DECODING NETWORK OPTIMIZATION USING MINIMUM TRANSITION ERROR TRAINING

Yotaro Kubo, Shinji Watanabe[1] *, Atsushi Nakamura*

NTT Communication Science Laboratories, Kyoto 619–0237, Japan

ABSTRACT

The discriminative optimization of decoding networks is important for minimizing speech recognition error. Recently, several methods have been reported that optimize decoding networks by extending weighted finite state transducer (WFST)-based decoding processes to a linear classification process. In this paper, we model decoding processes by using conditional random fields (CRFs). Since the maximum mutual information (MMI) training technique is straightforwardly applicable for CRF training, several sophisticated training methods proposed as the variants of MMI can be incorporated in our decoding network optimization. This paper adapts the boosted MMI and the differenced MMI methods for decoding network optimization so that state transition errors are minimized in WFST decoding. We evaluated the proposed methods by conducting large-vocabulary continuous speech recognition experiments. We confirmed that the CRF-based framework and transition error minimization are efficient for improving the accuracy of automatic speech recognizers.

Index Terms— Automatic speech recognition, conditional random fields, weighed finite-state transducers, transition errors

1. INTRODUCTION

Current speech recognizers involve several probabilistic models, such as acoustic models, lexicons, and language models. Therefore, the efficient integration of these models is important if we are to optimize the speech recognition performance. Conventionally, these probabilistic models are trained separately, and integrated simply by taking the product of probabilities. This basic scheme is consistent if we consider the maximum likelihood (ML) training of hidden Markov model (HMM)-based acoustic models, and N-gram language models. This is because these models are designed so that the joint objective function can be factorized into a likelihood term and a prior probability term computed by acoustic and language models, respectively.

On the other hand, discriminative techniques have been attracting attention since they can minimize the word error rates of automatic speech recognition (ASR) directly. Conventionally, discriminative training of generative models (HMMs and N-gram models) is commonly used as a realization of a discriminative technique. However, these approaches still train acoustic and language models separately. If we are to discriminatively train acoustic and language models jointly, the objective function cannot be factorized as in the ML case. This fact has motivated the speech community to develop a unified discriminative model for ASR.

Several studies have been devoted to the unified discriminative modeling. For example, conditional random fields (CRFs) are used to define the posterior probability of word sequences, given observation vectors. CRFs provides a unified way to model ASR by incorporating feature vectors that describe both linguistic and acoustic information. [1] introduced hidden CRFs that can be used to replace conventional HMM-based speech recognizers. Segmental CRFs [2] have been proposed for integrating several detectors to realize an accurate rescoring framework of speech recognition results. However, since these approaches are not based on decoder-friendly data structures, the tight integration with computationally efficient decoding techniques is not straightforwardly achieved in large-scale tasks.

Alternatively, approaches based on discriminative optimization of weighted finite state transducer (WFST)-based decoding network are advantageous in terms of computational efficiency of decoding. [3] and [4] achieved such optimization by employing the WFST-based features and the averaged perceptron (AP) approach. The AP approach is similar to the CRF approach, where the AP training method is based on the online learning technique enhanced by the trajectory averaging technique. Since the AP approach is integrated with perceptron-based training, the APs are not compatible with sophisticated training criteria based on *maximum-mutual-information* (MMI) criteria proposed for acoustic model training [5, 6]. Therefore, the use of a CRF-based framework is still important if we are to adapt these training methods to decoding network optimization because the MMI methods are also applicable for CRFs.

In this paper, to leverage both advantages of the WFST optimization methods and the CRF training methods, we propose methods for decoding network optimization that are based on MMI training of CRFs. By modeling the WFST-based decoding processes based on CRFs, the MMI-based training methods are directly applicable for decoding network optimization without sacrificing the availability of computationally efficient decoding methods. This paper first provides the CRF-based formulation of WFST decoding in Section 2. And then, in Section 3, the several training methods are proposed by deriving the MMI-based training methods proposed for acoustic model training. The experimental results are discussed in Section 4.

