical GT/EBW-based MMI re-estimation formulas described in [41] and [61].

Equation (84) or (85) gives an N-best string based solution to computing $\Delta\gamma(i, r, t)$. This is illustrated by the string-level summation over s_r (i.e., the label sequence for token r, including both correct and incorrect strings). For N-best string-level discriminative training, the summation over s_r in (84) or (85) amounts to going through all N-best string hypotheses and is computationally inexpensive when N is relatively small (e.g., N in the order of thousands as typical for most N-best experiments).

When a lattice instead of an explicit N-best list is provided for competing hypotheses in discriminative training, in theory, (84) or (85) can be applied just as for the N-best string based solution already discussed. This is because a lattice is nothing more than a compact representation of N best strings. However, since N in this equivalent "N-best list" would be huge (in the order of billions or more [66]), more efficient techniques for dealing with the summation over s_r in computing (84) or (85) will be needed. Readers are referred to [24] for details of such computation.

labeled sequence $s_1 \ldots s_R$, and are independent of the parameter set Λ to be optimized.

The derived rational-function form for MMI, MCE, and MPE/MWE allows the GT/EBW-based parameter optimization framework to be applied directly in discriminative learning. In the past, lack of the appropriate rational-function form was a difficulty for MCE and MPE/MWE, because without this form, the GT/EBW-based parameter optimization framework cannot be directly applied. Based on the unified rational-function form, in a tutorial style, we derived the GT/EBW-based parameter optimization formulas for both discrete HMMs and CDHMMs in discriminative learning using MMI, MCE, and MPE/MWE criteria.

The unifying review provided in this article has been based upon a large number of earlier contributions that have been

Summation Form of $C(s)$ (MCE and MPE/MWE)

Different from MMI, for MCE and MPE/MWE, we have

$$C(s) = C(s_1, \ldots, s_R) = \sum_{r=1}^{R} C(s_r),$$

or

$$C(s', s_r, s'') = C(s_r) + C(s', s'').$$

That is, the C function is in a summation instead of a product form. This changes the simplification steps for factor Ψ of (81) as follows:

$$\begin{aligned}\Psi &= \sum_{s'}\sum_{s''} p_{\Lambda'}(s', s'' | X', X'')C(s_r)\\ &\quad + \sum_{s'}\sum_{s''} p(s', s'' | X', X'';\ \Lambda')C(s', s'') - O(\Lambda')\\ &= C(s_r) + \sum_{s'}\sum_{s''} p(s', s'' | X', X'';\ \Lambda')C(s', s'') - O(\Lambda'). \quad (89)\end{aligned}$$

The idea behind the above steps is to make use of the summation form of the $C(s)$ function for subtracting out the common terms in the $O(\Lambda')$ function. To achieve this, we decompose $O(\Lambda')$, based on its original nonrational form, e.g., (21) or (17) and (18), as follows:

$$\begin{aligned}O(\Lambda') &= \frac{\sum_{s_r} p(X_r, s_r | \Lambda')\ C(s_r)}{\sum_{s_r} p(X_r, s_r | \Lambda')} + \sum_{i=1, i\neq r}^{R} \frac{\sum_{s_i} p(X_i, s_i | \Lambda')\ C(s_i)}{\sum_{s_i} p(X_i, s_i | \Lambda')}\\ &= \frac{\sum_{s_r} p(X_r, s_r | \Lambda')\ C(s_r)}{\sum_{s_r} p(X_r, s_r | \Lambda')} + \sum_{s', s''} p(s', s'' | X', X'' | \Lambda')C(s', s'').\end{aligned}$$

The second term above cancels out the same term in (89), leading to the simplification of

$$\Psi = C(s_r) - \frac{\sum_{s_r} p(s_r, X_r | \Lambda')C(s_r)}{\sum_{s_r} p(s_r, X_r | \Lambda')}. \quad (90)$$

