INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

LF-MMI Training of Bayesian and Gaussian Process Time Delay Neural Networks for Speech Recognition

Shoukang Hu, Xurong Xie, Shansong Liu, Max W. Y. Lam, Jianwei Yu, Xixin Wu, Xunying Liu, Helen Meng

The Chinese University of Hong Kong, Hong Kong SAR, China

{skhu, ssliu, wylam, jwyu, wuxx, xyliu, hmmeng}@se.cuhk.edu.hk {xrxie}@ee.cuhk.edu.hk

Abstract

Discriminative training techniques define state-of-the-art performance for deep neural networks (DNNs) based speech recognition systems across a wide range of tasks. Conventional discriminative training methods produce deterministic DNN parameter estimates. They are inherently prone to overfitting, leading to poor generalization when given limited training data. In order to address this issue, this paper investigates the use of Bayesian learning and Gaussian Process (GP) based hidden activations to replace the deterministic parameter estimates of standard lattice-free maximum mutual information (LF-MMI) criterion trained time delay neural network (TDNN) acoustic models. Experiments conducted on the Switchboard conversational telephone speech recognition tasks suggest the proposed technique consistently outperforms the baseline LF-MMI trained TDNN systems using fixed parameter hidden activations.

Index Terms: LF-MMI, Bayesian learning, Gaussian Process activation, Speech Recognition

1. Introduction

There has been a long history of using discriminative techniques for improving Automatic Speech Recognition (ASR) performance. In current deep neural network based systems, these techniques define state-of-the-art performance. In previous generation of Hidden Markov Models (HMMs) based acoustic approaches [1, 2, 3, 4, 5, 6] and the current systems using deep neural networks (DNNs) [7, 8, 9, 10], the lattice based sequence level criterion trained systems could improve upon the systems trained with cross-entropy criterion. Recently, motivated by the strong ability of end-to-end approaches which can learn non-deterministic hidden alignments between speech signals and transcriptions represented by Connectionist Temporal Classification (CTC) [11], a lattice free implementation of maximum mutual information (MMI) sequence trained algorithm was proposed in [10] and demonstrated the state-of-the-art speech recognition performance for a wide range of tasks [10, 12].

Discriminative training is in general easily prone to overfit especially when given data is very limited. This phenomenon can be confirmed from previous studies of DNN systems [9]. The overfitting problem can be further aggravated by using conventional stochastic gradient based optimization algorithms, when suboptimal learning rate schedule and smaller batch size were used in sequential update.

To address these issues, two categories of techniques have been used. The first category of methods attempt to address the overfitting issues by improving the underlying optimization techniques. Different scheduling algorithms like Adam [13] are used in combination with stochastic gradient descent to overcome overfitting issues. Second-order optimization methods, e.g., Hessian Free [14, 15, 16] and natural gradient [17, 18, 19], are also investigated. A related approach is that averaging the model parameters obtained from different batch intervals can reduce the overfitting at the end of a training epoch [20]. The second category of techniques are based on criterion smoothing. Inspired by the precursor techniques, e.g., I-smoothing [21], used in the discriminative training of HMM acoustic models, multitask training is used to make a combination between conventional cross-entropy and sequence level MMI criterion in the LF-MMI training [10].

This paper presents an alternative approach to improve the generalization of LF-MMI trained models by using Bayesian estimation. The hidden activations in the standard TDNN models are replaced by either Bayesian or Gaussian Process [22] activations. Then automatically the parameter uncertainty is taking into account in the LF-MMI training process.

The main contribution of this paper is summarized below. First, on top of the strong LF-MMI baseline system which includes cross-entropy regularization, we investigate the use of Bayesian techniques to further improve the generalization power of LF-MMI training for TDNN acoustic models. Experiments conducted on the Switchboard conversational telephone speech recognition tasks suggest the proposed technique consistently outperforms the baseline LF-MMI trained TDNN systems using fixed parameter hidden activation functions. Second, the use of LF-MMI trained GP activation functions, designed for allowing additional flexibility in learning the activation function form, provides further performance improvement. In our previous work [23], we only consider the loss function based on standard cross-entropy criterion. The use of sequence discriminative training techniques has not been investigated.

