Hidden Boosted MMI and Hierarchical State Posterior Feature for Automatic Speech Recognition based on Hidden Conditional Neural Fields

Yasuhisa Fujii, Kazumasa Yamamoto, Seiichi Nakagawa

Department of Computer Science and Engineering, Toyohashi University of Technology, Japan
{fujii,kyama,nakagawa}@slp.cs.tut.ac.jp

Abstract

We have investigated automatic speech recognition using Hidden Conditional Neural Fields (HCNF). In this paper, we propose a new objective function, Hidden Boosted MMI (HB-MMI) that considers the number of errors in the training data even if the correct state sequence is not known for training the HCNF. The experimental results show that HB-MMI can improve recognition accuracy if overfitting does not occur. We also present an automatic speech recognition method using a hierarchical state posterior feature where the output from the first stage HCNF is used as input for the second stage HCNF. The experimental results show that the feature improves recognition accuracy. By combining both of the proposed methods, we obtain further improvements.

Index Terms: hidden conditional neural fields, automatic speech recognition, hidden boosted MMI, state posterior feature

1. introduction

Current ASR systems employ a Hidden Markov Model (HMM) together with a Gaussian Mixture Model (GMM) as the emission probability for an acoustic model. However, the HMM has two major drawbacks for use as an acoustic model. First, it relies on a strong independency assumption whereby frames are independent in a given state, and thus, lacks the ability to deal with features that straddle several frames. Second, because the HMM is a generative model, it is not suitable for discriminating sequences. To solve the former problem, features that can deal with phenomena straddling multiple frames have been developed, such as the delta coefficient [1], segmental statistics [2], and modulation spectrum [3]. For the latter problem, discriminative training methods such as the minimum phone error (MPE) have been investigated [4].

We have proposed an ASR method using Hidden Conditional Neural Fields (HCNF) to overcome the above two problems [5]. HCNF can handle features that straddle several frames and has high discriminative power since it is a discriminative model. Experimental results on the TIMIT corpus showed the effectiveness of HCNF for ASR in a continuous phoneme recognition experiment [5].

In [5], we employed a training criterion based on posterior probability maximization, which corresponds to the MMI criterion in the context of HMM based acoustic model training. However, as is well known, improving the posterior probability does not necessarily improve the word error rate (WER). Therefore, a training criterion that considers the error rate is preferable to a straightforward posterior probability maximization.

Among several such criteria, Boosted MMI is a promising criterion due to its effectiveness and ease of implementation [6]. The same criterion for log-linear models (Hidden Conditional Random Fields, HCRF) is discussed in [7]. Although these approaches seem effective for HCNF training, they cannot be applied to HCNF directly because state sequences are not observable (i.e., they are hidden) in HCNF.

In this paper, we propose Hidden Boosted MMI (HB-MMI) as a new training criterion for HCNF. HB-MMI considers training errors more directly than posterior maximization even if state sequences are not known (fixed). Instead of taking forced-alignment and using fixed alignment, we compute the expected error counts from the posteriors of reference state sequences and all hypotheses state sequences.

In addition, we propose using a hierarchical state posterior feature to further improve HCNF based ASR. Posterior features are robust to context, speaker, and noise variabilities compared with raw acoustic features such as MFCC and PLP, and accordingly have shown improved recognition accuracy in several studies [8, 9, 10]. In this paper, we borrow the idea used in [10], except we use HCNF instead of MLP.

This paper is organized as follows. In the next section, we explain HCNF for ASR and in Section 3, HB-MMI for HCNF training. In Section 4, the hierarchical state posterior feature is introduced. The experimental setup and results are given in Section 5. Finally, we present our conclusions and some future works in Section 6.

