Incorporating Local Environment Information with Ensemble Neural Networks to Robust Automatic Speech Recognition

Chia-Yung Hsu[1], Ryandhimas E. Zezario[1], Jia-Ching Wang[1], Xugang Lu[2], and Yu Tsao[3]

[1] Department of Computer Science and Information Engineering, National Central University
[2]National Institute of Information and Communications Technology, Japan
[3] Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan

Abstract

This paper proposes an ensemble neural network (ENN) framework for robust automatic speech recognition (ASR). The proposed ENN framework can be divided into offline and online phases. In the offline phase, the ENN framework first applies an environment clustering technique to partition the training data into several subsets, where each subset characterizes specific local information of the entire acoustic space. Next, each subset of training data is adopted to train an NN acoustic model. Finally, the entire set of training data is used to estimate a gating function, which can determine the most suitable NN acoustic model given an input utterance. In the online phase, given the testing utterance, the gating function specifies the optimal NN acoustic model to perform speech recognition. Because local environment information is incorporated, ENN can effectively determine the NN acoustic model that optimally matches the testing condition. The proposed framework was evaluated on the Aurora-2 task. Experimental results show that the proposed ENN framework can provide a notable word error rate reduction of 5.35% (from 5.05% to 4.78%) when compared to the baseline

Index Terms: Ensemble neural network, robust ASR, environment clustering, mixture of local experts.

1. Introduction

The performance of automatic speech recognition (ASR) has been significantly improved since artificial neural network (ANN) has been used as the fundamental model to characterize speech patterns [1-4]. Despite the high recognition accuracy in training-testing matched conditions, the performance notably degrades under training-testing mismatched conditions. Therefore, robust ASR remains a crucial research topic. Numerous approaches have been proposed to improve the robustness of ANN-based ASR systems. A class of approaches aims to improve acoustic features. Successful examples include vector Taylor series [5, 6] and spectral masking approaches [7, 8]. Meanwhile in [9], ensemble neural nets are prepared to accommodate noise of different levels, and a second-stage neural net is used to obtain optimal information extraction. Another class of approaches handles the mismatch by estimating acoustic models to match the testing conditions. Well-known approaches include linear input network (LIN), linear hidden network (LHN), and linear output network (LON), which adopt transformations to the input, hidden, and output layers, respectively, of ANN acoustic models [10-12]. Some related approaches directly perform model adaptation upon parameters in ANNs [13-15]. Speaker adaptive training (SAT), on the other hand, aims to prepare acoustic models that are more independent of specific training speakers and thus can generalize better to unseen testing speakers [16, 17]. Moreover, some approaches combine different types of acoustic models [18] or features [19] into a unified system; due to the nice flexibility and modeling capability, ANN can incorporate complementary information to achieve better ASR performance.

In the machine learning community, using an ensemble of ANNs to perform pattern classification has been extensively investigated [20, 21]. The main idea of "ensemble of ANNs" is to classify an input pattern by combining the classification results from an ensemble of independently trained ANNs. Among the combination methods, bagging and boosting are two popular ones. Bagging generates multiple predictors to obtain several training sets from the original training set to train multiple predictors [22], and boosting combines multiple weak predictors to form a strong predictor [23]. Another successful combination method is based on the mixture of local experts (MLE) theory [24]. The main concept of the MLE theory is to handle a complicated task in a divide-and-conquer style: a set of local ANNs is prepared, and a gating function is designed and used to combine the outputs of local ANNs given the input pattern. The local ANNs and the gating function are trained simultaneously based on a unified cost function.

