LEARNING AUXILIARY CATEGORICAL INFORMATION FOR SPEECH SYNTHESIS BASED ON DEEP AND RECURRENT NEURAL NETWORKS

Zhengqi Wen[1], Kehuang Li[2], Zhen Huang[2], Jianhua Tao[1], Chin-Hui Lee[2]*

[1] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, 100190, Beijing, China

[2]School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA. 30332-0250, USA

{zqwen, jhtao}@nlpr.ia.ac.cn, {kehle, huangzhenee}@gatech.edu, chl@ece.gatech.edu

ABSTRACT

We proposed an auxiliary categorization framework for training speech synthesis systems using deep neural networks (DNNs) and recurrent neural networks (RNNs). The adopted artificial neural networks (ANNs) are regression models comprising a few hidden layers and an affine-transform layer for transforming the contextual features into a set of speech synthesis parameters. In order to incorporate categorization information into training ANNs, similar to DNN-based speech recognition, the proposed approach stacks a secondary classification layer on top of the hidden layers for the regression ANN and trained it together with the primary affine-transform. Four categorization tasks, for classification of voicing, phonation position, phone identity and hidden Markov model state, are considered. The experimental results show that the proposed framework can reduce the root mean square error (RMSE) of the generated log fundamental frequency by about 10.8% and 4.3% for DNN and RNN based synthesis systems, respectively. With the extra classification layers, subjective listening tests also favor DNN and RNN generated speech by about 24% and 15%, respectively, over the ANN baselines without using any categorical information.

Index Terms—speech synthesis, categorization, deep neural network, recurrent neural network

1. INTRODUCTION

Artificial neural Network (ANN) based technologies [1] have been widely used with promising results in a lot of research areas, such as automatic speech recognition (ASR) [2], computer vision [3] and natural language processing [4]. It has also been adopted in speech synthesis. In [5], Kang *et al.* used a deep belief network (DBN) to model a joint distribution of contextual and speech features. In [6], Ling *et al.* replaced the Gaussian mixture model (GMM) with DBN in the hidden Markov model (HMM)-based speech synthesis system [7]. In [8], Zen *et al.* proposed a deep neural network (DNN) based speech synthesis framework by mapping from the contextual features to the speech features and further proposed to use deep mixture density network (MDN) to model the mean and variance of the speech parameters [9]. In [10], Fan *et al.* adopted a recurrent neural network (RNN) with bi-direction long short term memory (BLSTM) [11] [12] to model the relationship between the contextual features and the speech parameters. ANN-based experimental results have also demonstrated improvements over HMM-based speech synthesis.

In general, the typical ANN in speech synthesis is used as a regression tool for mapping from the contextual features to the speech parameters. When compared with the HMM-based speech synthesis systems, the ANNs are learned with little discriminative information in the output layer because the decision trees [13] used in HMM-based speech synthesis for categorizing different classes of the speech parameters have been removed from the ANN-based speech synthesis systems. Leveraging upon the recent successes in DNN-based ASR [2] and DNN-based automatic speech attribute transcription (ASAT) [14, 15] a key motivation in this study is to facilitate an incorporation of some categorical information in decision trees into training ANN-based speech synthesis systems. It is realized by an auxiliary categorization framework with an additional classification layer on top of the hidden layers of the regression DNNs. This classification layer is trained together with the affine-transform layer in multi-task learning (MTL) [16] which has already been used in speech synthesis in [17].

In this paper we extend [17] to the use of different error ratios between the primary and secondary objective functions. We also propose to incorporate categorical information used in constructing the decision trees into ANN training. To construct the additional layer, four attribute categorization tasks, i.e., classification of voicing, phonation position, phone identity and HMM state, are considered and the abovementioned error ratios are also verified.