2. Hidden Conditional Neural Fields

2.1. Formulation

Fig. 1 shows the structure of HCNF. Given an observation sequence $X = (x_1, x_2, \dots, x_T)$, HCNF computes a score of a label sequence $Y = (y_1, y_2, \dots, y_T)$ as follows:

$$P(Y|X) = \frac{\sum_S \exp(\kappa(\Phi_n(X,Y,S) + \Psi_n(X,Y,S)))}{Z(X)}, \quad (1)$$

where $S = (s_1, s_2, \dots, s_T)$ is a hidden variable sequence that represents a state sequence, κ is a state-flattening coefficient [11] and $Z(X)$ is a partition function computed as

$$Z(X) = \sum_{Y'} \sum_{S} \exp(\kappa(\Phi_n(X,Y',S) + \Psi_n(X,Y',S))), \quad (2)$$

$\Phi_n(X,Y,S)$ and $\Psi_n(X,Y,S)$ are an observation function and a transition function, respectively and are given below.

$$\Phi_n(X,Y,S) = \sum_t \sum_g^K w_{y_t,s_t,g} h(\theta_{y_t,s_t,g}^T \phi(X,Y,S,t)), \quad (3)$$

$$\Psi_n(X,Y,S) = \sum_j u_j \sum_t \psi_j(X,Y,S,t,t-1), \quad (4)$$

where $\psi_j(X,Y,S,t,t-1)$ is a transition feature extracted at frame t and $t-1$, and u_j is a corresponding weight, $w_{y,s,g}$ is a weight specific to the triple y, s, and g, $\phi(X,Y,S,t)$ is a vector representation of features such as MFCCs, $\theta_{y,s,g}$ is the corresponding weight vector specific to the triple y, s, and g [1], and $h(x)$ is a gate function defined as

[1] By this definition, the gate functions are dependent on their states, unlike in the original CNF [12]. We used this definition as it was robust in our initial experiments. However, we can obtain the state independent gate functions by setting $\theta_{y_t,s_t,g} \triangleq \theta_g$.

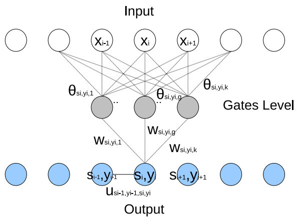

Figure 1: *Structure of HCNF.*

$$h(x) = \frac{c}{1 + \exp(-\alpha(x - \beta))} - d, \tag{5}$$

where c and d are terms to change the range of values, and α and β are terms to control the shape of the gate function. In HCNF, the observation function $\Phi_n(X, Y, S)$ uses K gate functions through which it considers non-linearity among the features. Because the feature functions used in HCNF are restricted to the current frame in the observation function, and the current and previous frames in the transition function, efficient algorithms such as the forward-backward algorithm and Viterbi algorithm can also be used in HCNF.

2.2. Training

The objective function for HCNF is defined as

$$l(\lambda; D) = -\sum_i \log P(Y^i|X^i). \tag{6}$$

The partial derivatives of (6) with respect to $w_{y,s,g}$ and $\theta_{y,s,g}$ can be computed as follows:

$$\begin{aligned}\frac{\partial l(\lambda; D)}{\partial w_{y,s,g}} =& -\kappa \sum_i E\left[\sum_t h(\theta_{y,s,g}^T \phi(X^i, Y^i, S, t))\right]_{S|X^i,Y^i} \\ &+ \kappa \sum_i E\left[\sum_t h(\theta_{y,s,g}^T \phi(X^i, Y, S, t))\right]_{Y,S|X^i}, \end{aligned}\tag{7}$$

$$\begin{aligned}&\frac{\partial l(\lambda; D)}{\partial \theta_{y,s,g}} = \\ &- \kappa \sum_i E\left[\sum_t w_{y,s,g} \frac{\partial h(\theta_{y,s,g}^T \phi(X^i, Y^i, S, t))}{\partial \theta_{y,s,g}}\right]_{S|X^i,Y^i} \\ &+ \kappa \sum_i E\left[\sum_t w_{y,s,g} \frac{\partial h(\theta_{y,s,g}^T \phi(X^i, Y, S, t))}{\partial \theta_{y,s,g}}\right]_{Y,S|X^i}. \end{aligned}\tag{8}$$

$$\begin{aligned}\frac{\partial l(\lambda; D)}{\partial u_j} =& -\kappa \sum_i E\left[\sum_t \psi_k(X^i, Y^i, S, t, t-1)\right]_{S|Y^i,X^i} \\ &+ \kappa \sum_i E\left[\sum_t \psi_k(X^i, Y, S, t, t-1)\right]_{Y,S|X^i}, \end{aligned}\tag{9}$$

The derivative of (5) is computed as

$$\frac{dh(x)}{dx} = \frac{\alpha}{c}(d + h(x))(c - d - h(x)). \tag{10}$$

In HCNF training, all parameters are trained together without distinction of acoustic and linguistic features, unlike in traditional HMM and N-gram training.

