INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Noise and Metadata Sensitive Bottleneck Features for Improving Speaker Recognition with Non-native Speech Input

Yao Qian, Jidong Tao, David Suendermann-Oeft, Keelan Evanini, Alexei V. Ivanov and Vikram Ramanarayanan

Educational Testing Service Research, USA

{yqian, jtao, suendermann-oeft, kevanini, aivanou, vramanarayanan}@ets.org

Abstract

Recently, text independent speaker recognition systems with phonetically-aware DNNs, which allow the comparison among different speakers with "soft-aligned" phonetic content, have significantly outperformed standard i-vector based systems [9-12]. However, when applied to speaker recognition on a non-native spontaneous corpus, DNN-based speaker recognition does not show its superior performance due to the relatively lower accuracy of phonetic content recognition. In this paper, noise-aware features and multi-task learning are investigated to improve the alignment of speech feature frames into the sub-phonemic "senone" space and to "distill" the L1 (native language) information of the test takers into bottleneck features (BNFs), which we refer to as metadata sensitive BNFs. Experimental results show that the system with metadata sensitive BNFs can improve speaker recognition performance by a 23.9% relative reduction in equal error rate (EER) compared to the baseline i-vector system. In addition, L1 info is just used to train the BNFs extractor, so it is not necessary to be used as input for BNFs extraction, i-vector extraction and scoring for the enrollment and evaluation sets, which can avoid the use of erroneous L1s claimed by imposters.

Index Terms: speaker recognition, DNN, bottleneck features, i-vector

1. Introduction

Deep learning, which can represent high-level abstractions in data with an architecture of multiple non-linear transformations [1], has had a huge impact on automatic speech recognition (ASR) and deep neural networks (DNN) for acoustic modeling have become the state of the art in ASR systems [2,3]. Motivated by the success of DNNs for acoustic modeling in speech recognition, many DNN based approaches have been tried in recent years in order to improve the performance of speaker recognition, with promising results [4-13]. Boltzmann machines or neural networks have been used to train a back-end classifier instead of the state-of-the-art PLDA model for speaker recognition [4,5]. Deep Belief Networks (DBN) have been trained to extract a pseudo i-vector, compactly representing a speech utterance in an alternative type of *i*-vector, which is a projected, low-dimensional vector based upon factor analysis [6], or to model i-vectors for a multi-session speaker recognition task [7].

Recently, phonetically-aware DNNs, which are typically used to train ASR systems, have also been employed for speaker recognition and have significantly outperformed the standard i-vector based approach [9-12]. The phonetic information at the sub-phonemic senone level is used to guide acoustic modeling by the DNN. The well-trained phonetically-aware DNNs are then employed to extract Baum-Welch statistics; i.e. the DNN, replacing the GMM, is used to compute frame posterior probabilities over each of the classes (senones instead of Gaussian Mixture components), for i-vector based text independent speaker recognition [9,10]. In addition, the output of one of the phonetically-aware DNN's hidden layers, also called bottleneck features, is used as feature vectors, instead of the conventional MFCC feature vectors, to train a universal background model (UBM) and build an i-vector extractor using the standard method [11]. The DNN bottleneck features, which can learn discriminative feature representations from a DNN trained in the sense of cross-entropy, are reported to achieve better performance on both speaker and language recognition tasks than using DNN output posteriors for extracting Baum-Welch statistics [11].

However, to the best of our knowledge, there is little research work to investigate the state-of-the-art i-vector based speaker verification technique, focusing on a large, non-native spontaneous corpus. For test security, especially for test of language proficiency, e.g. English, recordings are all from non-native English speakers. There are many challenges in developing a unified system based on phonetically-aware DNNs for speaker recognition in the context of large-scale language proficiency assessment. For instance, test takers in English proficiency tests can have a wide variety of L1 backgrounds worldwide, which makes it difficult to model all possible non-native accent patterns. In addition, the speech quality from different test centers can vary a lot due to various recording settings and environments.