Both objective measures and subjective listening tests have been evaluated. It is found that, for ANN-based speech synthesis, the additional classification layer reduces the root mean square error (RMSE) of the generated log fundamental frequency (LF0) from 0.145 to 0.132 for DNN based and from 0.138 to 0.132 for RNN based synthesis systems, respectively. Moreover, the ANN-generated speech with the classification layers is also preferred to that generated by conventional ANN-based synthesis systems with no additional classification layers by about 24% and 15% for DNN and RNN based speech synthesis, respectively.

2. PROPOSED ANN-BASED SPEECH SYNTHESIS

2.1. Classical ANN-based Speech Synthesis

A typical deep neural network (DNN) for speech synthesis shown in Fig. 1 is constituted with a few hidden layers and an output layer. The hidden layers can be considered as a nonlinear feature extractor from the input contextual features. The output layer stacked on the top of the hidden layers is an affine-transform layer for generating speech parameters from the nonlinearly transformed features. To train the DNN, the hidden layers are constituted by the pre-trained RBMs [18] with the contrastive convergence (CD) criterion [19]. The input of the first input layer is normalized as a Gaussian with zero mean and unity variance so the pre-trained

[*] work done while visiting Georgia Tech in 2014-2015

hidden layers are stacked as the first Gaussian-Bernoulli RBM and the rest Bernoulli-Bernoulli RBMs.

This topology is also used in recurrent neural network (RNN) based speech synthesis. But the hidden layers will be stacked by at least one recurrent layer, for example bidirectional long short term memory (BLSTM) [11]. The long short term memory (LSTM) [12] shown in Fig. 1 is proposed for overcoming the gradient vanishing problem for the conventional RNN.

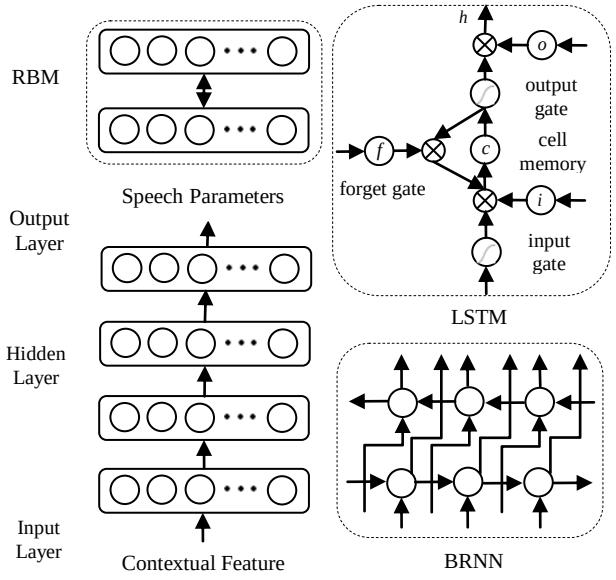


Figure 1: Artificial neural network based speech synthesis. Left: restricted Boltzmann Machine (RBM) and deep neural network; right: long short term memory (LSTM) and bidirectional recurrent neural network (BRNN).

2.2. Proposed Classification Layer

ANN-based parameter learning for speech synthesis is often cast as a regression problem and ANN is used to construct a mapping function directly from the contextual features to the speech parameters. Thus this regression function is usually learned with little discriminative information in the output layer. To alleviate this problem, decision trees [13] is adopt in the HMM-based speech synthesis system to classify the input contextual features and learn parameters for every small node. Besides this, the function will also introduce the over-smoothing problem because the difference between the generated speech parameters in the output layer is only decided by the input contextual features. To overcome this problem, Zen *et al.* adopted the maximum likelihood parametric generation (MLPG) algorithm [20] to get the speech parameters with the first and second order derivatives.

The two problems listed above could also be alleviated by adding another output layer for categorization. The additional classification layer is learned together with the affine-transform layer. The error signal of the categorization tasks will be back-propagated to update the hidden layers' weights. Therefore the hidden layers will be learned with the discriminative information for different categorization attributes. Furthermore, this additional classification layer will also help to overcome the over-smoothing problem in discriminative learning. The proposed framework for the ANN-based speech synthesis is demonstrated in Fig. 2. A detailed description about how to learn the additional classification layer is given in the followings.

