Integrating Acoustic and State-Transition Models for Free Phone Recognition in L2 English Speech Using Multi-Distribution Deep Neural Networks

Kun Li, Xiaojun Qian, Shiying Kang, Pengfei Liu, Helen Meng

Department of System Engineering and Engineering Management
The Chinese University of Hong Kong
{kli, xjqian, sykang, pfliu, hmmeng}@se.cuhk.edu.hk

Abstract

This paper investigates the use of Multi-Distribution Deep Neural Networks (MD-DNNs) for integrating acoustic and state-transition models in free phone recognition of L2 English speech. In Computer-Aided Pronunciation Training (CAPT) system, free phone recognition for L2 English speech is the key model of Mispronunciation Detection and Diagnosis (MDD) in the cases of allowing freely speaking. A simple Automatic Speech Recognition (ASR) system can be approached with an Acoustic Model (AM) and a State-Transition Model (STM). Generally, these two models are trained independently, hence contextual information maybe lost. Inspired by the Acoustic-Phonological Model, which achieves greatly improvements by integrating the AM and Phonological Model (PM) in MDD for the cases that L2 learners practice their English by following the prompts, we propose a joint Acoustic-State-Transition Model (ASTM) which uses a MD-DNN to integrate the AM and STM. Preliminary experiments with basic parameter configurations show that the ASTM obtains a phone accuracy of about 68% on the TIMIT data. It is better than the system of using separate AM and STM, whose accuracy is only about 52%. Further fine-tuning the ASTM achieves an accuracy of about 72% on the TIMIT data. Similar performance is obtained if we train and test the ASTM on our L2 English speech corpus (CU-CHLOE).

Index Terms: speech recognition, L2 English speech, deep neural networks, acoustic models, state transition model

1. Introduction

Computer-aided pronunciation training (CAPT) technologies enable self-directed language learning with round-the-clock accessibility and individualized feedback. They can supplement the teachers' instructions and help meet the demand of a growing population of learners in face of a shortage of qualified teachers. CAPT focuses on mispronunciation detection and diagnosis (MDD) - the former decides whether the learner's articulation is correct or incorrect, while the latter identifies the specific error(s) to generate corrective feedback and facilitate learning.

Our previous work [1–7] devoted much effort in the case of MDD that L2 learners utter English speech following the prompts. We first proposed the approach based on forced-alignment using Extended Recognition Networks (ERNs) [1–6], which cover not only the canonical transcriptions but some likely error patterns as well. The ERNs are used to constrain the search space in Viterbi decoding, thus achieve better performance for L2 English speech than free phone recognition. ERNs which serves as a type of phonological model (PM) of L2 speech are trained from the canonical and annotated transcriptions. In [7], an Acoustic-Phonological Model (APM) is proposed to incorporate the AMs and PMs. Experiments showed that the APM achieved an accuracy of about 83% and a correctness of about 89%. It significantly outperformed the ERN approach whose correctness is about 76%.

Few previous work in CAPT paid attention to the cases that L2 learners speak English without any prompts. In such cases, we have to rely on free phone recognition for L2 English speech. MDD for these cases can be conducted by recognizing the phones and words uttered by L2 learners and comparing the recognized phones with the canonical transcriptions of recognized words.

A typical automatic speech recognition (ASR) system uses hidden Markov models (HMMs) to model the sequential structure of speech signals [8]. Traditionally, Gaussian mixture models (GMMs) are used as parts acoustic models (AMs) to estimate the conditional distribution of speech signal spectrum for each HMM state. Apart from AMs, we need to estimate the state transition probabilities within each phone and the transition probabilities over phones, i.e., phone language models (LMs). These two kinds of transition probabilities can be unified by a single state-transition model (STM). If we aim to recognize words, we should also build a word LM.