The remaining part of this paper is organized as follows. Section 2 introduces the sequence discriminative training. Time dealy neural networks (TDNNs) are presented in Section 3. Bayesian and Gaussian Process activations are introduced in Section 4. Sequence discriminative training of Bayesian TDNN (B-TDNN) and Gaussian Process TDNN (GP-TDNN) are presented in Section 5. Section 6 shows the experiments and results. Finally, the conclusions are drawn in Section 7.

2. Sequence Discriminative Training

Neural networks for Automatic Speech Recognition (ASR) are typically trained to classify individual frames based on cross-entropy criterion. However, speech recognition is inherently a sequence classification task. In place of the sequence discriminative training of GMM-HMM systems, neural network acoustic models achieve state-of-the-art performance [7, 8, 10] when trained using the same sequence discriminative crite-

http://dx.doi.org/10.21437/Interspeech.2019-2379

rion in GMM-HMM systems like maximum mutual information (MMI) [1], boosted MMI (BMMI) [2], minimum phone error(MPE) [3] or minimum Bayes risk (MBR) [4, 5, 6]. Typically, the neural network parameters are copied from a cross-entropy based system and then estimated by using word lattices generated by a weak language model. In the lattice based sequence discriminative training, generated lattices are viewed as an approximation for all possible word sequences in the discriminative objective function. The generation of the word lattices is computationally very expensive. To improve the efficiency, a commonly fixed state of model alignments is used. Recently, motivated by the strong ability of end-to-end approaches to learn non-deterministic hidden alignments between speech signals and transcriptions represented by Connectionist Temporal Classification (CTC) [11], a lattice-free maximum mutual information (MMI) (LF-MMI) training of neural network acoustic models [10] was proposed. Compared with the standard lattice based sequence discriminative training, there are three advantages. First, no cross-entropy trained neural network acoustic models are required. Second, no generation of the fixed word lattices is needed. Third, the comparable performance in recognition can be provided by the LF-MMI trained neural network acoustic models. In the following two subsections, we introduce the MMI criterion and LF-MMI criterion.

2.1. MMI criterion

The MMI criterion [1] is chosen to maximize the mutual information between the distributions of acoustic observation sequence and the corresponding word sequences. With $\mathbf{O}_u$ as the sequence of observation for utterance u, and $\mathbf{H}_u$ as the word-sequence in the transcript for utterance u, the MMI criterion is written as follows:

$$F_{MMI}(\{\mathbf{H}_u, \mathbf{O}_u\}; \mathbf{\Theta}) = \sum_u \log \frac{p(\mathbf{O}_u|S_u)^k P(\mathbf{H}_u)}{\sum_{\mathbf{H}'_u} p(\mathbf{O}_u|S'_u)^k P(\mathbf{H}'_u)} \tag{1}$$

where S_u is the sequence of states corresponding to $\mathbf{H}_u$, S'_u is the sequence of states corresponding to the possible word sequence $\mathbf{H}'_u$, $\mathbf{\Theta}$ is the model parameters and k is the acoustic scaling factor. The sum of the denominator is taken over all possible word sequences for utterance u. In the lattice based sequence discriminative training, we use lattices generated by a weak language model as an approximation for all possible word sequences of utterence u.

2.2. LF-MMI

The LF-MMI training [10] calculates the numerator and denominator parts of the objective function directly without using lattices. It interprets the neural network output probability as a log of a pseudo-likelihood. There is no acoustic scaling factor and no division by the prior distribution for the output probability. Compared with the conventional lattice based sequence discriminative training, several modifications are made under this framework.