In this paper, we focus on speaker recognition on a non-native spontaneous speech corpus for verifying a language test taker's claimed identity. We focus on using DNN bottleneck features, which can take advantage of phonetically-aware DNNs for i-vector training. Previous work [9-12] has typically utilized corpora which yield high ASR accuracy to obtain more reliable DNN posteriors. In contrast, the present study is performed on a corpus with relatively lower phonetic recognition performance. Both noise features and multi-task learning are integrated together to improve the frame accuracy of senones and to "distill" the L1 (a test taker's native language) information, leading to more robust i-vector based speaker recognition performance.

2. I-Vector Based System

Based upon factor analysis, an *i*-vector is a compact representation of a speech utterance in a low-dimensional

http://dx.doi.org/10.21437/Interspeech.2016-548

subspace. In an *i*-vector model [13], a given speaker- and channel-dependent supervector M can be modeled as

$$M = m + T\omega \tag{1}$$

where m represents a speaker- and channel-independent supervector, which can be estimated by a UBM; T, a low rank matrix, represents the total variability space; and the components of the vector ω are total factors, i.e., segment-specific standard normal-distributed vectors, also called *i*-vectors, and are obtained using maximum a posterior(MAP) estimation. The matrix T is estimated using the EM algorithm [14].

In state-of-the-art i-vector based speaker recognition systems [13,14], speech utterances are first converted to a sequence of acoustic feature vectors, typically 20 dimensional mel-frequency cepstral coefficients (MFCC) and their dynamic counterparts; after that, speaker- and channel-independent super-vectors, which accumulate the zeroth, first and second order sufficient statistics, are computed by using the posterior probabilities of the classes from a pre-trained GMM-UBM; next, the total variability matrix, T, is used to transform the super-vectors to the low dimensional i-vectors, which contain both speaker and channel variability; then linear discriminant analysis (LDA) is often used to do channel compensation; finally a score between the target and the test speaker (or impostor) is calculated by a scoring function such as probabilistic LDA (PLDA) [15] for further compensation or by simply using the cosine distance.

3. Metadata Sensitive Bottleneck Features

3.1. Bottleneck features (BNFs)

Bottleneck features (BNFs) are generated from a DNN in which one of the hidden layers has a small number of units compared to the other layers. It compresses the classification related information into a low dimensional representation. The activations of a narrow hidden bottleneck (BN) layer are used as feature vectors to train a standard GMM. It has been argued that BN features can improve ASR accuracy but not perform as well as the best DNN based system because the BNFs from the middle layer of the DNN degrade the frame accuracy of the senones. However, an approach using a DNN trained by using a subset of the training set for feature extraction and the resulting features from the whole training set used for a GMM-HMM achieves better performance than a DNN-HMM [16]. In addition, a stacked BN, in which the second level consists of a merger NN fusing the posteriors from the first level, and linear activation function, which performs like an LDA or PCA transformation on the activations of previous layer, outperforms the DNN based approaches [17,18].

The DNN BNFs extracted from the second-last liner layer are used as acoustic features to train a GMM-UBM for speaker recognition. It has been shown that systems with BNFs achieve better performance than those are just using output posteriors of DNNs for extracting Baum-Welch statistics [11]. It is assumed that the loss of information at the BNFs is not much affecting the posterior prediction. The DNN bottleneck features should have the same phonetically-aware benefits as those of DNN posteriors, which allows the comparison among different speakers at the same phonetic content, since the BNFs are already precisely mapped to a senone-dependent space. In addition, BNFs carry more speaker-relevant information than DNN output posteriors, which aim at being speaker-independent. Furthermore, the GMM posteriors estimated from BNFs are supposed to be more general than those of DNNs, which learn senone posteriors directly and produce a sharp posterior distribution.