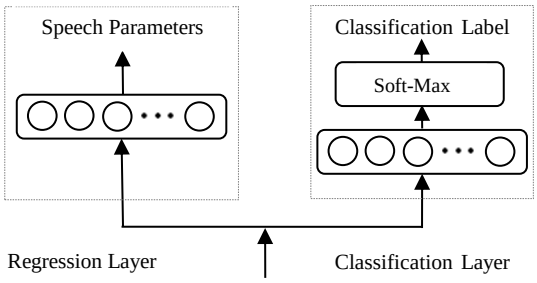


Figure 2: The framework of the output layers. Left: an affine-transform layer for generating speech parameters; right: a soft-max layer together with an affine-transform for classification.

For regression, the mean square error (MSE) in Eq. (1) is used as the criterion to be minimized to fine-tune the ANN parameters:

$$\mathrm{D}_{\mathrm{MSE}}(\hat{\mathrm{y}},\mathrm{y}) = \frac{1}{\mathrm{T}}\sum_{\mathrm{t}=1}^{\mathrm{T}}(\hat{\mathrm{y}}-\mathrm{y})^2 \quad (1)$$

where T is the frame number, y is the target speech feature vector and $\hat{\mathrm{y}}$ is the predicted speech feature vector as follow:

$$\hat{\mathrm{y}} = \tilde{\mathrm{g}}(\mathrm{W}_{\mathrm{A}},\mathrm{b}_{\mathrm{A}},\mathrm{h}) \quad (2)$$

where $\tilde{\mathrm{g}}$ is a linear function, $\mathrm{W}_{\mathrm{A}},\mathrm{b}_{\mathrm{A}}$ are the weight matrix and bias vector for the affine-transform layer, h is the output of the hidden layers.

As for classification, a soft-max layer is trained with the cross entropy (CE) criterion [21] in Eq. (3) as follow:

$$\mathrm{D}_{\mathrm{CE}}(\hat{\mathrm{s}},\mathrm{s}) = -\sum_{\mathrm{n}=1}^{\mathrm{N}}\sum_{\mathrm{t}=1}^{\mathrm{T}}\mathrm{s}\log\hat{\mathrm{s}} \quad (3)$$

where N is the sentence number, T is the frame number, s is the target label for the categorization tasks, and $\hat{\mathrm{s}}$ is the generated label as follow:

$$\hat{\mathrm{s}} = \frac{\exp(\tilde{\mathrm{g}}(\mathrm{W}_{\mathrm{S}},\mathrm{b}_{\mathrm{S}},\mathrm{h}))}{\sum\exp(\tilde{\mathrm{g}}(\mathrm{W}_{\mathrm{S}},\mathrm{b}_{\mathrm{S}},\mathrm{h}))} \quad (4)$$

A stochastic gradient descent (SGD) algorithm [22] is performed in mini-batches to update the parameters in Eq. (5).

$$(\mathrm{W},\mathrm{b}) \leftarrow (\mathrm{W},\mathrm{b}) + \lambda\frac{\partial \mathrm{D}}{\partial(\mathrm{W},\mathrm{b})} \quad (5)$$

where λ is the learning rate.

The outputs of the back-propagation [23] algorithm in Eq. (1) and Eq. (2) are added together with an error ration in Eq. (6) as the input for back-propagating the hidden layers.

$$\mathrm{D}(\hat{\mathrm{y}},\mathrm{y},\hat{\mathrm{s}},\mathrm{s}) = \mathrm{D}_{\mathrm{MSE}}(\hat{\mathrm{y}},\mathrm{y}) + \alpha\times\mathrm{D}_{\mathrm{CE}}(\hat{\mathrm{s}},\mathrm{s}) \quad (6)$$

where α is an error ratio.