Topology Change allows the 3-state left-to-right HMM that needs to be traversed by at least 3 frames to be transformed to a different topology that can be traversed in one frame in analogy to CTC [11]. **Phone Language Model** is estimated from phone-level alignments of the training data. **Transition Probability** is set to be a constant value (0.5) that makes each HMM-state sum to one. **Denominator Graph** is constructed from the phone-level denominator language model. The created denominator graph use the modified initial and final probabilities learned from the HMM in the forward-backward computation. **Numerator Graph** adds time constraints to the alignments to split up the numerator graphs appropriately. The weights of the denominator Graph are also added into the numerator Graph. **Regularization Methods** apply three different methods (Cross-entropy, L2 norm, Leaky HMM) to reduce the overfitting problem [9] in the sequence level training. Cross-entropy can be seen as a certain level of smoothing technique to overcome overfitting.

On top of the strong LF-MMI baseline system which includes cross-entropy regularization, we attempt to apply Bayesian estimation techniques to further improve its generalization performance.

3. Time Delay Neural Networks

Time delay neural network (TDNN) proposed in [24] has been shown to be effective in modeling long range temporal dependencies. Experiments in [24] pointed out that TDNN is translation invariant, i.e., the features learned by the time dealy neural network are insensitive to shifts in time. The translation invariant property is useful for representing such tokens that vary considerably from each other due to their different phonetic environment. TDNN may be considered as a precursor to the convolutional neural network [25] since the parameters are tied across different time steps. During back-propagation, the parameters are updated by the accumulative gradients from all the time steps of the input temporal context.

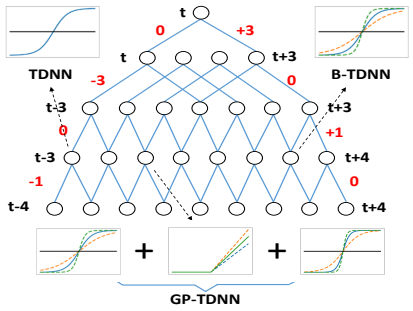

Figure 1: *TDNN/B-TDNN/GP-TDNN architecture example*

The hyper-parameter in the TDNN network is the number of input contexts of each layer required to compute the hidden outputs at one time step. In this paper, we apply the subsampling TDNN [26] structure, which could effectively reduce the computation time. The model structure is shown in Figue 1. Figue 1 shows how to calculate the hidden outputs in each layer and the relationship of different hidden activation outputs. The splicing indexes used in Figue 1 are -1,0 0,1 -3,0 0,3.

4. Bayesian and Gaussian Process Activation Function for TDNNs

In order to explore the generalization power of standard LF-MMI trained TDNN systems with fixed parameters, Bayesian activation considering the parameter uncertainty and Gaussian Process activation which considers both parameter uncertainty and additional activation function uncertainty are presented in this section. The following integral simplified below should be integrated over the total log-likelihood of the neural network.

Table 1: *The forms of parameter uncertainty considered in different B-TDNN and GP-TDNN systems, w.r.t. the number of parameters per layer in different TDNN systems, assuming the input vector size is a and the number of nodes is b.*

System	Uncertainty		#Param
	$\boldsymbol{\lambda}$	$\mathbf{w}$	
B-TDNN	$\times$	$\surd$	ab+a
GP-TDNN0	$\times$	$\times$	ab+3b
GP-TDNN1	$\surd$	$\times$	ab+3b+3
GP-TDNN2	$\times$	$\surd$	ab+3b+a
GP-TDNN3	$\surd$	$\surd$	ab+3b+a+3

4.1. Bayesian Activation Function

Instead of using fixed-point estimates of weight parameters, Bayesian activation shown in Figue 1 uses posterior distributions to model the weight parameter uncertainty [27]. The expected hidden node output $h_i^{(l)}(\mathbf{z}^{(l-1)})$ of the i-th node in the l-th layer is marginalized over different parameter estimates.

$$h_i^{(l)}(\mathbf{z}^{(l-1)}) = \int \phi\left(\mathbf{w}_i^{(l)} \bullet \mathbf{z}^{(l-1)}\right) p(\mathbf{w}_i^{(l)}) d\mathbf{w}_i^{(l)} \quad (2)$$

where $\mathbf{z}^{(l-1)}$ is the input vector fed into the l-th hidden layer, $p(\mathbf{w}_i^{(l)}) = p(\mathbf{w}_i^{(l)} \mid \{\mathbf{H}_u, \mathbf{O}_u\})$ denotes the node dependent activation parameter posterior distribution to be learned from training data, $\phi(\cdot)$ is the activation function and $\bullet$ denotes the dot product.