2. CRF-BASED FORMULATION OF WFST DECODING

Speech recognizers estimate a relevant label sequence, denoted as $\boldsymbol{\ell} = \{\ell_1, \ell_2, \cdots\}$, from a given observation vector sequence $\mathbf{X}$. Typically, the observation vector sequences consist of D-dimensional vectors as $\mathbf{X} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_t, \cdots | \boldsymbol{x}_t \in \mathbb{R}^D\}$. An output label sequence $\hat{\boldsymbol{\ell}}$ is obtained by maximizing the posterior probability function as $\hat{\boldsymbol{\ell}} = \operatorname*{argmax}_{\boldsymbol{\ell}} \log P(\boldsymbol{\ell}|\mathbf{X})$. Conventional speech recognizers model this process by introducing HMMs and the Viterbi approximation, as follows:

$$\begin{aligned}\operatorname*{argmax}_{\boldsymbol{\ell}} \log P(\boldsymbol{\ell}|\mathbf{X}) &\overset{\text{def}}{=} \operatorname*{argmax}_{\boldsymbol{\ell}} \log \sum_{\mathbf{q}} \prod_t p(\boldsymbol{x}_t|q_t) P(\mathbf{q}|\boldsymbol{\ell}) P(\boldsymbol{\ell}), \\ &\approx \operatorname*{argmax}_{\boldsymbol{\ell}} \max_{\mathbf{q}} \log \prod_t p(\boldsymbol{x}_t|q_t) P(\mathbf{q}|\boldsymbol{\ell}) P(\boldsymbol{\ell})^{\alpha},\end{aligned}$$

[1] The author is currently with Mitsubishi Electric Research Laboratories.

where α is a tunable parameter, which is called a language model scale factor, and $\mathbf{q} = \{q_1, \cdots, q_t, \cdots\}$ denotes a variable for HMM state sequences.

A weighted finite-state transducer (WFST) is a computationally efficient data structure for the above nested optimizations. The WFST is represented by a graph that has initial states and final states. In this paper, a WFST is treated as a set of graph paths that start from an initial state and reach a final state. A graph path is treated as a sequence of graph arcs (e.g. a_m) that contain information about the corresponding identifier $N[a_m] \in \{0, 1, \cdots\}$, input symbol $I[a_m] \in \{\epsilon\} \cup \{1, .., S\}$, output symbol $O[a_m] \in \{\epsilon\} \cup \mathbb{W}$, transition weight $H[a_m] \in \mathbb{R}^+$ and time stamp $T[a_m] \in \{1, .., T\}$, where $\mathbb{W}$ is a set of all words, T is the length of the observation vector sequence, and S is the number of HMM states. With this WFST notation, the time stamp $T[a_m]$ of the arc a_m is only defined when the input symbol $I[a_m]$ is not ϵ. By introducing WFSTs, the above optimization for ASR decoding is simplified to the following equivalent optimization:

$$\hat{\boldsymbol{\ell}} = \{O[\hat{a}_m] | O[\hat{a}_m] \neq \epsilon, \forall m\} \text{ where } \hat{\mathbf{a}} = \underset{\mathbf{a}\in\boldsymbol{\mathcal{D}}}{\mathrm{argmax}} P(\mathbf{a}|\mathbf{X}). \tag{1}$$

Here, $\hat{a}_m$ is the m^{th} element of the best arc sequence $\hat{\mathbf{a}}$, and $\boldsymbol{\mathcal{D}}$ is the decoding network as a set of the possible arc sequences, which is typically generated as a composition of an HMM state network $\boldsymbol{\mathcal{H}}$, a context dependent model network $\boldsymbol{\mathcal{C}}$, a lexicon network $\boldsymbol{\mathcal{L}}$, and a language model network $\boldsymbol{\mathcal{G}}$. The posterior probability of an arc sequence $\mathbf{a}$ is defined by using the total weight function $W(a_m; \mathbf{X})$, as follows:

$$P(\mathbf{a}|\mathbf{X}) \overset{\text{def}}{=} \frac{\exp\left\{\sum_m -W(a_m; \mathbf{X})\right\}}{\sum_{\mathbf{a}'\in\boldsymbol{\mathcal{D}}} \exp\left\{\sum_m -W(a'_m; \mathbf{X})\right\}}, \tag{2}$$

where a'_m is the m^{th} element of the arc sequences $\mathbf{a}'$, and $W(a_m; \mathbf{X})$ is the total weight corresponding to the arc a_m, defined by a sum of the frame score $g(a_m; \mathbf{X})$ and the transition weight $H[a_m]$, as follows:

$$\begin{aligned} W(a_m; \mathbf{X}) &= g(a_m; \mathbf{X}) + \alpha H[a_m], \\ g(a_m; \mathbf{X}) &= \begin{cases} -\log P(\boldsymbol{x}_{T[a_m]} | q_{T[a_m]} = I[a_m]) & I[a_m] \neq \epsilon, \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{3}$$

Here, the frame score $g(a_m; \mathbf{X})$ is typically computed by using a Gaussian mixture model (GMM)-based probabilistic density function $P(\boldsymbol{x}_{T[a_m]} | q_{T[a_m]} = I[a_m])$, corresponding to the $I[a_m]^{\text{th}}$ HMM state.