3. CATEGORIZATION TASKS

Decision trees [13] make a sharable structure for every state in the HMM-based speech synthesis system. It splits the whole database into several nodes by asking a number of questions, such as phonemes identity, left or right contextual information and voiced/unvoiced labels. These questions help the decision trees to split the space of the speech parameters into small groups in order to learn more accurate parameters. Due to differences between the HMM and ANN, it is very hard to directly incorporate all the related questions into ANN training. Here we only consider four types of questions for constructing the classification layer.

The first one is the voiced/unvoiced label. Due to the different vibrating state of glottis, the speech frames' spectra can be easily split into two groups: non-zero fundamental frequency with a harmonic structure and zero fundamental frequency with a noisy

structure. This additional classification layer therefore enhances the hidden layers to describe the differences between voiced and unvoiced frames.

The second one is the phone identity. In the HMM-based speech synthesis system, decision trees are constructed for every HMM state and the phone identity questions are asked in parallel with other contextual information. It means that the constructed decision trees are shared across all the phones. It is a cause of the over-smoothing problem existing in the HMM-based speech synthesis system. To alleviate this problem in ANN-based speech synthesis, we stack a phone identity classification layer on top of the hidden layers to re-enforce the phone identity's discrimination in the hidden layers.

The third one is the phonation position. Every phone phonates in different positions of the vocal tract. So the phones can also be categorized into small groups. According to the knowledge in phonetics, the syllable initials and finals in Mandarin can be split into 15 groups as listed in Table 1. This layer will group the phones and learn the groups in a discriminative manner.

Table 1: Mandarin Initials and finals based on phonation position.

labial	bilabial	p b m
	labiodental	f
coronal	dental	t d n l
	alveolar	z c s ii
velar		k g h
retroflex		zh ch sh r iii
alveolo-palatal		j q x
low	front	ai an
	central	a
	back	ang ao
middle	front	ei en
	central	eng er
	back	e o ong ou
high	front	i ia ian iang iao ie in ing iong iou v van ve vn
	back	u ua uai uan uang uei uen ueng uo

The fourth one is the HMM state. In the HMM-based speech synthesis system, HMM state occupied by a number of speech frames represents a short stationary part of a speech signal. So every speech frame can be categorized into a HMM state. This information can be obtained from the decision tree of the HMM-based speech synthesis system directly.

4. EXPERIMENTS AND DISCUSSION

In this section, we first describe the experimental configuration in Section 4.1. We then introduce the baseline systems in Section 4.2. The effectiveness of the proposed framework in ANN-based speech is next evaluated in Section 4.3 in order to compare with the regression method in [10] and finally we evaluate the results of the four different categorization tasks.

4.1. Experiment Setup

The corpus used in the following experiments was from a female talker speaking Mandarin for about seven hours. The contextual features [7] are represented as a vector. The speech parameters used in these experiments are line spectral pair (LSP) [24] with dynamic features as the spectral parameters and log fundamental frequency (LF0) as the excitation parameter. The spectral parameters were extracted from the STRAIGHT spectrum [25]. Before taken into ANN training, these two features were both normalized into a Gaussian distribution with zero mean and unity variance. The KALDI toolkit [26] was used for ANN training. The topology of the DNN used in the following experiments contains four hidden layers with 3072 units at each hidden layer and the RNN contains two BLSTM layers with 512 units.

The quality of the synthesized speech was verified in two ways. The first was through two objective measures, namely the root mean square error (RMSE) between the generated and the original speech parameters and log spectral distance (LSD) [27] between the generated and the original waveforms. The other was a subjective measure in terms of the ABX preference scores [28] in naturalness. In the preference tests, subjects were asked to listen to two versions of synthesized speech and choose one which sounds much better than the other. The better one will get a preference score of "1" or no preference (N/P) score of "1". The final scores were calculated by the mean value of the scores given by the 15 listeners who are working in some speech technology areas.

4.2. Baseline System with No Classification Layers

There are three baseline synthesis systems being evaluated here, namely HMM-based speech synthesis (HTS-SYN), DNN-based speech synthesis (DNN-SYN) and RNN-based speech synthesis (RNN-SYN) systems. They were trained with the same inputs to produce the desired outputs. The objective and subjective experimental results are shown in Tables 2 and 3, respectively.