4.2. Gaussian Process Activation Function

Not only the weight parameters inside activation functions can be regarded as uncertain variables in Bayesian activation, we can also regard the basis coefficients as additional uncertain variables to be integrated over. Thus the GP activation shown in Figue 1 can be modified into the double integration of both weight and coefficient variables in Eqn.(3).

$$\begin{aligned} h_i^{(l)}(\mathbf{z}^{(l-1)}) = \sum_m \int\int \lambda_i^{(l,m)} \phi_m\left(\mathbf{w}_i^{(l,m)} \bullet \mathbf{z}^{(l-1)}\right) \\ p(\mathbf{w}_i^{(l,m)}) p(\lambda_i^{(l,m)}) d\mathbf{w}_i^{(l,m)} d\lambda_i^{(l,m)} \end{aligned} \quad (3)$$

where $p(\lambda_i^{(l,m)}) = p(\lambda_i^{(l,m)} \mid \{\mathbf{H}_u, \mathbf{O}_u\})$ and $p(\mathbf{w}_i^{(l,m)}) = p(\mathbf{w}_i^{(l,m)} \mid \{\mathbf{H}_u, \mathbf{O}_u\})$ denote the basis activation coefficient and parameter posterior distributions respectively, and we assume the statistical independence between these two variables. The general form of Gaussian Process activation in Eqn.(3) proposed in [23] subsuming the previous work in [28] can be simplified to three different special cases in Table 1 depending on the parameter uncertainty being considered.

5. Sequence Discriminative Training of B-TDNN and GP-TDNN

For any error cost function using the cross-entropy or sequence training criterion MMI in Eqn.(1), the same back-propagation algorithm in the gradient chain given in the following Eqn.(4) can be used. The only term needs to be modified as the sequence training MMI criterion is the first part of the chain. This is independent of the choice of cost function at the output layer.

$$\nabla_{\theta_i^l} F = \sum_j \frac{\partial F}{\partial h_j^L} \underbrace{\left\{ \sum_k \frac{\partial h_j^L}{\partial h_k^{L-1}} \cdots \left[\sum_i \frac{\partial h_i^l}{\partial z_i^l} \frac{\partial z_i^l}{\partial \theta_i^l} \right] \right\}}_{\textbf{gradient chain}} \quad (4)$$

where $\theta_i^{(l)}$ corresponds the i-th node dependent parameter in the l-th layer, $h_i^{(l)}$ is the i-th hidden node output in the l-th layer and F is the error cost function, e.g., cross-entropy or MMI.

The commonly used approach to update the hyper-parameters in the posterior distributions requires the calculation of both Bayesian activation and GP activation. For the particular random variables in the Bayesian activation or the GP activation being considered, the general form of MMI loss function previously defined in Eqn.(1) needs to be modified as an integration over all the parameter choices. Thus, the integration in Eqn.(5) is directly non-trivial to be computed. In addition, if the determinant parameters on the last term are replaced by hyper-parameters of the posterior distribution in the Bayesian activation in Eqn.(2) or GP activation in Eqn.(3), the last term is no longer directly differentiable.

$$F = \int F_{MMI}(\{\mathbf{H}_u, \mathbf{O}_u\}; \boldsymbol{\Theta}) P_r(\mathbf{w}) d\mathbf{w} \quad (5)$$

where $\mathbf{w} \in \boldsymbol{\Theta}$, $P_r(\mathbf{w}) = \mathcal{N}(\boldsymbol{\mu}_r, \boldsymbol{\sigma}_r^2)$ denotes the weight prior distribution. We assume that the variational distribution and the prior distribution are both Gaussian distributions following the work in [29].