Recently, due to the development of highly effective machine learning techniques in ASR like Deep Neural Networks (DNNs) [9, 10], DNNs are used to replace GMMs as part of AMs and achieved significant improvements [11–14]. Many derivative types of DNNs, such as deep Convolutional Neural Networks (CNNs) [15–18] and deep Recurrent Neural Networks (RNNs) [18–20], also achieved impressive improvements. In [18], an ensemble deep learning is used to integrate different kinds of DNNs. Their phone recognition error rates over the TIMIT corpus are below 20% [11, 16–18].

At the same time, DNN-based methods have also shown success on learning word LMs. Early research showed that feed-forward neural networks [21–23] and RNNs [24] can yield better perplexity and word error rate compared with traditional n-gram LMs. With more hidden layers, DNN-based LMs were reported to achieve further improvement, which are competitive with the state-of-the-art LM techniques [25]. As STMs are much simpler than the word LMs, we believe DNNs can be applied to STMs.

Although both AMs and STMs can use DNNs, they are generally trained independently. That is, AMs are trained without considering the preceding or succeeding state sequence. This is based on the assumption that the current acoustic features x_t only depends on the current state s_t. In addition, STMs are trained over the whole corpus and do not consider the concrete

acoustic realization. With such independence assumption, contextual information is lost.

The situation is similar to the training of AMs and PMs in CAPT, which are generally assumed to be independent of each other. This assumption cause the loss of contextual information. Integrating these two kinds of models into an APM gains a significant improvement [7]. Inspired by the APM, we propose a joint Acoustic-State-Transition Model (ASTM) which uses a multi-distribution DNN to integrate AMs and STMs. To calculate the posterior probabilities of states, we not only consider the corresponding acoustic feature, but also their preceding states. To incorporate acoustic features, as well as preceding state sequence (encoded as binary vectors), a multi-distribution DNN is used in this work. Multi-distribution DNNs have been applied to speech synthesis [26, 27], lexical stress detection [28] and mispronunciation detection and diagnosis [7]. Similar to traditional DNNs, they are also constructed by stacking up multiple Restricted Boltzmann Machines (RBMs) from bottom up. This involves running a layer-by-layer unsupervised pre-training algorithm [9, 10], followed by fine-tuning the pre-trained network using back-propagation [29]. Excluding the bottom RBM, all the other ones are traditional Bernoulli RBM (B-RBM), whose hidden and visible units are binary. The bottom RBM is a type of mixed Gaussian-Bernoulli RBM (GB-RBM), whose hidden units are binary while visible units maybe Gaussian or binary.

For the sake of clarity and comparison, we first implement conventional free phone recognition, which is used as our baseline system. A monophone AM and a trigram STM are built, both of which use DNNs. Our major work in this paper is to propose an ASTM. The rest of the paper is organized as follows: Section 2 describes the free phone recognition for L2 English speech; Section 3 introduces the ASTM; Section 4 and 5 present the experimental results and conclusions, respectively.

2. Free Phone Recognition for L2 English Speech (Baseline)

To realize free phone recognition for L2 English, a monophone Acoustic Model (AM) and a trigram State Transition Model (STM) are built, both of which use DNNs.

2.1. Acoustic Model (AM)

The speech is sampled at 16 kHz. To compensate for the high-frequency part of speech signal, a pre-emphasis filter is applied to the speech, whose transfer function is $1-0.97z^{-1}$. Then Fast Fourier Transform is performed in a 25-ms Hamming window with a 10-ms frame shift. Finally, a set of 13 MFCC features are computed per 25-ms frame. Cepstral Mean Normalization is done for each utterance and the features are further scaled to have zero mean and unit variance over the whole corpus.

The diagram of our acoustic model is shown in Fig. 1a. In our experiments, we use 17 frames (1 current, 8 before and 8 after) of MFCCs as the input features, thus there are 221 Gaussian units in the bottom of the DNN. Above the bottom layer, there are 4 hidden layers and each of them has 256 units. For the top layer, there are 90 units generating the posterior probabilities for all the 90 phone-states.

To obtain the 90 phone-states, we first divide each annotated phone equally into three parts to train the AM. Based on the 48-phone set following [8] and 3 states per phone, there are in total of 144 phone-states in the output layer of the DNN. With this trained AM, we performed forced-alignment of the entire corpus based on the annotated phone transcription and merged the states with low occurrence into their neighboring states of the same phones. With the new phonetic boundaries, we re-trained the AM. These two steps were repeated until we had a 90-phone-state set.