Although using a large and deep neural network can automatically learn each type of noise and environment condition in a distributed way, we argue that the learned model is easily overfit since large quantity of free parameters need to be trained, particularly when with limited training data. In this study, rather than using ``brute force'' to build a large network which needs large quantity of training data samples, we propose an ensemble NNs (termed ENN) framework for robust ASR. The ENN framework includes two phases, offline and online. In the offline phase, we apply the environment clustering (EC) algorithm to partition the entire training set into several subsets, which are then used to prepare an ensemble of ANNs. Next, the entire set of training data is used to compute a gating function, which can specify the most suitable NN acoustic model given an input utterance. In the online phase, given a testing utterance, the gating function determines the optimal ANN to perform ASR. The main advantage of the proposed ENN framework is that since each ANN explicitly represents a particular acoustic condition, the gating function only requires a limited amount of data to determine an acoustic model that matches the testing condition. Moreover, particular member ANNs can be replaced by more suitable ones according to the testing task. We evaluate the proposed ENN framework on the Aurora-2 task [26]. Experimental results show that the proposed framework can provide a notable word error rate (WER) reduction of 5.35 % (from 5.05 % to 4.78 %) compared to the baseline system.

2. Related Works

This section first reviews the EC algorithm, which is used in the ENN framework. Next, the MLE theory is presented

2.1. Environment Clustering (EC)

In real-world speech processing applications, speech waveforms are usually distorted by multiple distortion sources, including speaker variability (gender, age, and accents, etc.) or adverse speaking environments (noise, channel, and reverberation, etc.). The main idea of the EC algorithm is to cluster the entire set of training data into several subsets, each subset representing similar acoustic characteristics and carrying useful local information. In this study, a tree is adopted to perform EC (as shown in Fig. 1). Assume that the tree has *C* nodes, including the root node, intermediate nodes, and leaf nodes, we cluster the entire set of training data into *C* subsets $\{D^1, D^2, \dots, D^C\}$, where each subset contains local information about the entire training data. Various approaches has been proposed to effectively use such local information on different tasks, such as speech enhancement [27], feature compensation [25], and model adaptation processes [28].

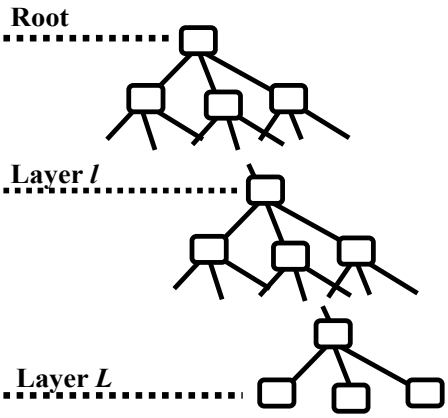

Figure 1: *The tree structure for environment clustering.*.

2.2. Mixture of Local Experts (MLE)

For a complicated pattern recognition task, a direct solution may not attain satisfactory performance, and a divide-and-conquer strategy has been proven a successful solution. The MLE theory is derived based on the divide-and-conquer strategy and constructs an architecture as shown in Fig. 2 [24]. In the MLE theory, several "expert" sub-ANNs and a "gating" ANN is estimated. Each expert tackles part of the problem, and the gating network integrates all of the expert networks. The cost function, J, of the MLE theory is:

$$J = \sum_{i=1}^{N} p_u^i \left\| d_u - y_u^i \right\| \qquad (1)$$

where y_u^i is the output vector of the i-th sub-ANN, p_u^i is the proportional contribution of the i-th sub-ANN, and d_u is the expected output vector for the case *u*. In [24], they also derive an alternative cost function, which can provide better classification performance:

$$J = -\log \sum_{i=1}^{N} p_u^i exp^{\left\| d_u - y_u^i \right\|^2} \qquad (2)$$

With the cost functions defined in Eqs. (1) and (2), the local ANNs and the gating function are trained simultaneously.

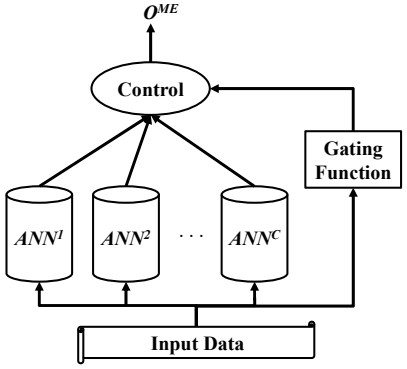

Figure 2: *The system architecture of the mixture of experts (MOE) network.*

3. Proposed ENN Framework

This section presents the offline and online phases in the proposed ENN framework.