In order to address these two issues, a commonly used variational lower bound approach is used to approximate the integration in Eqn.(6). The variational lower bound based approach derived for original cross-entropy trained Bayesian and GP neural network needs to be modified as below. For the first term, the integration is further approximated using sampling approaches as in Eqn.(7). This allows the gradient statistics with respect to the hyper-parameters $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ of posterior distribution to be calculated as in Eqn.(9).

$$\begin{aligned} F &\geq \int q(\mathbf{w}) F_{MMI}(\{\mathbf{H}_u, \mathbf{O}_u\}; \boldsymbol{\Theta}) d\mathbf{w} \\ &\quad - KL(q(\mathbf{w}) \| P_r(\mathbf{w})) \\ &= \mathcal{L}_1^{MMI} - \mathcal{L}_2^{MMI} = \mathcal{L}^{\mathcal{MMI}} \end{aligned} \quad (6)$$

where $q(\mathbf{w}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$ is the variational approximation of the parameter posterior distribution $p(\mathbf{w})$, $KL(q\|P_r)$ is the Kullback-Leibler (KL) divergence between q and P_r.

The first term $\mathcal{L}_1^{MMI}$ in Eqn.(6) can be efficiently approximated by Monte Carlo (MC) sampling method.

$$\mathcal{L}_1^{MMI} \approx \frac{1}{N} \sum_{k=1}^{N} F_{MMI}(\{\mathbf{H}_u, \mathbf{O}_u\}; \boldsymbol{\Theta}, \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}_k) \quad (7)$$

where $\boldsymbol{\epsilon}_k = \mathcal{N}(\mathbf{0}, \mathbf{I})$ is the k-th sample. For training, we find that sampling once is already sufficient to train the models; for testing, we directly use the mean statistics to get the output.

The KL divergence between q and P_r of the second term $\mathcal{L}_2^{MMI}$ in Eqn.(6) can be simplified as follows.

$$\mathcal{L}_2^{MMI} = \sum_j \left\{ \log \frac{\sigma_{r,j}}{\sigma_j} + \frac{\sigma_j^2 + (\mu_j - \mu_{r,j})^2}{2\sigma_{r,j}^2} - \frac{1}{2} \right\} \quad (8)$$

where μ_j and σ_j are the j-th component of variational posterior distribution hyper-parameters $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, $\mu_{r,j}$ and $\sigma_{r,j}$ are the j-th component of prior distribution hyper-parameters $\boldsymbol{\mu}_r$ and $\boldsymbol{\sigma}_r$.

The gradient statistics can be computed for the hyper-parameters $\theta = \{\mu_j, \sigma_j\}$ as below.

$$\frac{\partial \mathcal{L}^{\mathcal{MMI}}}{\partial \mu_j} = \frac{1}{N} \sum_{k=1}^{N} \frac{\partial F_{MMI}(\{\mathbf{H}_u, \mathbf{O}_u\}; \boldsymbol{\Theta}, \boldsymbol{\epsilon}_k)}{\partial \mu_j} - \frac{\mu_j - \mu_{r,j}}{\sigma_j^2} \quad (9)$$

$$\frac{\partial \mathcal{L}^{\mathcal{MMI}}}{\partial \sigma_j} = \frac{1}{N}\sum_{k=1}^{N}\frac{\partial F_{MMI}(\{\mathbf{H}_u, \mathbf{O}_u\}; \boldsymbol{\Theta}, \boldsymbol{\epsilon}_k)}{\partial \sigma_j} - \frac{\sigma_j^2 - \sigma_{r,j}^2}{\sigma_j \sigma_{r,j}^2} \quad (10)$$

where $\frac{\partial F_{MMI}(\{\mathbf{H}_u, \mathbf{O}_u\}; \boldsymbol{\Theta}, \boldsymbol{\epsilon}_k)}{\partial \mu_j}$, $\frac{\partial F_{MMI}(\{\mathbf{H}_u, \mathbf{O}_u\}; \boldsymbol{\Theta}, \boldsymbol{\epsilon}_k)}{\partial \sigma_j}$ can be directly calculated using the standard back-propagation method.