Now, substituting (90) back to (81), we obtain

$$\begin{aligned}\Delta\gamma(i, r, t) &= \sum_{s_r} p(s_r | X_r, \Lambda') \left(C(s_r) - \frac{\sum_{s_r} p(X_r, s_r | \Lambda')C(s_r)}{\sum_{s_r} p(X_r, s_r | \Lambda')} \right)\\ &\quad \times \gamma_{i,r,s_r}(t). \quad (91)\end{aligned}$$

For MCE that has $C(s_r) = \delta(s_r, S_r)$, the previous equation can be further simplified as:

$$\begin{aligned}\Delta\gamma(i, r, t) &= p(S_r | X_r, \Lambda')\\ &\quad \times \left[\gamma_{i,r,S_r}(t) - \sum_{s_r} p(s_r | X_r, \Lambda')\gamma_{i,r,s_r}(t) \right]. \quad (92)\end{aligned}$$

Again, if a lattice instead of an N-best list is provided for discriminative learning, a huge number of terms in the summation over s_r in (19) would be encountered. In order to keep the computation manageable, efficient computation of (91) based on the lattice is needed, which we describe in [24].

$$\mu_i = \frac{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\left[\gamma_{i,r}^{\text{num}}(t) - \gamma_{i,r}^{\text{den}}(t)\right] x_t + \tilde{D}_i\mu_i'}{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\left[\gamma_{i,r}^{\text{num}}(t) - \gamma_{i,r}^{\text{den}}(t)\right] + \tilde{D}_i}, \quad (87)$$

$$\Sigma_i = \frac{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\left[\gamma_{i,r}^{\text{num}}(t) - \gamma_{i,r}^{\text{den}}(t)\right](x_t - \mu_i)(x_t - \mu_i)^T + \tilde{D}_i\Sigma_i' + \tilde{D}_i(\mu_i - \mu_i')(\mu_i - \mu_i')^T}{\sum_{r=1}^{R}\sum_{t=1}^{T_r}\left[\gamma_{i,r}^{\text{num}}(t) - \gamma_{i,r}^{\text{den}}(t)\right] + \tilde{D}_i} \quad (88)$$

cited and discussed throughout the article. Here we provide a brief summary of such background work. The GT technique was originally introduced in [3] and [4] for ML estimation of discrete HMMs and it was extended in [18] to handle MMI estimation of the same type of HMMs. The dissertation of [41] extended the work of [18] from discrete HMMs to Gaussian CDHMMs in a small-scale speech recognition task. Extension to large-scale speech recognition tasks was accomplished in the work of [59] and [60]. The dissertation of [47] further improved the MMI criterion to that of MPE/MWE. In a parallel vein, the work of [20] provided an alternative approach to that of [41], with an attempt to more rigorously provide a CDHMM model re-estimation formula that gives positive growth of the MMI objective function. A crucial error of this attempt was corrected in [2] for establishing an existence proof of such positive growth. In yet another parallel vein in the development of discriminative learning, the rudimentary form of MCE emerged in [1], which was fully developed in [28] and [29] showing how it could be applied to speech recognition. The work of [22] and [23] further showed that using the formulation of [20] instead of that of [41], GT can also be applied to MCE. Recently, the work of [37] and [38] demonstrated that MCE can scale to large-scale speech recognition. Further, that nonsequential gradient method can be successfully used for MCE learning of CDHMMs was demonstrated in [52]. More

IN CONTRAST TO MMI AND MCE, WHICH ARE TYPICALLY AIMED AT LARGE SEGMENTS OF PATTERN SEQUENCES, MPE AIMS AT THE PERFORMANCE OPTIMIZATION AT THE SUBSTRING PATTERN LEVEL.