2.2. State-Transition Model (STM)

To generate the probabilities of phone-state transition, we build a trigram STM, whose diagram is shown in Fig. 1b. In this work, there are 14 binary input units indicating the previous 2 phone-states, as each phone-state is encoded with 7 bits. Above the bottom layer, there are 4 hidden layer and each of them has 128 units. For the top softmax output layer, there are 90 units generating the phone-state transition probabilities.

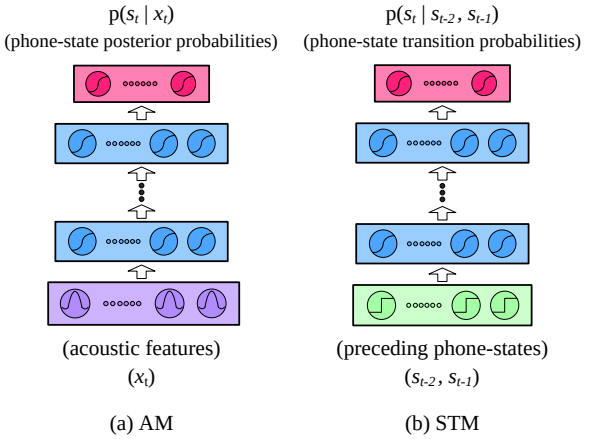

Figure 1: *Diagrams of the monophone Acoustic Model (AM) and trigram State Transition Model (STM).*

2.3. Viterbi decoding using AM and STM

In Viterbi decoding, the phone-state sequence with the highest posterior probability is determined as the recognized phone-state sequence, as given in Eq. (1):

$$\hat{\boldsymbol{s}} = \arg\max_{\boldsymbol{s}} p(\boldsymbol{s} \mid \mathbf{x}) \tag{1}$$

where $\mathbf{x}$ is the sequence of acoustic feature vectors, $\boldsymbol{s}$ denotes a possible phone-state sequence.

The posterior probability of $\boldsymbol{s}$ given $\mathbf{x}$ is:

$$\begin{aligned} p(\mathbf{s} \mid \mathbf{x}) &= p(s_1|\mathbf{x})p(s_2|s_1,\mathbf{x})\cdots p(s_t|s_1,\cdots,s_{t-1},\mathbf{x})\cdots \\ &\approx p(s_1|x_1)p(s_2|s_1,x_2)\cdots p(s_t|s_{t-2},s_{t-1},x_t)\cdots \end{aligned} \tag{2}$$

where x_t is the acoustic feature vector of the t^{th} frame, s_t denotes the phone-state at the t^{th} frame. Note that we use a trigram STM and x_t has a context windows of (8+1+8) frames

Applying Bayes Theorem, we have:

$$\begin{aligned} p(s_t|s_{t-2},s_{t-1},x_t) &= \frac{p(s_t)p(s_{t-2},s_{t-1},x_t|s_t)}{p(s_{t-2},s_{t-1},x_t)} \\ &\approx \frac{p(s_t)p(s_{t-2},s_{t-1}|s_t)p(x_t|s_t)}{p(s_{t-2},s_{t-1})p(x_t)} \\ &= p(s_t|s_{t-2},s_{t-1})\frac{p(s_t|x_t)}{p(s_t)} \end{aligned} \tag{3}$$

From Eq. (2) and Eq. (3), we have:

$$p(\mathbf{s}\,|\,\mathbf{x}) \approx p(s_1|x_1)p(s_2|s_1)\frac{p(s_2|x_2)}{p(s_2)}\cdots$$
$$p(s_t|s_{t-2},s_{t-1})\frac{p(s_t|x_t)}{p(s_t)}\cdots \quad (4)$$

where $p(s_t|x_t)$ is the phone-state posterior probability from the AM, $p(s_t|s_{t-2},s_{t-1})$ is the phone-state transition probability from the STM and $p(s_t)$ is the phone-state prior probability estimated from training data.