2.3. Inference

We used Viterbi algorithm to find the most likely sequence of hidden states instead of searching for the most likely output sequence that maximizes (1).

3. Hidden Boosted MMI

In this section, we propose Hidden Boosted MMI (HB-MMI) as a new training criterion for HCNF. HB-MMIM considers training errors in a more direct way than posterior maximization even if state sequences are not known (fixed). Instead of taking a forced-alignment and using fixed alignment, we compute expected error counts from the posteriors of reference state sequences and all hypotheses state sequences.

The objective function for HB-MMI is defined as follows:

$$\begin{aligned}&\ell_{HB-MMI}(\lambda; D) = \\ &-\sum_i \log \frac{\sum_S \exp(\kappa(\Lambda(X, Y, S)))}{\sum_{Y'} \sum_S \exp(\kappa(\Lambda(X, Y, S))) \exp(-b\mathrm{Acc}(S, D^i))}, \end{aligned}\tag{11}$$

$$\Lambda(X, Y, S) = \Phi_n(X, Y, S) + \Psi_n(X, Y, S), \tag{12}$$

where $\mathrm{Acc}(S, D^i)$ is the expected correct state count of a state sequence S given a reference $D^i = (X^i, Y^i)$. To compute $\mathrm{Acc}(S, D^i)$, we first compute the correct state count of state sequence S given the reference state sequence S' as follows:

$$\mathrm{Acc}(S, S') = \sum_t \delta(s_t, s'_t), \tag{13}$$

then, $\mathrm{Acc}(S, D^i)$ is computed as follows:

$$\begin{aligned}\mathrm{Acc}(S, D^i) &= \sum_{S'} P(S'|Y^i, X^i)\mathrm{Acc}(S, S'), \\ &= \sum_{S'} P(S'|Y^i, X^i) \sum_t \delta(s_t, s'_t), \\ &= \sum_t \sum_{S'} P(S'|Y^i, X^i)\delta(s_t, s'_t), \\ &= \sum_t \gamma^i(s_t, t), \end{aligned}\tag{14}$$

where $\gamma^i(s, t)$ is the posterior probability of state s at frame t given reference D^i,

$$\gamma^i(s, t) = \sum_{S'} P(S'|Y^i, X^i)\delta(s, s'_t). \tag{15}$$

In other words, $\mathrm{Acc}(S, D^i)$ is defined as the sum of the expected accuracy rates of state s_t at frame t. If $\mathrm{Acc}(S, D^i)$ is regarded as a constant, the partial derivative of (11) can be computed by (7)-(9). Since (14) is computed as the sum of local values at frame i, the expectations needed to compute the partial derivatives can easily be computed by merely adding $-b\gamma^i(s_t, t)$ to the score of state s_t at frame t, without changing the original algorithm for posterior maximization.

This kind of objective function is closely related to margin maximization [7]. Term b in (11) controls how strongly the margin is imposed in HB-MMI.

4. Hierarchical State Posterior Feature

In this section, we propose using a hierarchical state posterior feature to further improve HCNF based ASR. Posterior features are robust to context, speaker, and noise variabilities compared with raw acoustic features such as MFCC and PLP, and can provide information to detect phonemes that tend to be misrecognized as other phonemes given long-range posterior features. As such, they have improved the recognition accuracy in several studies [8, 9, 10].

In [10], a hierarchical phoneme posterior feature extracted by the first MLP and used as a feature for the second MLP is explored and the effectiveness thereof is shown. In this paper, we borrow the idea used in [10], except we use HCNF instead of MLP. That is, the output from the first HCNF is used as input for the second HCNF. By using HCNF, we obviate the need for time alignment, which is required by MLPs.