TWO EMPIRICAL ISSUES IN MCE IMPLEMENTATION

Here we discuss two empirical issues in MCE implementation that were raised in the section "Discriminative Learning Criteria of MMI, MCE, and MPE/MWE." First, in (11), if we use the exponent scale factor $\eta \neq 1$, we can obtain the following result corresponding to (13):

$$l_r(d_r(X_r,\Lambda)) = \frac{\sum_{s_r, s_r \neq S_r} p^{\eta}(X_r, s_r|\Lambda)}{\sum_{s_r} p^{\eta}(X_r, s_r|\Lambda)}.$$

The corresponding result to (16) then becomes

$$O_{\mathrm{MCE}}(\Lambda) = \sum_{r=1}^{R} \frac{p^{\eta}(X_r, S_r|\Lambda)}{\sum_{s_r} p^{\eta}(X_r, s_r|\Lambda)},$$

which can be reformulated into a rational-function using the same steps as in the section "Rational-Function Form for the Objective Function of MCE"

$$O_{\mathrm{MCE}}(\Lambda) = \frac{\sum_{s_1\ldots s_R} p^{\eta}(X_1\ldots X_R, s_1\ldots s_R|\Lambda)\, C_{\mathrm{MCE}}(s_1\ldots s_R)}{\sum_{s_1\ldots s_R} p^{\eta}(X_1\ldots X_R, s_1\ldots s_R|\Lambda)}. \quad (93)$$

The remaining derivations in the sections "Discriminative Learning for Discrete HMMs based on the GT Framework" and "Discriminative Learning for CDHMMs" will no longer follow strictly for the more general and practical case of (93). However (as our experiments reported in [22]), in the MCE implementation, we modify (91) for computing $\Delta\gamma(i,r,t)$ in the following way in order to include the effects of the exponent scale factor:

$$\Delta\gamma(i,r,t) = \sum_{s_r} \tilde{p}(s_r|X_r,\Lambda') \times \left(C(s_r) - \sum_{s_r} \tilde{p}(X_r|s_r,\Lambda') C(s_r) \right) \gamma_{i,r,s_r}(t), \quad (94)$$

where $\tilde{p}(s_r|X_r,\Lambda')$ is the generalized posterior probability of s_r, which can be computed as

$$\tilde{p}(s_r|X_r,\Lambda') = \frac{p^{\eta}(X_r, s_r|\Lambda')}{\sum_{s_r} p^{\eta}(X_r, s_r|\Lambda')}. \quad (95)$$

The second empirical MCE implementation issue concerns the use of $\alpha \neq 1$ in (12). For one-best MCE, α acts as η, or we equivalently set $\eta = \alpha$ and $\alpha = 1$. Then we can compute $\Delta\gamma(i,r,t)$ according to (94). For N-best MCE ($N > 1$), given the discriminant function defined in (11) and the sigmoid function defined in (12), we have the following result [which corresponds to (13)]:

$$l_r(d_r(X_r,\Lambda)) = \frac{\left(\sum_{s_r, s_r \neq S_r} p^{\eta}(X_r, s_r|\Lambda)\right)^{\alpha}}{p^{\eta\cdot\alpha}(X_r, S_r|\Lambda) + \left(\sum_{s_r, s_r \neq S_r} p^{\eta}(X_r, s_r|\Lambda)\right)^{\alpha}}. \quad (96)$$

Now, α is applied outside of the summation of scaled joint probabilities over all competing strings, making the rigorous computation intractable. In our practical MCE implementation, we instead use $\sum_{s_r, s_r \neq S_r} p^{\eta\cdot\alpha}(X_r, s_r|\Lambda)$ to approximate

$$\left(\sum_{s_r, s_r \neq S_r} p^{\eta}(X_r, s_r|\Lambda)\right)^{\alpha}.$$

This approximation (which is exact when η approaches infinity) makes it equivalent to setting the new η as $\alpha \cdot \eta$, and setting new $\alpha = 1$. We can again compute $\Delta\gamma(i,r,t)$ according to (94). It should be noted that, with this approximation, the computation for the lattice-based MCE does not require removing the correct word string S_r from the lattice. This contrasts with the solution in [35] and [54], where the removal was necessary without using the approximation, making it more difficult to implement in practice.

The above two empirical solutions have been implemented successfully in our speech recognition system, yielding strong practical results (published in [23] and [66]) that validate the solutions.

recently, the work in [35], [53], and [54] showed that MPE, MCE, and MMI are related by a generic, nonrational-function description of the objective function. Finally, in the current article, we show that all MMI, MCE, and MPE/MWE can be rigorously formulated as a rational function enabling rigorous GT-style optimization.