3.1. Offline Phase

In the offline phase, we first build a tree based on the EC algorithm to partition the training data. The tree can be prepared either in a knowledge-based or data-driven manner. For the knowledge-based manner, the available speaker and speaking environment information can be involved to cluster the entire set of training data. For the data-driven manner, on the other hand, some clustering techniques can be adopted to perform EC. In our previous study, we have proposed a data-driven approach for EC [25]. The overall procedure is presented in Fig. 3, which includes four steps: (1) The entire training set is used to compute a Gaussian mixture model universal background model (GMM-UBM). (2) An utterance-specific GMM is estimated by performing MAP [29] adaptation on the GMM-UBM; for *U* training utterances, we obtain *U* utterance-specific GMMs. (3) The mean parameters in each GMM are concatenated into a super-vector. (4) The k-means vector quantization (VQ) [30] is applied to the *U* utterance-specific super-vectors, and the VQ results are used as the EC results.

For the tree with *C* nodes, we obtain *C* subsets of training data $\{D^1, D^2, \dots, D^C\}$ and accordingly prepare *C* sets of ANN acoustic models, $\{\Lambda^1, \Lambda^2, \dots, \Lambda^C\}$. To effectively utilize the available training data, we first use the entire set of training data to build a root ANN acoustic model. Then, the ANN acoustic models are trained by the sub-set of training data in the next layer. The same procedure repeats until we estimate

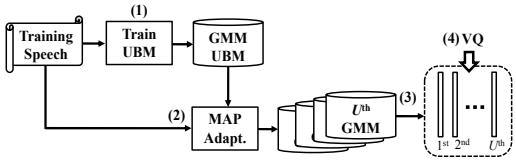

Figure 3: *The data-driven approach for training data partitioning in an unsupervised manner.*

the complete *C* sets of ANN acoustic models. In addition to the *C* sets of ANN acoustic models, we prepare a gating function in the offline phase. In this study, we prepare a GMM-based gating function. This gating function comprises *C* sets of GMMs, $\{\lambda^1, \lambda^2, \ldots, \lambda^C\}$, which are prepared by the *C* subsets of training data $\{D^1, D^2, \ldots, D^C\}$. We denote the GMM-based gating function as GMM-gating in the following discussion. Figure 4 presents the overall diagram of the ENN framework with multiple ANN acoustic models and a GMM-gating.

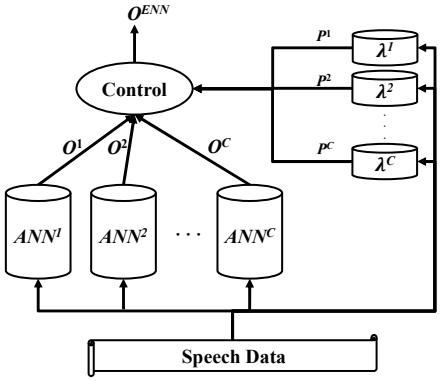

Figure 4: *The architecture of the ENN framework with a GMM-gating function.*

3.2. Online Phase

In the online phase, given a testing utterance, $Y = \{y_1, y_2, \ldots y_T\}$, where y_t denotes the *t*-th frame in *Y*, we can obtain *C* outputs $(O^1, O^2, \ldots, O^C)$ by the *C* sets of ANN acoustic models. Next, the same testing utterance, *Y*, is used to estimate the likelihood score (contribution) for the *p*-th model by

$$p^c = P(Y|\lambda^c) = \sum_{t=1}^{T}\sum_{k=1}^{K} w_k^c \mathcal{N}(y_t|\mu_k^c, \Sigma_k^c) \quad (3)$$

where K is the number of Gaussian components, μ_k^c, Σ_k^c and w_k^c are the mean, covariance matrix, and weight parameters, respectively, of the *k*-th Gaussian component of the c-th GMM. With the *C* likelihood scores $(p^1, p^2, \ldots, p^C)$ we determine the ENN output based on the best first criterion by:

$$c^* = \underset{c=1,2,\ldots,C}{\operatorname{argmax}}\, p^c \quad (4)$$

and thus

$$O^{ENN} = O^{c^*}, \quad (5)$$

where O^{c^*} is the output for the c^*-th ANN.