3. Acoustic-State-Transition Model (ASTM)

Figure 1 shows that the structures of AM and STM are similar, and their main difference is the input features. In this subsection, we try to integrate the monophone AM and the trigram STM.

3.1. Implementation of ASTM using MD-DNN

The structure of our ASTM is shown in Figure 2, which is a multi-distribution DNN [7,26–28]. There are 221 Gaussian and 14 binary visible units in the bottom of the DNN. The other layers are similar to the AM in Fig. 1. The ASTM can be represented by $p(s_t|s_{t-2},s_{t-1},x_t)$.

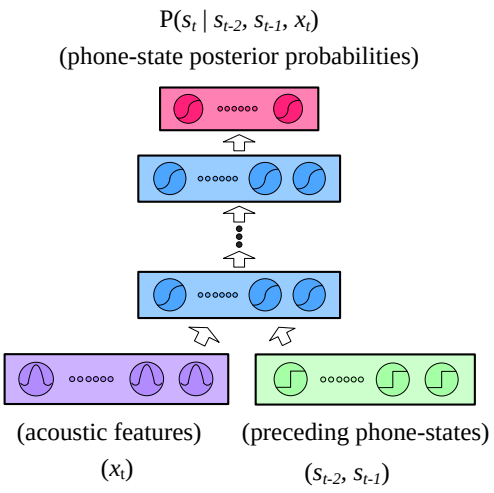

Figure 2: *Diagrams of the Acoustic-State-Transition Model (ASTM).*

3.2. Viterbi decoding using ASTM

The conventional approach in last subsection cannot compute $p(s_t|s_{t-2},s_{t-1},x_t)$, and hence uses its approximation in Eq. (3). It assumes that x_t only depends on s_t and is independent of its preceding states (s_{t-2},s_{t-1}). This is not true actually and thus contextual information is lost.

The ASTM calculates the posterior probability of $p(s_t|s_{t-2},s_{t-1},x_t)$. In this approach using ASTM, we do not need to train the AM and STM separately, nor estimate the prior probability $p(s_t)$, which may also cause problems, especially when there is insufficient data for training.

With ASTM, we can directly use Eq. (2) to approximate $p(\mathbf{s}|\mathbf{x})$, instead of using Eq. (4). From Eq. (2), we observe that the ASTM can be easily extended to use a longer contextual window, e.g., considering 5 preceding states, if sufficient data is available for training.

4. Experiments

4.1. Corpora

Our experiments are based on the TIMIT and CU-CHLOE (Chinese University Chinese Learners of English) corpora. The CU-CHLOE corpus contains 110 Mandarin speakers (60 males and 50 females) and 100 Cantonese speakers (50 males and 50 females). There are five parts in CU-CHLOE: confusable words, minimal pairs, phonemic sentences, the Aesops Fable "The North Wind and the Sun" and prompts from TIMIT. Excluding the TIMIT prompts, all the other parts are labeled by trained linguists, which account for about 30% of the whole CHLOE data.

The details of the TIMIT and CU-CHLOE corpora are shown in Table 1. For the TIMIT corpus, the training and test sets come from the original training and core test sets; while the development set is the full test set excluding the data presented in the core test set. For CU-CHLOE, we randomly split the corpus by speakers into a training set, a development set and a test set, whose rates are 70%, 10% and 20%, respectively.

Table 1: *Details of corpora used in our experiments.*

	TIMIT			CU-CHLOE		
	Train	Dev.	Test	Train	Dev.	Test
Speakers	462	144	24	147	22	42
Unlabeled	—	—	—	67h	—	—
Labeled	3h	1h	0.16h	26h	4h	7.5h

To transcribe the L2 English speech of CU-CHLOE, we first built acoustic models using HTK [30] based on the TIMIT corpus to align the canonical transcriptions with the L2 English speech. Then our linguists annotated the speech with actual pronunciations. To save labor, our linguists mainly focused in labeling (modifying) the phone sequences, thus the accuracy of the phone boundaries is not high. Hence, these annotated phone sequences should be re-aligned using the AM described in Section 2. We implement the forced-alignment and train the AM iteratively until the AM's performance improvement levels off, which is assessed via running phonetic recognition on the test set of the CU-CHLOE corpus.