This article is motivated by the striking success of the MMI, MCE, and MPE/MWE-based discriminative criteria in speech recognition. Yet in the past, there was a lack of common understanding of the interrelation among these techniques, despite the relatively long history of MMI (since 1983 [39]), MCE (since 1967 [1]), and MPE/MWE (since 2002 [45]). Due to the complexity of these techniques and the lack of a common underlying theoretical theme and structure, disparate discriminative learning procedures were developed and parameter optimization has become a major issue. The main goal of this article is to provide an underlying foundation for MMI, MCE, and MPE/MWE at the objective function level to facilitate the development of new parameter optimization techniques and to incorporate other pattern recognition concepts, e.g., discriminative margins [66], into the current discriminative learning paradigm.

ACKNOWLEDGMENT

We would like to thank an anonymous reviewer for pointing us to the work of [2].

AUTHORS

Xiaodong He (xiaohe@microsoft.com) received the B.S. degree from Tsinghua University, Beijing, China in 1996, the M.S. degree from the Chinese Academy of Science, Beijing, in 1999, and the Ph.D. degree from the University of Missouri-Columbia in 2003. He joined the Speech and Natural Language group of Microsoft in 2003 and joined the Natural Language Processing group of Microsoft Research at Redmond, Washington, where he currently serves as a researcher. His current research interests include statistical pattern recognition and machine learning, speech and natural language processing, machine translation, and speech recognition. In these areas, he published 30 refereed papers and is inventor and coinventor of ten patents. He served in the program committee of several technical conferences and served as reviewer of many technical journals. He is a Member of the IEEE, ACL, and Sigma Xi.

Li Deng (deng@microsoft.com) received the B.S. degree from the University of Science and Technology of China in 1982 and the M.S. and Ph.D. degrees from the University of Wisconsin-Madison in 1984 and 1986, respectively. He is a principal researcher at Microsoft Research, Redmond, and affiliate professor of electrical engineering at the University of Washington, Seattle. He was a professor at the University of Waterloo, Ontario (1989–1999) and held positions at Massachusetts Institute of Technology (MIT) in Cambridge, and ATR, Kyoto, Japan. He published more than 300 refereed papers in leading international conferences and journals, 15 book chapters, and two books. He was granted more than 20 patents as inventor or coinventor in acoustics, speech/language technology, multimedia/multisensor human-computer interaction, and signal processing. He currently serves on the IEEE Signal Processing Society's Multimedia Signal Processing Technical committee and is area editor of *IEEE Signal Processing Magazine* and associate editor for *IEEE Signal Processing Letters*. He was a technical chair of IEEE International Conference on Acoustics, Speech, and Signal Processing, and the general chair of the IEEE Workshop on Multimedia Signal Processing. He is Fellow of the Acoustical Society of America, and was a member of the IEEE Signal Processing Society Board of Governors. He is a Fellow of the IEEE.

Wu Chou (wuchou@avaya.com) received the M.S. degree in mathematics in 1986, the M.S. degree in electrical engineering in 1987, the M.S. degree in statistics in 1988, and the Ph.D. degree in electrical engineering in June 1990, all from Stanford University, California. He joined the Information Principle Research Lab of AT&T Bell Laboratories in July 1990. He moved to Avaya Labs Research in October 2000. Currently, he is director and Avaya Labs Fellow. He received the Bell Laboratories President's Gold Award in 1997 and Avaya Leadership Award in 2005. He is the coeditor of the book, *Pattern Recognition in Speech and Language Processing*. He was awarded 21 U.S. and international patents, authored or coauthored over 100 refereed technical papers, three book chapters, and one edited book. He is a Fellow of the IEEE.

REFERENCES

[1] S. Amari, "A theory of adaptive pattern classifiers," *IEEE Trans. Electron. Computers*, vol. 16, pp. 299–307, June 1967.