4. Experiments

4.1. Experimental Setup

We evaluated the proposed ENN framework on the Aurora-2 task. The Aurora-2 database includes two training sets, clean and multi-condition training sets, and the multi-condition training set is used in this study. The training set contains four different types of noise (subway, babble, car, and exhibition) with five levels of signal-to-noise-ratio (SNR): 5 dB, 10 dB, 15 dB, 20 dB, and clean. There are 8440 training utterances, which amount to around four hours of speech data in total. The testing set contains eight types of noise (subway, babble, car, exhibition, airport, street, train station and restaurant) with seven level SNRs (-5 dB, 0 dB, 5 dB, 10 dB, 15 dB, 20 dB, and clean). The testing set A includes the same noise types as those in the training set. The noise types in Set B are unseen in training set, and Set C includes subway and street noises with additional channel distortion. Aurora-2 is a continuous digit recognition task, and thus no language model was used for this task. An equal probability digit-loop was adopted for decoding. The word error rate (WER) in % was used as the evaluation metric. A lower WER indicates a better recognition result.

We used the ETSI advanced front-end feature (AFE) [31] as the acoustic feature. The ETSI AFE feature has shown its superior performance in various noise robust ASR tasks. The feature is designed to remove mismatch by several stages of noise reduction schemes, including a two-stage Wiener filter, SNR-dependent waveform processing, cepstrum calculation, and blind equalization. For the GMM used in the GMM-gating function, we use 13 dimensional AFE static features. For the ANN acoustic models, we first prepared 39 dimensional feature vector, with 13 static AFE features and their first and second order dynamic features. Next, 11 consecutive frames are concatenated to incorporate context information, and thus 429 (39×11) dimensional features are used as the input of ANN acoustic models. We use the Kaldi open source toolkit [32] to construct our ANN acoustic models.

In this study, the first layer of EC tree was decided according to the gender of training data, and thus the first layer has two nodes, one for male and one for female, each including 4220 training utterances. To determine the nodes in the second layer, we follow the super-vector with k-means clustering procedure as presented in Section 3.2. Since the first layer was constructed in a knowledge-based manner, and the second layer was constructed by a data-driven manner, we consider the prepared tree a hybrid knowledge and data-driven based EC tree. In addition to the root node and the male and female nodes in the first layer, the second layer prepares four nodes according to the k-means clustering results. Accordingly there are seven nodes (*C*=7) in total for the EC tree.

For all of the ANN acoustic models, the pre-training and back-propagation processes were the same. For each model, the HMM topology involved 3 states for silence and 16 states for digits. The dropout training strategy [33] was used during the training process, and the dropout rate was set to 0.7. A stack of RBMs was pre-trained to initialize weights of ANN acoustic models. Before training the next layer RBM, a back-propagation supervised fine-tune was performed with 30 iterations. Gaussian-Bernoulli RBM was trained using a low momentum of 0.5 with learning rate of 0.1, for the first 10 iterations, and then a larger momentum of 0.9 and learning of 0.001 was performed for 80 iterations. Bernoulli-Bernoulli RBM using low and high momentum training while 10 and 40 iterations, respectively, were performed for low and high momentum training. An L2-penalty of 0.0002 was used. After the stacked RBMs were trained, the standard back-propagation fine-tuning was carried out with 30 iterations. The learning rate was set to 0.015 initially and will be halved when the number of iterations was more than 10 and the training accuracy was less than that of the previous iteration. The batch size was set to 128 for both RBM training and fine-tuning.