4.2. DNN training

The DNNs training for AM and STM in this work is similar to [6, 7, 28]. In the pre-training stage, we try to maximize the log-likelihood of RBMs. The one-step Contrastive Divergence (CD) [9] is used to approximate the stochastic gradient. 20 epochs are performed with a batch size of 256 frames. In the fine-tuning stage, the standard back-propagation (BP) algorithm [29] is performed. A dropout [17, 31–33] rate of 10% is used in this work. To speed up the BP training process, a technique of asynchronous stochastic gradient descent (ASGD) [34] is used to parallelize computing.

However, we are facing a problem in training the ASTM with the traditional training methods. When combining the

acoustic model and language model together, the data sparse issue is also introduced into ASTM learning. Missing a large portion of the trigram state sequences in the training set makes it even more difficult to estimate the probabilities that are not on the optimal path in the decoding network. To overcome this problem, a randomized BP (RBP) training method is introduced to achieve better generalization on the unseen trigram state sequences. The true preceding state sequence (s_{t-2}, s_{t-1}) is randomly replaced with a random one (noise) at a fixed probability (typically 80%). When the replacement happens, we also apply a reduced weight (typically 0.5) on the output target labels during the BP training to indicate a low confidence on the false (impostor) trigrams. This training procedure is to ensure that the ASTM output mainly depends on the input acoustic features, in case the input preceding states are incorrect.

4.3. Experimental results with basic configurations

As this is our first attempt to realize free phone recognition for L2 English speech, we first build a simple system with basic configurations using the TIMIT corpus, which is much smaller than our CU-CHLOE corpus. In the fine-tuning stage, the standard BP is performed based on Minimum Mean Square Error (MMSE). The dropout technique is disabled here. The training process is conducted on the TIMIT training set with many epochs until its performance improvement levels off, which is evaluated on the development set.

The performance of phone recognition for this basic systems are shown in Table 2, which are assessed on the TIMIT core test set. The correctness and accuracy are calculated by the following equations [30]:

$$Corr. = \frac{N - S - D}{N}\ ; \qquad Acc. = \frac{N - S - D - I}{N}$$

where N is the total number of labels; while S, D and I denote for the counts of substitution, deletion and insertion errors, respectively.

It shows that the baseline system with separate AM and STM only achieves an accuracy of about 51.7%. Our proposed ASTM with 256 nodes in each hidden layer obtains an accuracy of 64.1%. Note that the ASTM has a worse correctness than the baseline system, whose values are 67.8% and 71.5%, respectively. This is mainly due to more deletion generated by the ASTM. With the help of randomized BP (see next subsection), the ASTM obtained better performance on both correctness and accuracy, whose values are 74.8% and 70.2% respectively. It means that integrating AM and STM achieves better performance, i.e., Eq. (2) gives a better approximation of $p(\mathbf{s}|\mathbf{x})$ than Eq. (4).

Table 2: *Performance of phone recognition with basic configurations.*

	Correctness	Accuracy
AM (256) & STM (128)	**71.45%**	51.68%
ASTM (256)	67.80%	**64.16%**

Note: The above DNNs are only trained on the TIMIT corpus; The starting and ending silences are not counted in this paper's experiments.

4.4. Contribution of randomized BP

Table 3 presents the contribution of randomized BP, without which the ASTM only obtains an accuracy of 64%. Employing randomized BP results an improvement of about 4%. Note that the correctness of the ASTM with randomized BP is better than that of using separate AM and STM.

Table 3: *Performance of ASTM with and without Randomized BP.*

	Correctness	Accuracy
without Randomized BP	67.80%	64.16%
with Randomized BP	**73.25%**	**68.11%**

Note: Both the above DNNs of ASTM have 4 hidden layers and each hidden layer has 256 nodes.