The above definition of arc-sequence posterior probability is a specialization of the general CRF formulation, where the CRF potential function is defined by using the weight function $W(a_m; \mathbf{X})$, and the CRF label variable is designed to denote arc sequences. Conventionally, as in Eq. (3), the potential function of this CRF is fixed by the weighted sum of GMM log-probability $g(a_m; \mathbf{X})$ and the transition weight $H[a_m]$. However, if some optimizable functions are introduced into this potential function, the WFST decoding process becomes optimizable.

In this paper, we introduce a linear term into the potential function. We define the following optimizable weight function W' and substitute W in Eq. (3) by W':

$$\begin{aligned} P(\mathbf{a}|\mathbf{X}, \boldsymbol{\Lambda}) &\overset{\text{def}}{=} \frac{\exp\left\{\sum_m -W'(a_m; \mathbf{X}, \boldsymbol{\Lambda})\right\}}{\sum_{\mathbf{a}'\in\boldsymbol{\mathcal{D}}} \exp\left\{\sum_m -W'(a'_m; \mathbf{X}, \boldsymbol{\Lambda})\right\}}, \\ W'(a_m; \mathbf{X}, \boldsymbol{\Lambda}) &= g(a_m; \mathbf{X}) + \alpha H[a_m] + \boldsymbol{\lambda}_{N[a_m]}^{\mathsf{T}} \boldsymbol{\phi}(\mathbf{X}, a_m), \end{aligned} \tag{4}$$

where $N[a_m]$ is the arc identifier, $\boldsymbol{\Lambda} \overset{\text{def}}{=} \{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots\}$ is a set of parameter vectors, and feature extraction function $\boldsymbol{\phi}(\mathbf{X}; a_m)$ is defined as follows:

$$\boldsymbol{\phi}(\mathbf{X}, a_m) \overset{\text{def}}{=} \begin{cases} [\mathbf{0}, 1]^{\mathsf{T}} & I[a_m] = \epsilon, \\ \left[\boldsymbol{\psi}(\boldsymbol{x}_{T[a_m]}), 0\right]^{\mathsf{T}} & \text{otherwise.} \end{cases} \tag{5}$$

Here, we consider two kinds of features; the frame-level acoustic feature $\boldsymbol{\psi}(\boldsymbol{x}_{T[a_m]})$ and the epsilon-arc occupancy feature, which is 0 when the arc corresponds to the specific acoustic features, and 1 otherwise.

3. TRAINING CRITERIA OF DECODING NETWORK

In this section, we describe the training methods of the abovementioned CRFs. Hereafter, we assume that each observation sequence $\mathbf{X}^{(i)}$ in the given training data set $\{\mathbf{X}^{(i)} | \forall i\}$ has a corresponding *correct* reference arc sequence $\mathbf{a}^{(i)} \in \boldsymbol{\mathcal{D}}$ that is obtained by using the transcribed word sequence. Specifically, in the experiments, reference arc sequences are obtained by performing a decoding process, as follows:

$$\mathbf{a}^{(i)} = \underset{\mathbf{a}'\in(\boldsymbol{\mathcal{D}}\circ\boldsymbol{\mathcal{W}}^{(i)})}{\mathrm{argmax}} P(\mathbf{a}'|\mathbf{X}^{(i)}, \boldsymbol{\Lambda} = \mathbf{0}), \tag{6}$$

where $\boldsymbol{\mathcal{W}}^{(i)}$ is a finite state acceptor (FSA) that accepts the i^{th} transcribed word sequence, and $\circ$ is the composition operator of the WFSTs. We emphasize that the training methods proposed in this section optimize the additional parameters of the WFST decoding networks $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots\}$ (in Eq. (4)) that are not acoustic and/or language model parameters.