4.2. Experimental Results

For the Aurora-2 evaluation, we are more interested in the results from SNR 0 dB to 20 dB conditions. Thus, we only

presented the average WERs over SNR 0dB to 20dB conditions for the three testing sets (Set A, Set B and Set C). In addition, an average result across the ten noise types over SNR 0dB to 20dB conditions was also presented and denoted as "Avg".

First, we investigate the optimal structure for the ANN acoustic models. The WERs of ANN acoustic models using one, two, three, and four layers are shown in Table 1. For each layer number, we tested ASR performance using different numbers of neuron (from 128 to 3072) and only reported the best results in Table 1. From the table, we found that the single-layer (with 2560 neurons) ANN acoustic model achieves the best performance across the tree test sets and the average results. The result is not exactly the same as other studies that report ANN with deeper structures can achieve better ASR results [7, 34]. A possible reason for such inconsistency could be that the AFE acoustic feature has adopted several stages of complex processes, and thus the mismatch has already been effectively suppressed. Consequently, a single-layer ANN can already achieve satisfactory performance. Please note that, the main focus of this study is to demonstrate the effectiveness of the ENN framework. Thus it is more important to show that by integrating multiple ANN acoustic models we can gain higher ASR accuracy. Therefore, the single-layer ANN with 2560 neurons was used as our ANN acoustic model structure.

Table 1. *WER (in %) of ANN acoustic model with 1 to 4 hidden layers. The best results are shown with bold digits.*

Model Structure	Set A	Set B	Set C	Avg.
ANN (1 Layer)	**4.57**	**5.44**	**5.24**	**5.05**
ANN (2 Layer)	4.74	6.64	5.73	5.70
ANN (3 Layer)	5.30	7.32	6.62	6.37
ANN (4 Layer)	4.81	6.88	5.84	5.84

Next, we intend to optimize the GMM-gating function in the ENN framework. To this end, we carried out a gender classification test for GMM-gating with different Gaussian mixture components. Since GMM-gating is to specify the optimal ANN acoustic model for recognition, the gender classification result serve as a good indicator to evaluate the gating function performance. The Aurora-2 database provided the gender labels for the testing utterances, and we used the GMM-gating with two GMMs, one for male and one for female. The two GMMs use the same number of Gaussian mixture components. The results of gender classification error rate

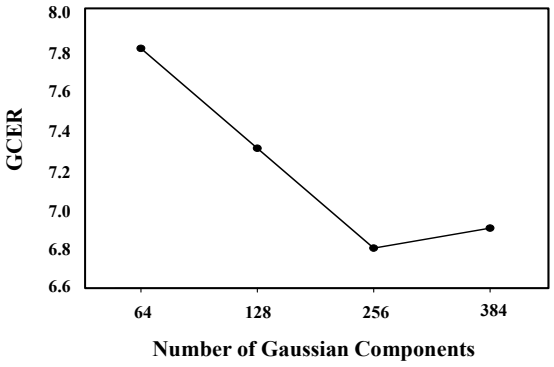

Figure 5: *GCER (in %) using GMM with different numbers of Gaussian components.*

(GCER) with different Gaussian components are shown in Fig. 5. From the figure, we note that GMM with 256 Gaussian mixture components gives the best performance with 6.8% GCER (93.2% accuracy). In the following discussion, the GMM with 256 Gaussian components was used for the GMM-gating for the ENN framework.

Table 2 shows the WERs of ENN with two different gating functions: GMM-gating and gender informed GMM-gating (termed GIG for simplicity). For the GMM-gating, Eqs. (3) to (5) were used to determine the optimal ANN acoustic model given the testing utterances; for the GIG–gating, we assume that accurate gender information is accessible and used into GMM-gating. The baseline results are the ones of ANN (1 Layer) in Table 1. From Table 2, we note that GMM-gating ENN notably outperforms the baseline with a clear 5.35% (from 5.05% to 4.78%) WER reduction. Moreover, GIG-gating ENN provides further improvements, with a notable 9.70% (from 5.05% to 4.56%) WER reduction. The results confirm three points: (1) ENN can effectively improve the baseline (single ANN) system; (2) the gating function plays a crucial role in ENN; (3) with additional information, better gating function can be designed to obtain higher ASR results.