An important issue when training B-TDNN and GP-TDNN systems is the parameter prior to use. In this paper, we set the priors for B-TDNN and GP-TDNN systems to be based on the TDNN systems using the same activation with fixed parameters.

6. Experiments

This section describes our experiments carried out on the Switchboard telephone speech recognition tasks using Kaldi speech recognition toolkit [20]. The performance of three LF-MMI based TDNN, B-TDNN, GP-TDNN systems is evaluated.

6.1. Experimental Setup

By using the 300-hour full set and a four-gram language model (LM) trained on the Switchboard and Fisher transcripts, a LF-MMI trained TDNN baseline system gave a WER of **10.0** on the **Hub5' 00 swbd** set [1]. A randomly selected 75-hour Switchboard dataset was used to do the the first set of experiments. The full 300-hour Switchboard dataset was used to conduct the other set of experiments.

In our baseline GMM-HMM system, Maximum Likelihood Linear Transform (MLLT) estimation [30, 31] was used to train the GMM-HMM system on top of Linear Discriminant Analysis (LDA) transformed Perceptual Linear Prediction (PLP) coefficients. The input 39-dimension PLP features include differential parameters up to the second order. The speaker adaptive training (SAT) [32, 33, 34] approach was also applied to further generate the alignments for neural network training and the numerator lattices for LF-MMI training. For neural network training, we follow the kaldi chain model setup[2], except that we used 40-dimension filterbank features as the input features. iVector [35] and speed perturbation were not incorporated. Due to computational resources restriction, all of our models were trained with one thread for 3 epochs on 300-hour Switchboard task and 4 epochs on 75-hour Switchboard task. For performance evaluation, we used HUB5 2000, RT03S and RT02 evaluation sets.

In order to obtain a fair comparison, the B-TDNN and GP-TDNN systems shared the same model structure as the TDNN systems except that the first hidden layer was modified to use the Bayesian activation of Eqn.(2) or the GP activation of Eqn.(3). To retain a comparable number of free parameters, the hyper-parameter $\boldsymbol{\mu}$ was shared within a same hidden node for the GP-TDNN systems and $\boldsymbol{\sigma}$ was shared among all the hidden nodes in the same layer for both the B-TDNN and GP-TDNN systems.

6.2. Experiments on 75-Hour Switchboard Task

In this section, we compare the performance of TDNN, B-TDNN and GP-TDNN systems on the 75-hour Switchboard training set, shown in Table 2. There are two main trends observed in the results of Table 2. First, on top of the strong LF-MMI baseline system which includes cross-entropy regularization, the B-TDNN systems consistently outperform the TDNN baseline systems by **0.7%** absolute WER reduction on the **CHM** subset of the **Hub5' 00** set, **0.6%** absolute WER reduction on the **SWB2** subset of the **Rt03S** set and **1.3%** absolute WER reduction on the **SWB5** subset of the **Rt02** set. Second, further improvement by **0.9%** absolute WER reduction on the **CHM** subset of the **Hub5' 00** set was obtained in the GP-TDNN systems over the B-TDNN systems. Compared with the TDNN baseline systems, **1.6%**, **1.3%** and **1.0%** absolute WER reductions were obtained on the **CHM** subset of the **Hub5' 00** set, **SWB2** subset of the **Rt03S** set and **SWB5** subset of the **Rt02** set respectively by GP-TDNN1 system and GP-TDNN2 system.