Maximum mutual information

The maximum mutual information (MMI) criterion is regarded as a natural training criterion for posterior probability models including CRFs. The MMI criterion maximizes the empirical posterior probability computed by using the given training dataset, as follows:

$$\begin{aligned} \hat{\boldsymbol{\Lambda}} &= \underset{\boldsymbol{\Lambda}}{\mathrm{argmax}} \sum_i \log P(\mathbf{a}^{(i)} | \mathbf{X}^{(i)}, \boldsymbol{\Lambda}), \\ &= \underset{\boldsymbol{\Lambda}}{\mathrm{argmax}} \sum_i \log \frac{\exp\left\{\sum_m -W'(a_m^{(i)}; \mathbf{X}^{(i)}, \boldsymbol{\Lambda})\right\}}{\sum_{\mathbf{a}'\in\boldsymbol{\mathcal{D}}} \exp\left\{\sum_m -W'(a'_m; \mathbf{X}^{(i)}, \boldsymbol{\Lambda})\right\}}. \end{aligned} \tag{7}$$

Here, the summation over all possible arc sequences ($\mathbf{a}' \in \boldsymbol{\mathcal{D}}$) is usually approximated by using a set of arc sequences in the decoding lattice ($\mathbf{a}' \in \boldsymbol{\mathcal{L}}^{(i)}$). By introducing the lattice approximation, the gradient vector of this objective function is obtained as follows:

$$\begin{aligned} \nabla_\lambda F^{\text{MMI}}(\boldsymbol{\Lambda}') &= \sum_i \boldsymbol{\xi}_i^{\text{num}}(\boldsymbol{\Lambda}') - \boldsymbol{\xi}_i^{\text{den}}(\boldsymbol{\Lambda}') \\ \boldsymbol{\xi}_i^{\text{num}}(\boldsymbol{\Lambda}') &= \sum_m \boldsymbol{\phi}(\mathbf{X}^{(i)}, a_m^{(i)}) \\ \boldsymbol{\xi}_i^{\text{den}}(\boldsymbol{\Lambda}') &= \sum_{\mathbf{a}'\in\boldsymbol{\mathcal{L}}^{(i)}} P(\mathbf{a}'|\mathbf{X}^{(i)}, \boldsymbol{\Lambda}')^\kappa \sum_m \boldsymbol{\phi}(\mathbf{X}^{(i)}, a'_m) \end{aligned} \tag{8}$$

where κ is a lattice smoothing factor. This gradient vector can easily be computed by performing the forward-backward algorithm. In contrast to the AP approach, the MMI objective function and its gradient vector consider all competing hypotheses. In the following sections, we denote this training criterion as "MMI-CRF."

Introducing transition error measurement

Since MMI estimates parameters to maximize the posterior probabilities of correct arc sequences in the training dataset, the training procedure only takes account of sequence-level errors. However, since most ASR applications require improved accuracy in terms of word error rates (WERs), the sequence-level improvements are not sufficient. For acoustic model training, discriminative training techniques are proposed that aim at reducing fine-grained errors. The use of fine-grained error measurements for training criteria is known to be important not only for the consistency between the training criteria and the application requirements, but also for the robustness of the trained speech recognizers.

In this paper, we focus on the transition error measure of the decoding that counts the number of error arc transitions. To reduce frame-level errors, we extended the boosted MMI technique [5] for our CRF training. The boosted MMI, which was originally developed for discriminative training of acoustic models to reduce phoneme (or word) errors, achieves error reduction by emphasizing importance of erroneous sequences that include a lot of phoneme errors. In this paper, to reduce transition errors, we defined the following objective function:

$$\underbrace{\sum_i \log \frac{\exp\left\{\sum_m -W'(a_m^{(i)};\mathbf{X}^{(i)},\mathbf{\Lambda})\right\}}{\sum_{\mathbf{a}'\in\boldsymbol{\mathcal{D}}}\exp\left\{\sum_m -W'(a'_m;\mathbf{X}^{(i)},\mathbf{\Lambda})+\sigma E(\mathbf{a}^{(i)},\mathbf{a}')\right\}}}_{F_\sigma^{\mathrm{bMMI}}(\Lambda)}, \tag{9}$$