We use the state posteriors at each frame as the output from the first HCNF and the input for the second HCNF. This we call the hierarchical state posterior feature. The hierarchical posterior feature $\gamma(s,t|X)$ of state s at frame t given an input sequence X is defined as

$$\gamma(s,t|X) = \sum_{Y'}\sum_{S'} P(Y',S'|X)\delta(s,s'_t). \tag{16}$$

5. Experiment

5.1. Setup

We used the TIMIT corpus to examine the effectiveness of our proposed method because it offers a good test bed to study algorithmic improvements [13]. The training set in the TIMIT corpus consists of 3696 utterances by 462 speakers ($\approx 3h$). For the evaluation, we used the core test set consisting of 192 utterances by 24 speakers. We also used the ASJ+JNAS corpus[2], which is about 11 times larger than the TIMIT corpus. The training set in the ASJ+JNAS corpus consists of 20337 utterances by 133 speakers ($\approx 33h$). For evaluation, we used the IPA100 test set consisting of 100 utterances by 23 speakers. We extracted 13 MFCC features together with their deltas and double deltas to form a 39-dimensional observation for the TIMIT corpus. Log energy was used instead of the 0-th MFCC for the ASJ+JNAS corpus. The speech was analyzed using a 25 ms Hamming window with a pre-emphasis coefficient of 0.97 and shifted with a 10 ms fixed frame advance. For the TIMIT corpus, the 61 TIMIT phonemes were mapped into 48 phonemes for training and further collapsed from 48 phonemes to 39 phonemes for evaluation [14]. For the ASJ+JNAS corpus, 43 Japanese phonemes were used. All phonemes were represented as 3 state left-to-right monophone models. We defined four observation functions as given below:

$$\phi^{M1}_{sb}(X,Y,S,t) = \delta(s_t = s)x_{d,t+f}, \tag{17}$$

$$\phi^{M2}_{sb}(X,Y,S,t) = \delta(s_t = s)x^2_{d,t+f}, \tag{18}$$

$$\phi^{Occ}_{s}(X,Y,S,t) = \delta(s_t = s), \tag{19}$$

$$\phi^{Uni}_{y}(X,y,s,t) = \delta(y_t = y), \tag{20}$$

where f is used to consider the surrounding frames. In the experiments, we used $-4 \leq f \leq 4$, which means we used the number of observations in the nine frames centered at the current frame. In the experiments considering the hierarchical state posterior feature, a window of $-11 \leq f \leq 11$ (23 frames) was also examined, as in [10]. Moreover, we defined the following transition function:

$$\psi^{Tr}_{ss'}(X,Y,S,t,t-1) = \delta(s_t = s)\delta(s_{t-1} = s'), \tag{21}$$

$$\psi^{Bi}_{yy'}(X,y,y',s,s',t,t-1) = \delta(y_t = y)\delta(y_{t-1} = y'). \tag{22}$$

The M1 and M2 features were normalized to have zero mean and unit variance. All parameters of the HCNF were randomly initialized between -0.5 and 0.5. The state-flattening parameter κ was set to 0.1. The number of gates was set to four when using delta features and to 16 when not using delta features. When using hierarchical state posterior features, the second HCNF employed four gates per state. The parameters of the gate function were set to $\alpha = 0.1,\ \ \beta = 0.0,\ \ c = 6.0,$ and $d = 3.0$. When hierarchical state posterior features were used, $\alpha = 1.0$ was used. The parameters for the HCNF were trained by 30 iterations for the TIMIT corpus and 15 iterations for the ASJ+JNAS corpus by Stochastic Gradient Descent with L2 regularization [5]. Regularization parameter C [5] was set to 1.0.

For comparison, we conducted recognition experiments using HMMs with the same topology as the HCNFs (monophone). Initially, the HMM with diagonal covariance matrices and a 32 mixture GMM was trained in an MLE manner using HTK. The MLE-HMM model was used to train MMI- and MPE- HMMs with I-smoothing set to 100, a learning rate parameter of 2 and a scaling factor of 0.2 (similar to the state-flattening coefficient used in HCNFs), and the parameters were updated 10 times. A bigram phone language model was trained from the training corpora.

5.2. Results of training with HB-MMI

Table 1 gives the phoneme recognition results on the TIMIT core test set with several values of b in (11) and using the delta features. Since b is a term that determines how strongly the margin is imposed in HB-MMI, the larger b is, the more strict is the algorithm in terms of training error. In Table 1, the results for $b = 0$, which corresponds to the MMI criterion, that is, when HB-MMI is not used, are the best. The results for $b = 0$ are the closest to those given in [5], despite the parameters for the gate functions being different. Although we observed a reduced training error with a larger b, it appears to be just overfitting.