Table 2: *Performance (WER%) comparison of baseline TDNN, B-TDNN and GP-TDNN systems on the **HUB5' 00**, **RT03S** and **RT02** evaluation sets*

System	Hub5' 00		Rt03S		Rt02		
	SWB1	CHM	FSH	SWB2	SWB3	SWB4	SWB5
TDNN	12.2	24.9	16.6	26.3	15.0	19.6	27.3
B-TDNN	**11.8**	**24.2**	16.5	**25.7**	14.4	19.4	**26.0**
GP-TDNN0	12.1	24.5	16.6	26.2	**14.3**	19.6	27.3
GP-TDNN1	**11.6**	**23.3**	16.3	25.5	14.6	19.3	27.3
GP-TDNN2	**11.6**	24.2	**16.0**	**25.0**	**14.3**	**19.0**	**26.3**
GP-TDNN3	11.8	24.0	16.2	25.3	14.4	19.3	26.7

6.3. Experiments on 300-Hour Switchboard Task

The performance of different TDNN, B-TDNN and GP-TDNN systems on the full 300-hour Switchboard training set is shown in Table 3. Several trends can be observed. First, in common with the trend we found in Table 2, absolute WER reductions of **0.4%**, **0.8%**, **0.4%**, **0.8%**, **0.7%**, **0.6%** for the **SWB1** subset of **Hub5' 00** set, **FSH** and **SWB2** subsets of the **Rt03S** set, and **SWB3**, **SWB4** and **SWB5** subsets of the **Rt02** set were obtained respectively using B-TDNN systems. Second, in contrast with the trend in Table 2, the GP-TDNN systems do not outperform the B-TDNN systems which are currently under further investigation.

Table 3: *Performance (WER%) comparison of TDNN, B-TDNN and GP-TDNN systems on the **HUB5' 00**, **RT03S** and **RT02** evaluation sets*

System	Hub5' 00		Rt03S		Rt02		
	SWB1	CHM	FSH	SWB2	SWB3	SWB4	SWB5
TDNN	10.0	20.8	14.0	22.0	12.4	16.7	22.8
B-TDNN	**9.6**	20.7	**13.2**	**21.6**	**11.6**	**16.0**	**22.2**
GP-TDNN0	9.8	20.8	13.9	22.2	12.0	16.5	22.8
GP-TDNN1	9.7	20.7	13.6	22.3	12.0	16.7	23.4
GP-TDNN2	**9.6**	**20.5**	13.5	22.0	**11.6**	16.5	22.8
GP-TDNN3	9.8	**20.5**	13.6	22.0	12.1	16.5	23.0

7. Conclusions

In this paper, we investigate the use of Bayesian and Gaussian Process (GP) based hidden activation functions to improve the generalization power of LF-MMI trained neural network acoustic models. Consistent performance improvements using the B-TDNN and GP-TDNN systems were obtained over the baseline TDNN systems with deterministic activation function and fixed parameter estimates in the perspective of WER reduction. Future research will focus on the scalability of LF-MMI trained B-TDNN and GP-TDNN systems.

8. Acknowledgements

This research is supported by Hong Kong Research Grants Council General Research Fund No. 14200218 and Shun Hing Institute of Advanced Engineering Project No. MMT-p1-19.

[1] The comparable LF-MMI system reported in [10] gave a WER of 10.2% in Hub5' 00 swbd set.

[2] All of this is in published Kaldi code at github.com/kaldi-asr/kaldi/egs/swbd/s5c/local/chain/run_tdnn.sh

9. References

[1] L. R. Bahl, P. F. Brown, P. V. De Souza, and R. L. Mercer, "Maximum mutual information estimation of hidden markov model parameters for speech recognition," in *proc. icassp*, vol. 86, 1986, pp. 49–52.

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

[3] D. Povey, "Discriminative training for large vocabulary speech recognition," Ph.D. dissertation, University of Cambridge, 2005.

[4] J. Kaiser, B. Horvat, and Z. Kacic, "A novel loss function for the overall risk criterion based discriminative training of hmm models," in *Sixth International Conference on Spoken Language Processing*, 2000.

[5] M. Gibson and T. Hain, "Hypothesis spaces for minimum bayes risk training in large vocabulary speech recognition," in *Ninth international conference on spoken language processing*, 2006.

[6] D. Povey and B. Kingsbury, "Evaluation of proposed modifications to mpe for large scale discriminative training," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, vol. 4. IEEE, 2007, pp. IV–321.