where σ is a tunable parameter that adjusts the importance of fine-grained errors, and $E(\mathbf{a}^{(i)},\mathbf{a}')$ is an transition error count function, defined as follows:

$$E(\mathbf{a}^{(i)},\mathbf{a}') = \sum_t \left(1-\delta(n(\mathbf{a}^{(i)},t);n(\mathbf{a}',t))\right), \tag{10}$$

where $n(\mathbf{a},t)$ is the arc identifier $N[a_m]$ of an arc $a_m \in \mathbf{a}$ that satisfies $T[a_m]=t$, $\delta(x;y)$ is Kronecker's delta, i.e. $\delta(x;y)=1$ if $x=y$ otherwise $\delta(x;y)=0$.

The gradient vector corresponding to the objective function (Eq. (9)) can also be computed by using lattice-based forward-backward computation [5]. In the following sections, we denote this training criterion as "bMMI-CRF."

Direct minimization of transition error count

Although bMMI-CRF takes account of the transition error measurement by emphasizing the importance of erroneous sequences, the number of errors is not minimized directly. Several training criteria that directly minimize the number of errors are proposed in the field related to the discriminative training of acoustic models, such as *minimum classification error* (MCE) [7], and *minimum phone error* (MPE) [8].

By introducing the transition error counts into the MPE objective function, we derive the following objective function:

$$\underbrace{\sum_i \frac{-\sum_{\mathbf{a}'\in\boldsymbol{\mathcal{D}}}\exp\left\{\sum_m -W'(a'_m;\mathbf{X}^{(i)},\mathbf{\Lambda})\right\}E(\mathbf{a}^{(i)},\mathbf{a}')}{\sum_{\mathbf{a}'\in\boldsymbol{\mathcal{D}}}\exp\left\{\sum_m -W'(a'_m;\mathbf{X}^{(i)},\mathbf{\Lambda})\right\}}}_{F^{\mathrm{MPE}}(\Lambda)}, \tag{11}$$

In this paper, we introduce a variant of the MPE criterion, called differenced MMI (dMMI), proposed for acoustic model training [6]. The dMMI criterion is based on the following identity:

$$F^{\mathrm{MPE}}(\Lambda) = \frac{\partial}{\partial\sigma}F_\sigma^{\mathrm{bMMI}}(\Lambda)\mid_{\sigma=0} \tag{12}$$

This identity suggests a variant of the MPE that is obtained by substituting the partial derivative in the above identity by a numerical differentiation, as follows:

$$F_{(\sigma_1,\sigma_2)}^{\mathrm{dMMI}}(\Lambda) = \frac{1}{\sigma_2-\sigma_1}\left(F_{\sigma=\sigma_2}^{\mathrm{bMMI}}(\Lambda) - F_{\sigma=\sigma_1}^{\mathrm{bMMI}}(\Lambda)\right). \tag{13}$$

This objective function $F_{(\sigma_1,\sigma_2)}^{\mathrm{dMMI}}(\Lambda)$ converges to the MPE objective function in the limit of $\sigma_1 \to -0, \sigma_2 \to +0$. Furthermore, this objective function also converges to the bMMI objective function $F_\sigma^{\mathrm{bMMI}}(\Lambda)$ in the limit of $\sigma_1 \to -\infty, \sigma_2 \to \sigma$. Thus, we can define intermediates between MPE and bMMI by using dMMI criteria. In the following sections, we denote this training criterion as "dMMI-CRF."

4. EXPERIMENTS

We conducted the speech recognition experiments to evaluate the efficiency of our proposed approach. We applied the proposed method to an MIT-OCW/World lecture recognition task [9], and evaluated the word error rates. Moreover, to ensure a stability, this approach is also validated by using the WSJ task (WSJ20k). The task descriptions are summarized in Table 1.

In the experiments, 12 Mel-frequency cepstral coefficients (MFCCs) and logarithmic energy were augmented by their derivatives and accelerations, and used as observation vectors for both the HMM-based acoustic models and the proposed CRFs. The numbers of clustered HMM states were 2,565 and 2,466 for the MIT-OCW and the WSJ20k tasks, respectively, and the number of mixture components was set at 32 for both tasks. The language model scale factor α was determined by using a development dataset and a speech recognizer without decoding network optimization. The lattice smoothing factor κ was simply fixed at 1.0.