3.2. Metadata sensitive bottleneck features

When we first employed DNNs in our speaker recognition task with a non-native, spontaneous speech corpus, we found that they were not able to achieve the same high performance as shown in [9-11]. We conjecture that the low frame accuracy of senones was caused by the wide range of L1 accents among the language test takers and in turn the low frame accuracy results poor "soft labeling" of the corresponding phonetic content. Fortunately, DNNs are more flexible and versatile than GMMs in acoustic modeling, e.g. there are no assumptions about the underlying statistical distributions and modality of the input data in DNN, both continuous and binary features can be augmented and modeled together naturally. Deep learning technologies like transfer learning or multi-task learning [19], which can exploit commonalities between the training data of different learning tasks so as to transfer learned knowledge from one to another, can be applied directly to acoustic modeling. It has also been shown that noise-aware or room-aware DNN training, where noise or reverberation information is augmented to the input feature vectors, can reduce word error rate (WER) in noisy or reverberant speech recognition [20,21]. Multi-task learning has also been successfully employed to improve phoneme recognition [22] and multilingual speech recognition [23,24].

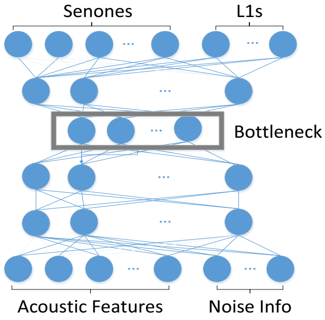

Fig. 1. *Noise-aware DNN with multi-task learning*

Based on those findings, we propose to use metadata to enhance BNF training for non-native speaker recognition. The structure of the DNN used is illustrated in Figure 1, where noise-aware input feature vectors and multi-task learning are employed. If o represents the observed feature vector, which is used as the input vector for DNN training, it is formed as

$$o_t = [x_{t-\tau}, \ldots, x_{t-1}, x_t, x_{t+1}, \ldots, x_{t+\tau}, n_t] \tag{2}$$

where t is the frame index; τ, the number of frames for the sliding window; and n, the noise estimate. We assume that noise is invariant throughout a test taker's whole utterance, i.e., n_t can then be approximated by the average of the beginning and ending frames and fixed over the utterance. In Figure 1, there are two tasks, including: the primary one is senone classification, while the classification of a test taker's native language, L1, is the secondary task. The objective function used in the multi-task learning is

$$\Gamma = \alpha \sum_t \ln p(s_t \mid o_t) + (1-\alpha) \sum_t \ln p(l_t \mid o_t) \tag{3}$$

where s_t and l_t are the senone and L1 labels at the t-th frame, respectively; α is the weight for the task and optimized in terms of recognition accuracy for the validation set. To prevent over-fitting, a regularization term (also called a weight decay term) is added to Equation 3, or the learning can simply be terminated at the point where performance on a held-out validation set starts to deteriorate.

4. Experiments and Results

4.1. Corpora

Our approaches for speaker recognition are evaluated on a corpus of 16 kHz non-native spontaneous speech, which is collected from test takers' responses in an English proficiency assessment.

The training set, which is used to train speaker recognition system's hyper-parameters (GMM-UBM, DNN, i-vector extractor T-matrix, LDA and PLDA projection matrices) consists of 800 hours of speech from 8,700 test-takers. It is drawn from an international assessment of academic English for non-native speakers, which measures the test taker's ability to use and understand English at the university level. Each speaker has 6 utterances, i.e., 45-second spoken responses to express their opinions on a familiar topic or 60-second spoken responses based on reading and listening to relevant prompt materials, roughly 5 minutes per speaker. It contains a total of 140 different L1s. The most frequent L1s are Chinese, Korean, Japanese, Arabic, Spanish, German, Turkish, French, Telugu and Hindi, which account for over 77% of the test taker population (L1 information was not available for 0.1% of the test takers). Figure 2 shows the cumulative distribution of signal-to-noise ratio (SNR) over the data set.