4.5. Contribution of more hidden units

Table 4 shows the performance of phone recognition with more hidden nodes. Increasing the units of each hidden layer from 256 to 512 gains an improvement of about 2% in accuracy.

Table 4: *Performance of phone recognition with more hidden nodes.*

	Correctness	Accuracy
ASTM (256)	73.25%	68.11%
ASTM (512)	**74.84%**	**70.15%**

4.6. Results of ASTM with further configurations

Due to the effectiveness of ASTM, we will focus on further fine-tuning its parameters and leave the separate AM and STM models behind. The dropout technique as described in subsection 4.2 is employed. The minimum cross entropy error is used to replace the MMSE as the objective of DNN training. The randomized BP is also used to replace the standard BP.

Table 5 shows that the ASTM trained on the TIMIT corpus achieves an accuracy of about 72.4%. Although it is generally more challenging to recognize the non-native speech, a similar performance is obtained on the CU-CHLOE corpus. The main reason is that there are more data in the CU-CHLOE corpus, which contains about 26 hours of labeled data and 67 hours of unlabeled data for training; while the TIMIT corpus has only 3 hours of labeled data for training (see Table 1).

Note that our performance on the TIMIT corpus is still lower than the performances published in [11,16–18]. The main reasons are that we only try some basic configurations of DNN parameters (e.g., there are only 4 hidden layers and each hidden layer has only 256 or 512 nodes) and only use monophone acoustic models.

Table 5: *Performance of phone recognition using ASTM on different corpora.*

Corpus	Correctness	Accuracy
TIMIT	75.38%	72.37%
CU-CHLOE	74.51%	72.00%

5. Conclusions and Future Work

This paper investigates the use of Multi-Distribution Deep Neural Networks (MD-DNNs) for integrating acoustic and state-transition models in free phone recognition of L2 English speech. We first implement a baseline system using separate Acoustic Model (AM) and State-Transition Model (STM). As these two models are trained independently, hence context information maybe lost. In order to integrate these two models, we propose a joint Acoustic-State-Transition Model (ASTM), whose features cover the MFCC features as well as the preceding phone-state sequence (encoded as a binary vector). Due to the different kinds of distribution of these features, a multi-distribution DNN is used in this work. Experimental results with basic parameter configurations show that the ASTM obtains a phone accuracy of about 68% on the TIMIT data. It is better than the system of using separate AM and STM, whose accuracy is only about 52%. Further configuring the ASTM achieves an accuracy of about 72% on the TIMIT data. Similar performance is obtained if we train and test the ASTM on the CU-CHLOE corpus.

The success of integrating the AM and STM motivates us to further integrate the STM with our proposed Acoustic-Phonological Model (APM) [7] in the future. The APM, which was developed for the cases of MDD that the prompts for L2 learners are known in advance, achieved a phone accuracy of about 83%. This performance was much better than that of using separate AM and STM, whose accuracy was only about 58%. We conjecture that the joint Acoustic-Phonological-State-Transition Model (APSTM) will be effective in performance improvement.

6. Acknowledgements

The work is partially supported by a grant from the HKSAR Government GRF (project number 415511).

7. References

[1] A. M. Harrison, W.-K. Lo, X.-j. Qian, and H. Meng, "Implementation of an extended recognition network for mispronunciation detection and diagnosis in computer-assisted pronunciation training," in *Proc. IEEE SLaTE Workshop*, 2009.

[2] W.-K. Lo, S. Zhang, and H. Meng, "Automatic derivation of phonological rules for mispronunciation detection in a computer-assisted pronunciation training system," in *Proc. INTERSPEECH*, 2010.

[3] X. Qian, F. K. Soong, and H. Meng, "Discriminative acoustic model for improving mispronunciation detection and diagnosis in computer-aided pronunciation training (CAPT)," in *Proc. INTERSPEECH*, 2010.

[4] X.-j. Qian, H. Meng, and F. Soong, "Capturing L2 segmental mispronunciations with joint-sequence models in computer-aided pronunciation training (CAPT)," in *Proc. ISCSLP*, 2010.