In these experiments, we used a simple frame-level acoustic feature function, defined as $\boldsymbol{\psi}(\boldsymbol{x}) \overset{\mathrm{def}}{=} [\boldsymbol{x}^\mathsf{T},1]^\mathsf{T}$, i.e. we used the raw observation vectors augmented by a bias term. To obtain the arc identifier $N[a_m]$, we numbered the arcs in the WFST $\boldsymbol{\mathcal{D}}^1 \overset{\mathrm{def}}{=} \boldsymbol{\mathcal{H}}\circ\boldsymbol{\mathcal{C}}\circ\boldsymbol{\mathcal{L}}\circ\boldsymbol{\mathcal{G}}^1$, which we created by composing an HMM WFST $\boldsymbol{\mathcal{H}}$, a context WFST $\boldsymbol{\mathcal{C}}$, a lexicon WFST $\boldsymbol{\mathcal{L}}$, and a unigram WFST $\boldsymbol{\mathcal{G}}^1$. We then composed the decoding WFST as $\boldsymbol{\mathcal{D}} \overset{\mathrm{def}}{=} \boldsymbol{\mathcal{D}}^1\circ\boldsymbol{\mathcal{G}}^{(2,3)}$, where $\boldsymbol{\mathcal{G}}^{(2,3)}$ denotes a trigram WFST, by using the on-the-fly composition algorithm [10]. The arc identifier is $N[a_m]$ is taken from the arc number annotated to the corresponding arcs in the first WFST $\boldsymbol{\mathcal{D}}^1$. We used Rprop [11] for the optimization of the parameter vectors $\mathbf{\Lambda}$. The utterance-level cepstral mean normalization technique was used for normalizing the training and test dataset. The standard derivations of each variable in the observation vectors were estimated from the training dataset, and all the training data and test data were normalized according to the estimated standard deviations.

For the training we used 12 nodes (100 threads) of computers driven by an MPI-based program. The computational time required for each iteration in MIT-OCW tasks was approximately 2 minutes where the previous AP-based approach requires 5 h for each iteration [3]. This computational efficiency results from the fact that the proposed training methods are based on lattice-based computation where AP-based approach requires decoding over full recognition networks.

Table 2 summarize the word error rates of the CRF method (MMI-CRF) and the compared methods. First, we confirmed that decoding network optimization techniques ("dMMI-HMM + AP" and "dMMI-HMM + CRF") achieved better results than the state-of-the-art methods for discriminative training, such as MPE and

Table 1. Task descriptions

		Training	**Evaluation**
MIT-OCW	# utterances	60,392 (101 h)	6,989 (7.8 h)
	# vocabulary	44,485	
WSJ20k	# utterances	37,513 (73.6 h) (SI284)	430 (0.4 h) (Nov'93)
	# vocabulary	19,982	

Table 2. Word error rates of WFST-based CRF and perceptron

Method	**MIT-OCW** **WER [%]**	**WSJ20k** **WER [%]**
ML-HMM	32.8	7.8
MPE-HMM	28.3	–
bMMI-HMM [6]	28.3	–
dMMI-HMM [6]	28.2	7.7
dMMI-HMM + AP [3]	27.8	–
dMMI-HMM + MMI-CRF	27.7	7.6

dMMI. We consider that this improvement arises from the fine modeling of long context information. Since these WFST-based methods involve parameters for each WFST arc, these models can be considered intermediates between whole-word models and phoneme-level models. Although the use of whole-word HMMs is prohibitive in large vocabulary tasks, the proposed method enables handling of word-level information by exploiting a computationally efficient representation obtained by WFST minimization. Moreover, we also confirmed that the results of CRFs were comparable to those of the AP-based method. This result is consistent because the MMI training criterion and the AP training criterion are similar. The slight improvements (0.1% in absolute WER) might be due to the use of lattices where the AP-based approach only uses a one-best competitor sequence.

Table 3 shows the results of the CRF-based systems trained using criteria with transition error measurements. We confirmed that introducing transition error was efficient for reducing word errors. Furthermore, we confirmed that bMMI training was more efficient than dMMI training. This tendency is somewhat different from the experimental results obtained for the discriminative training of acoustic models (cf. Table 2). We speculate that this was because of overfitting. In fact, the number of frame errors over the training dataset was reduced by introducing the dMMI technique. Further, we confirmed that the error rates of dMMI converged to those of bMMI by setting $\sigma_1 \ll 0$. In the WSJ20k tasks, where the performance is almost saturated by the ML baseline, we observed the same tendency, as in the MIT-OCW case. Thus, the proposed method could be considered efficient for various ASR tasks.