In Table 2, the objective measures were improved from HMM-based to ANN-based speech synthesis, especially for RNN where the LF0's RMSE was reduced by about 6.7% (from 0.148 in the top row for HTS-SYN to 0.138 in the bottom row for RNN-SYN). The preference scores also confirm the effectiveness of ANN-based speech synthesis. Even with no classification, RNN-based speech synthesis generated the best quality and was preferred to by about 33.33% over HMM-based speech synthesis shown in the second row of Table 3, and by 34.29% over DNN-based speech synthesis shown in the bottom row in Table 3.

Table 2: The RMSE and LSD measure for HMM-based speech synthesis system (HTS-SYN), DNN-based speech synthesis (DNN-SYN) and RNN-based speech synthesis (RNN-SYN) systems.

	LSD	LF0	LSP
HTS-SYN	5.018	0.148	1.119
DNN-SYN	4.932	0.145	1.118
RNN-SYN	4.896	0.138	1.112

Table 3: ABX preference scores with a 0.005 confidence interval for HMM-based speech synthesis (HTS-SYN), DNN-based speech synthesis (DNN-SYN), and RNN-based speech synthesis (RNN-SYN).

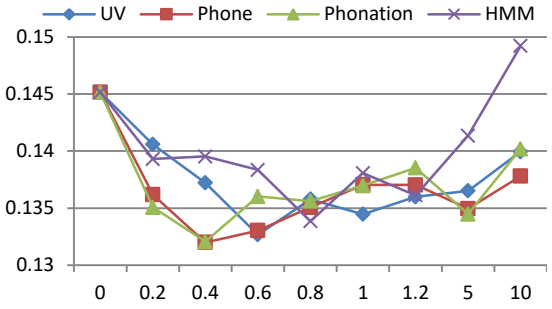
HTS-SYN	DNN-SYN	RNN-SYN	N/P
0.2952	0.3524	-	0.3524
0.2	-	0.5333	0.2667
-	0.1238	0.4667	0.4095

4.3. Classification Tasks

4.3.1. Objective Measure

The error ratio in the objective function in Eq. (6) is crucial for a proper incorporation of the classification layer into ANN training. A series of experiments were carried out to decide which ratio is appropriate for the different categorization tasks. Fig. 3 describes the LF0's RMSE changes with different error ratios for the four different categorization tasks. It reveals that different categories should take different error ratios. According to the preliminary results, the error ratio choices were set to be 0.6, 0.4, 0.4 and 0.8

for classification of the voiced/unvoiced attribute, the phone identity, the phonation position, and the HMM state, respectively. These values will stay the same for the remaining experiments for both DNN and RNN based speech synthesis.



In [10], the target voiced/unvoiced label is generated directly from the output of the regression layer. It is different from our proposed method that a classification layer is added only for the voiced/unvoiced label's classification. The objective measures are shown in Table 4 for DNN-based speech synthesis systems. The Voiced/Unvoiced error is decreased by about 1.56% from the regression-based (UV-R-DNN) to classification-based (UV-C-DNN) method. When compared with Table 2, RMSE and LSD measures are also improved with the help of the classification layers, especially for RMSE of LF0 that was reduced by about 6%.

Table 4: The RMSE, LSD, LSP and V/U (Voiced/Unvoiced) Error for the method in [10] (UV-R-DNN) and our proposed method (UV-C-DNN) where R indicates "Regression" and C indicates "Classification".

	LSD	LF0	LSP	V/U Error
UV-R-DNN	4.983	0.153	1.120	5.449%
UV-C-DNN	4.899	0.133	1.113	3.888%

Three objective measures were compared for DNN and RNN based speech synthesis with the additional classification layer in Tables 5 and 6, respectively. It can be seen that these values for the four different categorization tasks vary very small among themselves. Clearly the results with RNN in Table 6 are slightly better than those with DNN in Table 5, and they are all better than the baseline speech synthesis systems without the classification layers as shown in Table 2.