[2] S. Axelrod, V. Goel, R. Gopinath, P. Olsen, and K. Visweswariah, "Discriminative estimation of subspace constrained gaussian mixture models for speech recognition," *IEEE Trans. Audio Speech Lang. Processing*, vol. 15, no. 1, pp. 172–189, Jan. 2007.

[3] L.E. Baum and J.A. Eagon, "An inequality with applications to statistical prediction for functions of Markov processes and to a model of ecology," *Bull. Amer. Math. Soc.*, vol. 73, pp. 360–363, Jan. 1967.

[4] L. Baum and G. Sell, "Growth transformations for functions on manifolds," *Pacific J. Math.*, vol. 27, no. 2, pp. 211–227, 1968.

[5] R. Battiti, "First- and second- order methods for learning: Between steepest descent and Newton's method," *Neural Comput.*, vol. 4, pp. 141–166, Mar. 1992.

[6] E. Birney, "Hidden Markov models in biological sequence analysis," *IBM J. Res. Develop.*, vol. 45, pp. 449–454, May 2001.

[7] P. Brown, "The acoustic modeling problem in automatic speech recognition," Ph.D. dissertation, Carnegie Mellon Univ., Dept. Comp Science, Pittsburgh, PA, 1987.

[8] W. Chou and L. Li, "A minimum classification error framework for generalized linear classifier in machine learning and its application to text categorization/retrieval," in *Proc. Int. Conf. Machine Learning Applications*, Dec. 2004, pp. 382–390.

[9] W. Chou, "Minimum classification error approach in pattern recognition," in *Pattern Recognition in Speech and Language Processing*, W. Chou and B.-H. Juang, Eds. Boca Raton, FL: CRC, 2003, pp. 1–49.

[10] M. Collins, "Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms," in *Proc. Empirical Methods of Natural Language Processing (EMNLP)*, 2005, pp. 1–8.

[11] L. Deng and D. O'Shaughnessy, *Speech Processing—A Dynamic and Optimization-Oriented Approach*. New York: Marcel Dekker, 2003.

[12] L. Deng, D. Yu, and A. Acero, "A generative modeling framework for structured hidden speech dynamics," in *Proc. Neural Information Processing System (NIPS) Workshop*, Whistler, British Columbia, Canada, Dec. 2005, pp. 11–14.

[13] L. Deng, J. Wu, J. Droppo, and A. Acero, "Analysis and comparison of two feature extraction/compensation algorithms," *IEEE Signal Processing Lett.*, vol. 12, no. 6, pp. 477–480, June 2005.

[14] J.E. Dennis and R.B. Schnabel, "Numerical methods for unconstrained optimization and nonlinear equations," *Classics in Applied Mathematics*, Philadelphia, PA: SIAM, 1996.

[15] J. Droppo and A. Acero, "Joint discriminative front end and back end training for improved speech recognition accuracy," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, 2006, pp. 281–284.

[16] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids.* Cambridge, U.K.: Cambridge Univ. Press, 1998.

[17] S.E. Fahlman, "An empirical study of learning speed in back-propagation networks," Carnergie Mellon Univ., Pittsburgh, PA, 1988, Tech. Rep. CMU-CS-88-162.

[18] P. Gopalakrishnan, D. Kanevsky, A. Nadas, and D. Nahamoo, "An inequality for rational functions with applications to some statistical estimation problems," *IEEE Trans. Inform. Theory.*, vol. 37, pp. 107–113, Jan. 1991.

[19] Y. Gao and J. Kuo, "Maximum entropy direct models for speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 14, no. 3, pp. 873–881, 2006.

[20] A. Gunawardana and W. Byrne, "Discriminative speaker adaptation with conditional maximum likelihood linear regression," in *Proc. EUROSPEECH*, 2001, pp. 1203–1206.

[21] A. Gunawardana, M. Mahajan, A. Acero, and J. Platt, "Hidden conditional random fields for phone classification," in *Proc. Interspeech*, Lisbon, Portugal, 2005, pp. 1117–1120.

[22] X. He and W. Chou, "Minimum classification error linear regression for acoustic model adaptation of continuous density HMMs," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Apr. 2003, pp. 556–559.