5. CONCLUSIONS

This paper extends the discriminative optimization approach based on the averaged perceptron technique used for WFST-based decoding to a conditional random field (CRF) approach. The proposed CRFs and the previous approach enable optimization of the WFST by extending the decoding score function, which was originally defined as the sum of logarithmic likelihoods of acoustic models and language models, by introducing linear optimizable terms. Thanks to the development of CRF training methods, the proposed speech recognizer can be trained using several sophisticated techniques. We evaluated the performance of CRFs trained using original maximum mutual information (MMI), boosted MMI, and differenced MMI training. We found that modifying the optimization criteria for training by introducing a fine-grained error measurements is an efficient way to reduce word error rates of speech recognizers.

Table 3. Word error rates of CRFs trained with several criteria

Method	σ (bMMI) (σ_1, σ_2) (dMMI)	**MIT-OCW** **WER [%]**	**WSJ20k** **WER [%]**
+ MMI-CRF	–	27.7	7.6
+ bMMI-CRF	1.0	27.4	7.6
	2.0	27.1	7.4
	4.0	27.4	7.3
+ dMMI-CRF	(-1.0, 1.0)	27.8	7.4
	(-2.0, 2.0)	27.3	7.4
	(-4.0, 4.0)	27.6	7.3
+ dMMI-CRF	(-0.25, 0.25)	27.9	–
(→ MPE)	(-0.0625, 0.0625)	28.0	–
+ dMMI-CRF	(-10.0, 2.0)	27.3	–
(→ bMMI)	(-50.0, 2.0)	27.1	–

Future work will include feature augmentation. In this paper, we focused solely on the linear classification of Mel-frequency cepstral coefficients (MFCC)-based features. However, since CRFs cannot handle nonlinearity in a feature distribution, an explicit definition of the nonlinear warped features is important. Thus, enhancing features would be an important extension.

Acknowledgments

We thank the MIT Spoken Language Systems Group for helping us to perform experiments based on the MIT-OCW dataset. We also thank Yasuhisa Fujii at Toyohashi University of Technology for helping us to set up the WSJ evaluation environment.

6. REFERENCES

[1] A. Gunawardana, M. Mahajan, A. Acero, and J. C. Platt, "Hidden conditional random fields for phone classification," in *Proc. INTERSPEECH*, 2005.

[2] G. Zweig and P. Nguyen, "A segmental CRF approach to large vocabulary continuous speech recognition," in *Proc. IEEE ASRU*, 2009, pp. 152–157.

[3] S. Watanabe, T. Hori, and A. Nakamura, "Large vocabulary continuous speech recognition using WFST-based linear classifier for structured data," in *Proc. INTERSPEECH*, Aug. 2010, pp. 346–349.

[4] M. Lehr and I. Shafran, "Learning a discriminative weighted finite state transducer for automatic speech recognition," *IEEE Trans. ASLP*, vol. 19, pp. 1360–1367, 2011.

[5] D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariah, "Boosted MMI for model and feature-space discriminative training," in *Proc. ICASSP-2008*, 2008, pp. 4057–4060.

[6] E. McDermott, S. Watanabe, and N. Atsushi, "Discriminative training based on an integrated view of MPE and MMI in margin and error space," in *Proc. ICASSP-2010*, 2010.

[7] E. McDermott and S. Katagiri, "String-level MCE for continuous phoneme recognition," in *Proc. EUROSPEECH*, 1997.

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

[9] J. Glass, T. J. Hazen, S. Cyphers, I. Malioutov, D. Huynh, and R. Barzilay, "Recent progress in the MIT Spoken Lecture Processing Project," in *Proc. INTERSPEECH*, 2007, pp. 2553–2556.

[10] T. Hori, C. Hori, Y. Minami, and A. Nakamura, "Efficient WFST-based one-pass decoding with on-the-fly hypothesis rescoring in extremely large vocabulary continuous speech recognition," *IEEE Trans. ASLP*, vol. 15, pp. 1352–1365, 2007.

[11] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The Rprop algorithm," in *Proc. IEEE ICNN*, 1993, pp. 586–591.