Table 5: The RMSE and LSD measures for DNN based speech synthesis with different classification tasks.

	LSD	LF0	LSP
Voiced/Unvoiced	4.899	0.133	1.112
Phoneme	4.889	0.132	1.110
Phonetic Feature	4.910	0.132	1.112
HMM State	4.917	0.134	1.113

Table 6: The RMSE and LSD measure for RNN based speech synthesis with different classification layers.

	LSD	LF0	LSP
Voiced/Unvoiced	4.891	0.131	1.110
Phoneme	4.885	0.132	1.108
Phonetic Feature	4.875	0.131	1.109
HMM State	4.881	0.132	1.112

4.3.2. Subjective Preference Scores

Listening tests were also carried out to evaluate the effectiveness of the proposed technique. Table 7 lists the preference scores for the voiced/unvoiced label used in DNN-based speech synthesis at a regression or a classification layer. The score for the classification based method (UV-C-DNN) at 37.8% in the middle column of the bottom row in Table 7 is preferred at 37.78% to the regression based method (UV-R-DNN) in the left column at 8.89%.

Since the differences between the four tasks in Tables 5 and 6 are very small, we only consider the voiced/unvoiced attribute in Table 8 with the best error ratio of 0.6. The preference scores are compared for ANN based speech synthesis with (UV-C-DNN and UV-C-RNN) and without (just DNN and RNN) the classification layer. The results again confirm that speech generated with the classification layer is much preferred to speech synthesis without the classification layer by about 24% (from 0.2266 to 0.4667) for DNN-based speech synthesis at the top row in Table 8, and by about 15% (from 0.1867 to 0.34) for RNN based speech synthesis in the bottom row of Table 8.

Table 7: Preference scores with a 0.05 confidence interval for DNN based speech synthesis systems with regression (UV-R-DNN) or classification (UV-C-RNN) for voiced/unvoiced label.

UV-R-DNN	UV-C-RNN	N/P
0.0889	0.3778	0.5333

Table 8: Preference scores at a 0.05 confidence interval for DNN and RNN based synthesis with and without the classification tasks.

DNN	UV-C-DNN	RNN	UV-C-RNN	N/P
0.2266	0.4667	-	-	0.3067
-	-	0.1867	0.34	0.4733

From these results, it could be concluded that by adding the classification layer on top of the hidden layers to the regression ANNs we could strengthen the ANN's modeling ability to generate better speech parameters from the contextual features.

5. CONCLUSION AND FUTURE WORK

We propose an auxiliary categorization framework for training ANN based speech synthesis systems. An auxiliary classification layer is added on top of the hidden layers in parallel to the affine-transform layer. Four attribute categorization tasks have been considered: namely classification of voiced/unvoiced attribute, phonation position, phone identity and hidden Markov model (HMM) state. Our experimental results show that the proposed framework can generate much better sound than that without the classification layer for speech synthesis systems based on the deep neural network (DNN) and recurrent neural network (RNN).

In this study, only a female voice and four categorization tasks have been included. In our future work, the proposed approach will be tested for male speech synthesis as well. Other categorization tasks for different speech attributes will also be explored and combined together for the hidden layers' back-propagation. Other future work also includes the incorporation of speech recognition attribute features into ANN training of speech synthesis systems.

6. ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China (NSFC) (No.61403386, No.61273288, No.61233009, No.61203258, No.61305003, No. 61332017, No.61375027), and the Major Program for the National Social Science Fund of China (13&ZD189).

7. REFERENCES

[1] G.-E. Hinton., "Learning multiple layers of representation," *Trends in Cognitive Sciences*, vo.11, pp. 428–434, 2007.

[2] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "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.

[3] A. Krizhevsky, I. Sutskever. and G. Hinton., "ImageNet Classification with Deep Convolutional Neural Networks," in *Proc. of NIPS*, pp.1106-1114, 2012.