[7] B. Kingsbury, "Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 3761–3764.

[8] K. Veselỳ, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks." in *Interspeech*, vol. 2013, 2013, pp. 2345–2349.

[9] H. Su, G. Li, D. Yu, and F. Seide, "Error back propagation for sequence training of context-dependent deep networks for conversational speech transcription," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 6664–6668.

[10] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi." in *Interspeech*, 2016, pp. 2751–2755.

[11] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 369–376.

[12] V. Peddinti, V. Manohar, Y. Wang, D. Povey, and S. Khudanpur, "Far-field asr without parallel data." in *INTERSPEECH*, 2016, pp. 1996–2000.

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[14] B. Kingsbury, T. N. Sainath, and H. Soltau, "Scalable minimum bayes risk training of deep neural network acoustic models using distributed hessian-free optimization," in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[15] T. N. Sainath, L. Horesh, B. Kingsbury, A. Y. Aravkin, and B. Ramabhadran, "Accelerating hessian-free optimization for deep neural networks by implicit preconditioning and sampling," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*. IEEE, 2013, pp. 303–308.

[16] I.-H. Chung, T. N. Sainath, B. Ramabhadran, M. Picheny, J. Gunnels, V. Austel, U. Chauhari, and B. Kingsbury, "Parallel deep neural network training for big data on blue gene/q," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 6, pp. 1703–1714, 2017.

[17] H. H. Yang and S.-i. Amari, "Complexity issues in natural gradient descent method for training multilayer perceptrons," *Neural Computation*, vol. 10, no. 8, pp. 2137–2157, 1998.

[18] N. L. Roux, P.-A. Manzagol, and Y. Bengio, "Topmoumoute online natural gradient algorithm," in *Advances in neural information processing systems*, 2008, pp. 849–856.

[19] D. Povey, X. Zhang, and S. Khudanpur, "Parallel training of deep neural networks with natural gradient and parameter averaging," *arXiv preprint arXiv:1410.7455*, 2014.

[20] D. Povey, A. Ghoshal, G. Boulianne, and et al., "The kaldi speech recognition toolkit," Tech. Rep., 2011.

[21] D. Povey and P. C. Woodland, "Minimum phone error and i-smoothing for improved discriminative training," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, 2002, pp. I–105.

[22] M. Seeger, "Gaussian processes for machine learning," *International journal of neural systems*, vol. 14, no. 02, pp. 69–106, 2004.

[23] S. Hu, M. W. Lam, X. Xie, S. Liu, J. Yu, X. Wu, X. Liu, and H. Meng, "Bayesian and gaussian process neural networks for large vocabulary continuous speech recognition," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6555–6559.

[24] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *Backpropagation: Theory, Architectures and Applications*, pp. 35–61, 1995.

[25] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[26] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[27] R. M. Neal, *Bayesian learning for neural networks*. Springer Science & Business Media, 2012, vol. 118.

[28] M. W. Lam, S. Hu, X. Xie, S. Liu, J. Yu, R. Su, X. Liu, and H. Meng, "Gaussian process neural networks for speech recognition," *Proc. Interspeech 2018*, pp. 1778–1782, 2018.

[29] D. Barber and C. M. Bishop, "Ensemble learning in bayesian neural networks," *NATO ASI SERIES F COMPUTER AND SYSTEMS SCIENCES*, vol. 168, pp. 215–238, 1998.

[30] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer speech & language*, vol. 9, no. 2, pp. 171–185, 1995.

[31] M. J. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer speech & language*, vol. 12, no. 2, pp. 75–98, 1998.

[32] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*, vol. 2. IEEE, 1996, pp. 1137–1140.

[33] T. Anastasakos, J. McDonough, and J. Makhoul, "Speaker adaptive training: A maximum likelihood approach to speaker normalization," in *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2. IEEE, 1997, pp. 1043–1046.

[34] D. Povey, H.-K. J. Kuo, and H. Soltau, "Fast speaker adaptive training for speech recognition," in *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[35] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.