Table 2. *WER (in %) of baseline and GMM-gating and GIG-gating ENN. The best results are shown with bold digits.*

Methods	Set A	Set B	Set C	Avg.
Baseline	4.57	5.44	5.24	5.05
GMM-gating ENN	4.31	5.17	4.92	4.78
GIG-gating ENN	**4.13**	**4.94**	**4.64**	**4.56**

5. Conclusion

In this paper, we proposed an ENN framework for robust ASR under training-testing mismatched conditions. Experiments were conducted on the Aurora-2 task. For a fair comparison, the same set of training data is used to build the ENN framework and the baseline (with a single ANN acoustic model), and the same structures are adopted for all of the ANN acoustic models in this study. Results showed that ENN can achieve better recognition performance than the baseline. Two additional advantages of ENN are noted: (1) the modularized structure of ENN allows flexibility of including relevant or removing irrelevant ANN models for specific tasks; (2) the gating function can be improved when useful information is available. This study adopts GMM to construct the gating function and the best first method to determine the outputs. Designing advanced gating functions (linear or nonlinear) will be our first future work. Meanwhile, we will evaluate the proposed framework on large vocabulary continuous speech recognition (LVCSR) tasks.

6. References

[1] G. Hinton, L. Deng, D. Yu and G. E. Dahl, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 82-97, 2012.

[2] A,-R. Mohamed, G. E. Dahl and G. Hinton, "Acoustic modeling using deep belief networks," IEEE Transactions on Audio, Speech, and Language Processing, vol. 20, no. 1, pp. 14-22, 2012.

[3] G. Dahl, D. Yu, L. Deng and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," IEEE Transactions on Audio, Speech, and Language Processing, vol. 20, no. 1, pp. 30-42, 2012.