[5] ——, "On mispronunciation lexicon generation using joint-sequence multigrams in computer-aided pronunciation training (CAPT)." in *Proc. INTERSPEECH*, 2011.

[6] ——, "The use of dbn-hmms for mispronunciation detection and diagnosis in L2 English to support computer-aided pronunciation training," in *Proc. INTERSPEECH*, 2012.

[7] K. Li and H. Meng, "Mispronunciation detection and diagnosis in L2 english speech using multi-distribution deep neural networks," in *Proc. ISCSLP*, 2014.

[8] K.-F. Lee and H.-W. Hon, "Speaker-independent phone recognition using hidden markov models," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 37, no. 11, pp. 1641–1648, 1989.

[9] G. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, pp. 1527–1554, 2006.

[10] G. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *SCIENCE*, vol. 313, pp. 504–507, 2006.

[11] A.-r. Mohamed, T. N. Sainath, G. Dahl, B. Ramabhadran, G. E. Hinton, and M. A. Picheny, "Deep belief networks using discriminative features for phone recognition," in *Proc. ICASSP*, 2011.

[12] A.-r. Mohamed, G. E. Dahl, and G. E. Hinton, "Acoustic modeling using deep belief networks," *IEEE Trans. on Audio, Speech and Language Proc.*, vol. 20, pp. 14–22, 2012.

[13] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large vocabulary speech recognitio," *IEEE Trans. on Audio, Speech and Language Proc.*, vol. 20, pp. 30–42, 2012.

[14] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[15] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition," in *Proc. ICASSP*, 2012.

[16] T. N. Sainath, A.-r. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for lvcsr," in *Proc. ICASSP*, 2013.

[17] L. Deng, O. Abdel-Hamid, and D. Yu, "A deep convolutional neural network using heterogeneous pooling for trading acoustic invariance with phonetic confusion," in *Proc. ICASSP*, 2013.

[18] L. Deng and J. C. Platt, "Ensemble deep learning for speech recognition," in *Proc. INTERSPEECH*, 2014.

[19] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. ICASSP*, 2013.

[20] L. Deng and J. Chen, "Sequence classification using the high-level features extracted from deep neural networks," in *Proc. ICASSP*, 2014.

[21] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *The Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2003.

[22] H. Schwenk and J.-L. Gauvain, "Training neural network language models on very large corpora," in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2005, pp. 201–208.

[23] H. Schwenk, "Continuous space language models," *Computer Speech & Language*, vol. 21, no. 3, pp. 492–518, 2007.

[24] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, "Recurrent neural network based language model." in *Proc. of INTERSPEECH*, 2010.

[25] E. Arisoy, T. N. Sainath, B. Kingsbury, and B. Ramabhadran, "Deep neural network language models," in *Proceedings of the NAACL-HLT 2012 Workshop: Will We Ever Really Replace the N-gram Model? On the Future of Language Modeling for HLT*. Association for Computational Linguistics, 2012, pp. 20–28.

[26] S. Kang, X. Qian, and H. Meng, "Multi-distribution deep belief network for speech synthesis," in *Proc. ICASSP*, 2013.

[27] S. Kang and H. Meng, "Statistical parametric speech synthesis using weighted multi-distribution deep belief network," in *Proc. INTERSPEECH*, 2014.

[28] K. Li and H. Meng, "Lexical stress detection for L2 English speech using deep belief networks," in *Proc. INTERSPEECH*, 2013.

[29] D. E. Rumelhart, G. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Natrue*, vol. 323, pp. 533–536, 1986.

[30] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey *et al.*, "The htk book (for htk version 3.4)," 2006.

[31] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[32] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. ICASSP*, 2013.

[33] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Y. Ng, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567v2*, 2014.

[34] S. Zhang, C. Zhang, Z. You, R. Zheng, and B. Xu, "Asynchronous stochastic gradient descent for dnn training," in *Proc. ICASSP*. IEEE, 2013.