[4] R. Collobert and J. Weston, "A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning," In *Proc. of ICML*, pp.160-167, 2008.

[5] S. Kang, X. Qian, and H. Meng, "Multi-distribution deep belief network for speech synthesis", in *Proc. of ICASSP*, pp.7962-7966, 2013.

[6] Z.-H. Ling, L. Deng, and D. Yu,"Modeling spectral envelopes using restricted Boltzmann machines for statistical parametric speech synthesis", in *Proc. of ICASSP*, pp.7825-7829, 2013.

[7] H. Zen, K. Tokuda and A.W. Black, "Statistical parametric speech synthesis", *Speech Communication*, 51(11):1039-1064, 2009.

[8] H. Zen, K. Tokuda, and A. Black, "Statistical Parametric Speech Synthesis," *Speech Communication*, 51(11), 1039-1064, 2009.

[9] H. Zen and A. Senior, "Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis", in *Proc. of ICASSP*, pp. 3844-3848, 2014.

[10] Y.-C. Fan, Y. Qian, F.-L. Xie and F.K. Soong, "TTS Synthesis with Bidirectional LSTM based Recurrent Neural Networks", in. *Proc. of Interspeech*, pp.1964-1968, 2014.

[11] S. Mike, K. Paliwal. "Bidirectional recurrent neural networks." *IEEE Transactions on Signal Processing*, vol.45, no.11, pp.2673-2681, 1997.

[12] H. Sepp and S. Jürgen, "Long short-term memory," *Neural Computation*, vol.9, no.8, pp. 1735-1780, 1997.

[13] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis", in *Proc. of Eurospeech*, pp.2347-2350, 1999.

[14] S. M. Siniscalchi, D. Yu, L. Deng, and C.-H. Lee, "Exploiting Deep Neural Networks for Detection-Based Speech Recognition," *Neurocomputing* (106), pp. 148-157, 2013.

[15] C.-H. Lee and S. M. Siniscalchi, "An Information-Extraction Approach to Speech Processing: Analysis, Detection, Verification and Recognition," *Proc. IEEE*, Vol. 101, No. 5, pp. 1089-1115, May 2013.

[16] R. Caruana, "Multitask learning," *Machine Learning Journal*, vol. 28, pp. 41–75, 1997.

[17] Z. Wu, C. Valentini-Botinhao, O. Watts and S. King, "Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis", in *Proc. of ICASSP*, pp.4460-4464, 2015.

[18] G. Hinton, S. Osindero, and Y. Teh, "A Fast Learning Algorithm for Deep Belief Nets," *Neural Computation*, vol. 18, pp. 1527-1554, 2006.

[19] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, pp. 1771–1800, 2002.

[20] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. of ICASSP*, pp.1315-1318, 2000.

[21] H. Bourlard and N. Morgan, *Connectionist speech recognition*, Kluwer Academic Publishers, 1994.

[22] D. P. Bertsekas. *Nonlinear Programming. Athena Scientific*, Belmont, MA, 2nd edition, 1999.

[23] D.-E. Rumelhart, G.-E Hinton, and R.-J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[24] F.-K., Soong, and B.-H., Juang, "Line spectrum pair (UP) and speech data compression," in *Proc. of ICASSP*, San Diego, Vol. 1, pp. 1.10.1-1.10.4, May 1984.

[25] H., Kawahara, I., Masuda-Katsuse, A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, 27(5), 187–207, 1999.

[26] 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," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 2011.

[27] Y. Ephraim and D. Malah, "Speech Enhancement using a Minimum Mean-Square Error Log-Spectral Amplitude Estimator," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSp-33, No. 2, pp. 443-445, 1985.

[28] L. Blin, O. Boeffard and V. Barreaud, "WEB-based listening test system for speech synthesis and speech conversion evaluation," in *Proc. of LREC* (Marrakech (Morocco)), 2008.