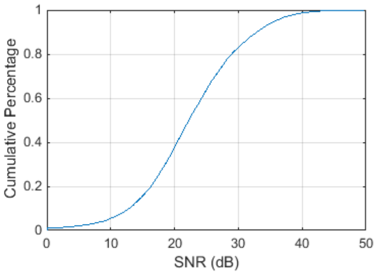

Fig. 2. *Cumulative distribution of SNR in the corpus*

The enrollment and evaluation sets are extracted from the same English proficiency test as that of the training set. The enrollment set contains 6,642 utterances from 1,107 speakers. The evaluation set contains 6,498 utterances from 1,083 speakers, in which 1,000 speakers are repeated test takers, i.e., the same speakers as those in enrollment set but from a different appointment, and 83 speakers are known imposters. The time gap of two appointments for 90% of the repeated test takers is less than half a year.

There are totally 6,001,116 trails, in which 6,828 are targets and the rest is non-target. Target is defined as repeated test takers or manually identified imposters, while non-target is different speakers, i.e., non-repeated test takers, from enrollment and evaluation sets, or imposter for repeated test takers.

4.2. Experimental setup

The speaker recognition systems including the baseline conventional *i*-vector based system, the DNN BNF based system and the metadata sensitive BNF based system, are constructed using the tools from Kaldi [25] and CNTK [26].

4.2.1. Baseline system

The front-end for the baseline speaker recognition system contains 20 dimensional MFCCs including C0, extracted from a 20 ms Hamming window with 10 ms time shift along with their first and second derivatives. Non-speech segments within utterances were deleted through an energy-based voice activity detection (VAD) method. Utterance-based cepstral mean normalization was performed on the acoustic feature vectors. A GMM with 2048 components and a full covariance matrix was trained as the UBM by using the training set mentioned in Section 4.1. The same training set was also used to train a 400-dimensional i-vector extractor T-matrix as well as LDA and PLDA projection matrices.

4.2.2. DNN BNF based system

The training set mentioned in Section 4.1 was first used to train a GMM-HMM and then employed to train a DNN for BNF extraction.

The input feature vectors used to train the GMM-HMM contained 13-dimensional MFCCs and their first and second derivatives. Tri-phone, linear discriminant analysis (LDA), maximum likelihood linear transform (MLLT), and speaker adaptive training (SAT) were used to train the GMM-HMM in the maximum likelihood (ML) sense. In order to get an accurate frame alignment of senones for later DNN training, the parameters of the GMM-HMM were refined by discriminative maximum mutual information (MMI) training.

We used two input feature sets to train DNNs. One is MFCC features with the same dimensions as those used in GMM-HMM. The other one is 40-dimensional mel-scaled log filter-bank features and 3-dimensional pitch related features (FBP). The input features stacked over a 21 frame window (10 frames to either side of the center frame for which predictions are made) are used as the input layer of the DNNs. The output layer of the DNNs has 4,057 nodes, the senones of GMM-HMM obtained by decision-tree clustering. The DNNs have 7 hidden layers; each layer consists of 1024 nodes except the 6th bottleneck layer with 60 nodes. The sigmoid activation function is used for all hidden layers except for the bottleneck layer, which is linear. All of the DNN parameters were firstly initialized using "layer-wise BP" pre-training [27], and then trained by optimizing the cross-entropy function through back-propagation. 90% of the training set mentioned in section 4.1 is employed to train the DNN. The remaining 10% is used as a held-out validation set.

The BNFs extracted from the DNN are used to train the GMM-UBM, i-vector extractor T-matrix, and other hyper-parameters as is done for a standard baseline speaker recognition system, as described in Section 4.2.1.