[4] L. Deng, J. Li, J. T. Huang, K. Yao, D. Yu, F. Seide, M. Seltzer, G. Zweig, X. He, J. Williams, Y. Gong and A. Acero, "Recent advances in deep learning for speech research at Microsoft," In Proc. ICASSP'13, pp. 8604-8608, 2013.
[5] B. Li and K. C. Sim, "Noise adaptive front-end normalization based on vector Taylor series for deep neural networks in robust speech recognition," In Proc. ICASSP'13, pp. 7408-7412, 2013.
[6] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," IEEE Transactions on Audio, Speech, and Language Processing, vol. 22, no. 4, pp. 745-777, 2014.
[7] B. Li and K. C. Sim, "A spectral masking approach to noise-robust speech recognition using deep neural networks," IEEE Transactions on Audio, Speech and Language Processing, vol. 22, pp. 1296-1305, 2014.
[8] A. Narayanan and D. Wang, "Improving robustness of deep neural network acoustic models via speech separation and joint adaptive training," IEEE Transactions on Audio, Speech and Language Processing, vol. 23, pp. 92-101, 2015.
[9] F. Li, P. S. Nidadavolu, and H. Hermansky, "A long, deep and wide artificial neural net for robust speech recognition in unknown noise," In Proc. INTERSPEECH'14, 2014.
[10] J. Neto, L. Almeida, M. Hochberg, C. Martins, L. Nunes, S. Renals and T. Robinson, "Speaker-adaptation for hybrid HMM-ANN continuous speech recognition system," In Proc. Eurospeech'95, pp. 2171–2174, 1995.
[11] R. Gemello, F. Mana, S. Scanzio, P. Laface and R. D. Mori, "Linear hidden transformations for adaptation of hybrid ANN/HMM models," Speech Communication, vol. 49, no. 10, pp. 827–835, 2007.
[12] K. Yao, D. Yu, F. Seide, H. Su, L. Deng, and Y. Gong, "Adaptation of context-dependent deep neural networks for automatic speech recognition," IEEE Spoken Language Technology Workshop, 2012.
[13] D. Yu, K. Yao, H. Su, G. Li, and F. Seide, "KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition," In Proc. ICASSP'13, pp. 7893–7897, 2013.
[14] H. Liao, "Speaker adaptation of context dependent deep neural networks," In Proc. ICASSP'13, pp. 7947–7951, 2013.
[15] S. M. Siniscalchi, J. Li, and C.-H. Lee, "Hermitian polynomial for speaker adaptation of connectionist speech recognition systems," IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 10, pp. 2152-2161, 2013.
[16] Y. Miao, H. Zhang, and F. Metze, "Towards speaker adaptive training of deep neural network acoustic models," In Proc. INTERSPEECH'14, 2014.
[17] T. Ochiai, S. Matsuda, X. Lu, C. Hori, and S. Katagiri, "Speaker adaptive training using deep neural networks," In Proc. ICASSP'14, pp. 6349-6353, 2014.
[18] S. Liu and K. C. Sim, "On combining DNN and GMM with unsupervised speaker adaptation for robust automatic speech recognition," In Proc. ICASSP'14, pp. 195-199, 2014.
[19] A. Senior, "Improving DNN speaker independence with I-vector inputs," In Proc. ICASSP'14, pp. 225-229, 2014.
[20] L. K. Hansen, P. Salamon, "Neural network ensembles," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 10, pp.993-1001, 1990.
[21] Z. H. Zhou, J. Wu and W. Tang, "Ensembling neural networks: many could be better than all," Artificial Intelligence, vol. 137, no.1-2, pp. 239-263, 2002.
[22] L. Breiman, "Bagging predictors," The Journal of Machine Learning Research, vol. 24, no. 2, pp. 123-140, 1996.
[23] H. Schwenk and Y. Bengio, "Boosting neural network," Neural Computation, vol. 12, no. 8, pp. 1869-1887, 2000.
[24] R. A. Jacobs, M. I. Jordan, S. J. Nowlan and G. E. Hinton, "Adaptive mixtures of local experts," Neural Computation, vol. 3, no. 1, pp. 79-87, 1991.
[25] Y. Tsao, X. Lu, P. Dixon, T.-y. Hu, S. Matsuda, and C. Hori, "Incorporating local information of the acoustic environments to MAP-based feature compensation and acoustic model adaptation," Computer Speech and Language, vol. 28, no. 3, pp. 709-726, 2014.
[26] D. Pearce and H.-G. Hirsch, "The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in ASR2000 Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop, 2000.
[27] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Ensemble modeling of denoising autoencoder for speech spectrum restoration," In Proc. INTERSPEECH'14, 2014.
[28] Y. Tsao and C.-H. Lee, "An ensemble speaker and speaking environment modeling approach to robust speech recognition," IEEE Transactions on Audio, Speech, and Language Processing, vol. 17, pp. 1025-1037, 2009.
[29] J. Gauvian, C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," IEEE Transactions on Speech Audio Process, vol. 2 , no. 2, pp. 291-298, 1994.
[30] R. O. Duda, P. E. Hart, D. G. Stork "Pattern Classification," Wiley, New York, 2001.
[31] ETSI, "Speech processing, transmission and quality aspects (STQ); distributed speech recognition; advanced front-end feature extraction algorithm; compression algorithm," ETSI standard document ES 202 050 V1.1.5, 2007
[32] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," IEEE Workshop on Automatic Speech Recognition and Understanding, 2011.
[33] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," The Journal of Machine Learning Research, pp. 1929–1958, 2014.
[34] B. Li, Y. Tsao and K. C. Sim, "An investigation of spectral restoration algorithms for deep neural networks based noise robust speech recognition," In Proc. INTERSPEECH'13, 2013.