[23] X. He, L. Deng, and W. Chou, "A novel learning method for hidden Markov models in speech and audio processing," in *Proc. IEEE Workshop on Multimedia Signal Processing*, Victoria, British Columbia, Canada, Oct. 2006, pp. 80–85.

[24] X. He and L. Deng, Microsoft Research, "Discriminative learning in speech recognition," Tech. Rep. MSR-TR-2007-129 [Online]. Available: http://research.microsoft.com/research/pubs/view.aspx?tr_id=1372

[25] X. Huang, A. Acero, and H. Hon, *Spoken Language Processing*. Englewood Cliffs, NJ: Prentice-Hall, 2001.

[26] T. Jebara, "Discriminative, generative, and imitative learning," Ph.D. dissertation, Media Laboratory, MIT, Cambridge, MA, Dec. 2001.

[27] T. Jebara and A. Pentland, "On reversing Jensen's inequality," in *Neural Information Processing Systems*, vol. 13, Dec. 2000, pp. 231–238.

[28] B.-H. Juang and S. Katagiri, "Discriminative learning for minimum error classification, *IEEE Trans. Signal Processing.*, vol. 40, no. 12, pp. 3043–3054, 1992.

[29] B.-H. Juang, W. Chou, and C.-H. Lee, "Minimum classification error rate methods for speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 5, May 1997, pp. 257–265.

[30] D. Kanevsky, "A generalization of the Baum algorithm to functions on nonlinear manifolds," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, 1995, pp. 473–476.

[31] S. Kapadia, V. Valtchev, and S.J. Young, "MMI training for continuous phoneme recognition on the TIMIT database," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, 1993, vol. 2, pp. 491–494.

[32] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. Int. Conf. Machine Learning*, 2001, pp. 282–289.

[33] Y. Li, L. Shapiro, and J. Bilmes, "A generative/discriminative learning algorithm for image classification," in *Proc. IEEE Int. Conf. Computer Vision*, Beijing, China, 2005, pp. 1605–1612.

[34] A. McCallum, D. Freitag, and F. Pereira, "Maximum entropy Markov models for information extraction and segmentation," in *Proc. Int. Conf. Machine Learning*, 2000, pp. 591–598.

[35] W. Macherey, L. Haferkamp, R. Schlüter, and H. Ney, "Investigations on error minimizing training criteria for discriminative training in automatic speech recognition," in *Proc. Interspeech*, Lisbon, Portugal, 2005, pp. 2133–2136.

[36] E. McDermott, "Discriminative training for automatic speech recognition using the minimum classification error framework," *Neural Information Processing System (NIPS) Workshop*, Whistler, British Columbia, Canada, Dec. 2005.

[37] E. McDermott and T. Hazen, "Minimum classification error training of landmark models for real-time continuous speech recognition," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, 2006., pp. 15–18.

[38] 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. Speech Audio Processing*, vol. 15. no. 1, 2007, pp. 203–223.

[39] A. Nádas, "A decision theoretic formulation of a training problem in speech recognition and a comparison of training by unconditional versus conditional maximum likelihood," *IEEE Trans. Acoustics, Speech Signal Processing*, pp. 814–817, 1983.

[40] A. Nádas, D. Nahamoo, and M.A. Picheny, "On a model-robust training Method for speech recognition," *IEEE Trans. Acoustics, Speech Signal Processing*, pp. 1432–1436, 1988.

[41] Y. Normandin, "Hidden Markov models, maximum mutual information estimation, and the speech recognition problem," Ph.D. dissertation, McGill Univ., Montreal, Canada, 1991.

[42] Y. Normandin, "Maximum mutual information estimation of hidden Markov models," in *Automatic Speech and Speaker Recognition*, C.-H. Lee, F.K. Soong, and K.K. Paliwal, Eds. Norwell, MA: Kluwer, 1996.

[43] F. Och, "Minimum error rate training in statistical machine translation," in *Proc. 41st Meeting Association Computational Linguistics*, 2003, pp. 160–167.