Table 2 shows the phoneme recognition results on the TIMIT core test set when delta features are not used. In this case, in contrast to that where delta features are used, HB-MMI is effective and improves the recognition results. The best result in Table 2 occurs with $b = 3.0$, which is comparable to the best result in Table 1 where delta features are used. This result shows that HCNF is able to extract discriminative features by considering training errors explicitly using HB-MMI even if delta features are not used. Interestingly, the PERs on training data in Table 2 are much higher than the PERs on training data in Table 1. These results indicate the models trained without delta features are more robust than the models trained with delta features.

Table 1: *Phoneme recognition results with delta features on TIMIT core test set [%]. PER means Phoneme Error Rate. Train-PER means PER on training data (48 phonemes).*

b	Del	Ins	Subs	PER	Train-PER
0.0 (MMI)	7.7	2.1	18.2	28.0	16.1
1.0	7.1	2.5	18.9	28.5	13.8
3.0	6.6	3.1	19.5	29.2	11.9
5.0	6.0	3.5	19.7	29.1	11.4
HMM (MPE)	9.1	2.2	17.1	28.4	18.4

Table 2: *Phoneme recognition results without delta features on TIMIT core test set [%].*

b	Del	Ins	Subs	PER	Train-PER
0.0 (MMI)	11.9	1.2	17.7	30.9	28.7
1.0	10.3	1.5	17.6	29.4	26.8
3.0	8.7	1.9	17.3	27.9	24.8
5.0	8.3	2.5	18.1	28.8	23.9
10.0	7.6	2.9	18.5	29.1	23.9

[2] http://www.mibel.cs.tsukuba.ac.jp/_090624/jnas/instruct.html

To confirm that no effect of HB-MMI with delta features was caused by overfitting, we conducted experiments on the ASJ+JNAS corpus, which is about 11 times larger than the TIMIT corpus. Table 3 gives the recognition results on this corpus. In the experiment, HB-MMI was clearly effective and improved the recognition results. Also, HCNF trained by the HB-MMI criterion clearly outperformed the HMM trained by MPE criterion. From these results, we conclude that HB-MMI would be effective, particularly if overfitting is not an issue.

Table 3: *Phoneme recognition results with delta features on IPA100 test set [%].*

b	Del	Ins	Subs	PER	Train-PER
0.0 (MMI)	6.2	0.8	8.7	15.7	14.3
1.0	5.0	1.0	8.6	14.6	13.0
3.0	4.5	1.3	8.1	13.9	12.8
HMM (MLE)	6.4	1.2	11.4	19.0	18.7
HMM (MMI)	5.0	1.2	9.5	15.8	15.6
HMM (MPE)	5.5	0.9	8.6	15.0	13.2

5.3. Investigation of Hierarchical State Posterior Feature

We also conducted experiments using the hierarchical state posterior feature, extracted according to the results in Section 5.2. The recognition results for $b = 0.0$ in Table 1 were applied when delta features were used in the first HCNF, whereas recognition results for $b = 3.0$ in Table 2 were applied when not using delta features in the first HCNF. Tables 4 and 5 give the results with and without using delta features in the first HCNF, respectively. These tables show that the hierarchical state posterior feature consistently improves recognition results regardless of the use of delta features. The results with a window length of 23 frames are superior to those with a window length of 9. This is consistent with the results in [10]. The best result $PER = 25.3$ was obtained without using delta features in the first HCNF, with a window length of 23 in the second HCNF, and with the use of HB-MMI and $b = 5.0$ in the second HCNF. The combination of HB-MMI and the hierarchical state posterior feature produced the best results. The reason why the results obtained without using delta features in the first HCNF are superior to those obtained with delta features in the first stage HCNF, could be that models trained without delta features are more robust than those trained with delta features because HCNF can extract useful features without the help of delta features, which act as a kind of filter in the modulation spectrum domain.