4.2.3. Metadata sensitive BNF based system

A DNN with the same structure as in Section 4.2.2, except for the input and output layers, is trained by the approach illustrated in Figure 1. The noise vector and the conventional feature vector (FBP is used here) are concatenated together as the input. We use 20 frames at the beginning and the end of an utterance to estimate the noise. Although there are a total of 140 L1s in the corpus, we only use the 28 most common L1s, which makes up over 95% of the test taker population, as the output nodes for L1 classification. The other L1s are relabeled as "UNK" (unknown L1). The training strategies of the DNN are the same as described in Section 4.2.2, while Equation 3 is used as objective function, i.e., a weighted combination of senone classification and L1 classification, instead of just senone classification. The optimal weight α is 0.8 determined on the validation set. The L1s of the training set are used for DNN training and the resulting BNFs are therefore L1-sensitive. In our approach, the L1 info is not necessary for BNF extraction, i-vector extraction, or scoring for the enrollment and evaluation sets, which can avoid the use of erroneous L1s claimed by imposters.

4.2.4. Experimental results and analysis

The frame accuracy is often used to evaluate the performance of a DNN by isolating the issues caused by the vocabulary and the language model in the ASR system. We use it to estimate the accuracy of aligning the frame in the senone space. Figure 3 shows the frame accuracy of epochs (24 hours of data in each) for DNNs with different input and output layers on the validation set. The DNN trained by FBP (filter bank+pitch) achieved higher frame accuracy than that trained by MFCC, i.e. the frame accuracy can be improved from 51.5% to 54.0%. Filter bank features, a relatively rawer feature than MFCCs, are also observed to obtain better recognition performance than MFCCs in DNN-based ASR. As shown in the figure, noise-aware input features and the use of L1 classification as an auxiliary learning task for the DNN training can further improve the frame accuracy from 54.0% to 58.1%.

In order to better understand the contributions of L1 to the frame accuracy, a cheating experiment randomizing L1 labels as DNN output nodes together with senone labels to train a DNN is performed. The corresponding results (FBP+Noise-aware+L1-random) shown in Figure 3 indicate that adding the correct L1 information as targets to train the DNN can improve frame accuracy from 55.3% (FBP+Noise-aware) to 58.1% (FBP+Noise-aware+L1), while the randomized L1 labels as targets decrease the frame accuracy from 55.3% (FBP+Noise-aware) to 54.2% (FBP+Noise-aware+L1-random).

Table 1 shows the performance of the baseline, DNN BNF based and our proposed metadata sensitive BNF based systems. The system performance is evaluated using EER. The baseline system using 60-dimensional MFCCs as input features to train the GMM-UBM and i-vector extractor achieves a 2.51% EER. Neither of the two DNN BNF based systems (BNFs extracted from a DNN trained with MFCCs or FBPs) can outperform the baseline system. We conjecture that the relatively low frame accuracy prevents BNFs from obtaining the phonetically-aware benefits of DNN-based speaker recognition approaches. The system with the metadata-sensitive DNN BNFs, which can improve 4.1% of frame accuracy and also carry noise-aware and discriminative L1 information, achieves a lower EER than the baseline system with a relative 15.1% reduction in EER, i.e., the EER is reduced from 2.58% to 2.19%. The benefit from L1 information to speaker recognition is much larger than that of noise-aware information. The improvement in terms of relative EER reduction is 4.26% and 11.34% by noise-aware and L1 information, respectively.

We also found that BNFs and MFCCs are complementary. The system which uses the combination of 40-dimensional BNFs extracted from the metadata-sensitive DNN and 20-dimensional static MFCCs performs the best among all systems. 40-dimensional BNFs are obtained by retraining the DNN with a 40-node bottleneck layer. This reduces relative EER by 23.9%, compared to the baseline *i*-vector based system, i.e., the EER is reduced from 2.51% to 1.91%.

There are also many features or transforms, e.g., *i*-vector or fMLLR, we can borrow from ASR to improve frame accuracy. However, intuitively they try to remove speaker variability during DNN training or make DNNs more robust to unseen speakers, in which the resulting features seem to be less discriminative for speaker recognition. A detailed survey will be performed in the future.