[44] F. Pereira, "Linear models for structure prediction," in *Proc. Interspeech*, Lisbon, Portugal, 2005, pp. 717–720.

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

[46] D. Povey, M.J.F. Gales, D.Y. Kim, and P.C. Woodland, "MMI-MAP and MPE-MAP for acoustic model adaptation," in *Proc. Eurospeech*, 2003, pp. 1981–1984.

[47] D. Povey, "Discriminative training for large vocabulary speech recognition," Ph.D. dissertation, Cambridge Univ., Dept. Eng., Cambridge, U.K., 2004.

[48] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig, "fMPE: Discriminatively trained features for speech recognition," in *Proc. DARPA EARS RT-04 Workshop*,Palisades, New York, 2004, pp. 1–4.

[49] C. Rathinavalu and L. Deng, "Speech trajectory discrimination using the minimum classification error learning," *IEEE Trans. Speech Audio Processing*, vol. 6, pp. 505–515, Nov. 1998.

[50] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[51] M. Riedmiller and H. Braun, "A direct adaptive method for faster back propagation learning: The RPROP algorithm," in *Proc. IEEE Int. Conf. Neural Networks*, San Francisco, CA, 1993, pp. 586–591.

[52] J. Le Roux and E. McDermott, "Optimization method for discriminative training," in *Proc. Interspeech*, Lisbon, Portugal, 2005, pp. 3341–3344.

[53] R. Schlüter, "Investigations on discriminative training criteria," Ph.D. dissertation, RWTH Aachenm Univ. Technology, Dept. Comp. Science, Aachen, Germany, 2000.

[54] R. Schlüter, W. Macherey, B. Muller, and H. Ney, "Comparison of discriminative training criteria and optimization methods for speech recognition," *Speech Commun.*, vol. 34, pp. 287–310, June 2001.

[55] S. Sun, C. Greenwood, and R. Neal, "Haplotype inference using a Bayesian hidden Markov model," *Genetic Epidemiology*, vol. 31, pp. 1–12, Dec. 2007.

[56] S. Theodoridis and K. Koutroumbas, *Pattern Recognition,* 3rd ed. New York: Academic, 2006.

[57] V. Vapnik, *Statistical Learning Theory*. New York: Wiley-Interscience, 1998.

[58] V. Valtchev, "Discriminative methods for HMM-based speech recognition, Ph.D. dissertation, Cambridge Univ., Dept. Eng., U.K., 1995.

[59] V. Valtchev, J.J. Odell, P.C. Woodland, and S.J. Young, "MMIE training of large vocabulary recognition systems," *Speech Commun.*, vol. 22, no. 4, pp. 303–314, 1997.

[60] V. Valtchev, J.J. Odell, P.C. Woodland, and S.J. Young, "Lattice-based discriminative training for large vocabulary speech recognition," in in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, 1996, pp. 605–608.

[61] P. Woodland and D. Povey, "Large scale discriminative training for speech recognition," in *Proc. ITRW ASR*, 2000, pp. 7–16.

[62] Y. Xiong, Q. Huo, and C. Chan, "A discrete contextual stochastic model for the offline recognition of handwritten Chinese characters" *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, pp. 774–782, July 2001.

[63] R. Yan, J. Zhang, J. Yang, and A. Hauptmann, "A discriminative learning framework with pairwise constraints for video object classification," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 28, no. 4, pp. 578–593, 2006.

[64] J. Yang, Y. Xu, and C.S. Chen, "Hidden Markov model approach to skill learning and its application in telerobotics," *IEEE Trans. Robotics Automat.*, vol. 10, no. 5, pp. 621–631, 1994.

[65] C. Yen, S. Kuo, and C.-H. Lee, "Minimum error rate training for PHMM-based text recognition," *IEEE Trans. Image Processing*, vol. 8, pp. 1120–1124, Aug. 1999.

[66] D. Yu, L. Deng, X. He, and A. Acero, "Large-margin minimum classification error training for large-scale speech recognition tasks," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Honolulu, HI, Apr. 2007, pp. 1137–1140.