Table 4: *Phoneme recognition results with hierarchical state posterior feature on TIMIT core test with delta features [%].*

Window (stage)	b	PER	Train-PER
9 (first)	0.0	28.0	16.1
9 (second)	0.0	26.6	13.4
9 (second)	3.0	26.9	12.6
23 (second)	0.0	26.2	10.9
23 (second)	3.0	26.9	9.4

6. Conclusion

In this paper, we proposed a new training criterion, HB-MMI, which considers training errors for HCNF. Experimental results show that HB-MMI is effective when overfitting is not an issue. This means that HB-MMI would be more effective when applied to a larger corpus. This observation has motivated us to apply HCNF with HB-MMI to LVCSR, which is our first future work. We also proposed using a hierarchical state posterior feature to further improve HCNF based ASR. While the hierarchical state posterior feature clearly improved the recognition results, it yielded the best results when combined with HB-MMI. We want to explore whether the feature is also effective in LVCSR. This is our second future work. To apply HCNF to LVCSR, we need to speed up the HCNF training. Therefore, we also intend exploring parallel training methods and the use of GPUs for HCNF training.

Table 5: *Phoneme recognition results with hierarchical state posterior feature on TIMIT core test without delta features [%].*

Window (stage)	b	PER	Train-PER
9 (first)	3.0	27.9	24.8
9 (second)	0.0	26.4	22.5
9 (second)	3.0	25.7	21.8
23 (second)	0.0	26.1	18.6
23 (second)	3.0	25.5	17.0
23 (second)	5.0	25.3	16.6
23 (second)	10.0	25.9	16.6

7. Acknowledgments

This work was supported by the Global COE Program "Frontiers of Intelligent Sensing" from the Ministry of Education, Culture, Sports, Science and Technology, Japan.

8. References

[1] S. Furui, "Speaker-independent isolated word recognition using dynamic features of speech spectrum," *IEEE Transactions of Acoustics Speech and Signal Processing*, vol. 34, no. 1, pp. 52 – 59, Feb. 1986.

[2] S. Nakagawa and K. Yamamoto, "Speech recognition using hidden markov models based on segmental statistics," *Systems and Computers in Japan*, vol. 28, no. 7, pp. 31–38, Jun. 1997.

[3] N. Kanedera, T. Arai, H. Hermansky, and M. Pavel, "On the relative importance of various components of the modulation spectrum for automatic speech recognition," *Speech Communication*, vol. 28, pp. 43–55, 5 1999.

[4] D. Povey, "Discriminative training for large vocabulary speech recognition," Ph.D. dissertation, Cambridge University Engineering Dept, 2003.

[5] Y. Fujii, K. Yamamoto, and S. Nakagawa, "Automatic speech recognition using hidden conditional neural fields," in *Proc. ICASSP*, Mar. 2011, pp. 5036 – 5039.

[6] D. Povey, D. Kanevsky, and B. Kingsbury, "Boosted MMI for model and feature-space discriminative training," in *Proc. ICASSP*, 2008, pp. 4058 – 4061.

[7] G. Heigold, T. Deselaers, R. Schlüter, and H. Ney, "Modified MMI/MPE: A direct evaluation of the margin in speech recognition," in *Proc. ICML*, 2008.

[8] H. Hermansky, D. Ellis, and S. Sharma, "Tandem connectionist feature stream extraction for conventional HMM systems," in *Proc. ICASSP*, 2000.

[9] E. Fosler.-L. and J. Morris, "Crandem systems: Conditional random field acoustic models for hidden markov models," in *Proc. ICASSP*, 2008, pp. 4049–4052.

[10] J. Pinto, S. Garimella, M. M.-Doss, H. Hermansky, and H. Bourland, "Analysis of MLP-based hierarchical phoneme posterior probability estimatior," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 2, pp. 225–241, 2011.

[11] M. Mahajan, A. Gunawardana, and A. Acero, "Training algorithms for hidden conditional random fields," in *Proc. ICASSP*, May 2006, pp. I–273–I–276.

[12] J. Peng, L. Bo, and J. Xu, "Conditional neural fields," in *Proc. Advances in Neural Information Processing Systems 22*, 2009, pp. 1419–1427.

[13] T. N. Sainath, B. Ramabhadran, and M. Picheny, "An exploration of large vocabulary tools for small vocabulary phonetic recognition," in *Proc. ASRU*, 2009, pp. 359 – 364.

[14] K.-F. Lee and H.-W. HON, "Speaker-Independent Phone Recognition Using Hidden Markov Models," *IEEE Transactions of Acoustics Speech and Signal Processing*, vol. 37, no. 11, pp. 1641 – 1648, Nov. 1989.