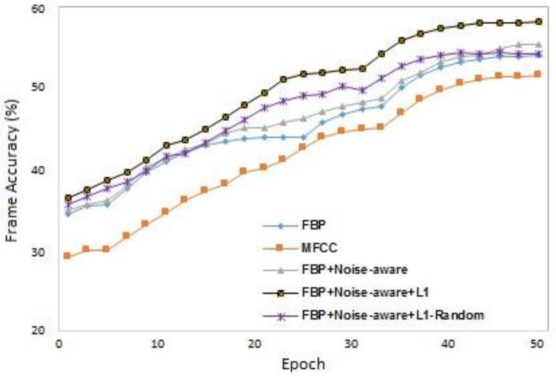

Fig. 3. *Frame accuracy of DNNs with different input and output layers on validation set*

Table. 1 . *EER of different systems*

Systems	EER (%)
Baseline system (MFCC), 60-dim	2.51
BNF (MFCC), 60-dim	2.85
BNF (FBP) , 60-dim	2.58
BNF (FBP+Noise-aware), 60-dim	2.47
BNF (FBP+Noise-aware+L1), 60-dim	2.19
BNF (FBP+Metadata), 40-dim + MFCC, 20 dim	1.91

5. Conclusions

In this paper, we employ metadata (L1 of a test taker) sensitive BNFs to improve the performance of speaker recognition on a corpus of non-native spontaneous speech. The experimental results show that metadata sensitive BNFs are beneficial to speaker recognition. The system with the fusion of BNFs and MFCCs achieves the best performance among all systems. Our future research will explore the effects of additional metadata, e.g. age, and the configuration of DNNs to the performance of BNFs used for speaker recognition.

6. Acknowledgements

The authors would like to thank Frank K. Soong and Wenping Hu from Microsoft Research for valuable discussions while this work was carried out. The authors also would like to thank Yu Zhang from MIT CSAIL for sharing the Kaldi Reader tool, which facilitates the use of a GMM-HMM trained by Kaldi in CNTK.

7. References

[1] Y. Bengio. "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, Vol.2:No.1, pp.1-127, 2009.

[2] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-depedent deep neural networks," in *Proc. of Interspeech*, pp. 437–440, 2011.

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

[4] T. Stafylakis, P. Kenny, M. Senoussaoui and P. Dumouchel, "Preliminary investigation of Boltzmann machine classifiers for speaker recognition," in *Proc. of Odyssey Speaker and Language Recognition Workshop*, 2012.

[5] J. Wang, D. Wang, Z. Zhu, T. F. Zheng and F. K. Soong, "Discriminative scoring for speaker recognition based on I-vectors," in *Proc. of APSIPA*, pp. 1-5, 2014.

[6] V. Vasilakakis, S. Cumani and P. Laface, "Speaker recognition by means of deep belief networks," in *Biometrix Technologies in Forensic Science*, 2013.

[7] O. Ghahabi and J. Hernando, "I-vector modeling with deep belief networks for multi-session speaker recognition," in *Proc. of IEEE Odyssey*, pp. 305–310, 2014.

[8] T. Yamada, L. Wang, and A. Kai, "Improvement of distant-talking speaker identification using bottleneck features of DNN," in *Proc. of Interspeech*, pp.3661–3664, 2013.

[9] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware Deep Neural Networks," in *Proc. of ICASSP 2014*, pp. 1695–1699, 2014.

[10] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep Neural Networks for extracting Baum-Welch statistics for speaker recognition," in *Proc. of Odyssey 2014*, pp. 293–298, 2014.

[11] F. Richardson, D. A. Reynolds and N. Dehak, "A unified Deep Neural Network for speaker and language recognition," in *Proc. of Interspeech 2015*, pp.1146-1150, 2015.

[12] S. Cumani, P. Laface and F. Kulsoom, "Speaker recognition by means of acoustic and phonetically informed GMMs," in *Proc. of Interspeech 2015*, pp.200~204, 2015.

[13] N. Dehak, P. Kenny, R. Dehak, P. Ouellet, and P. Dumouchel, "Front end factor analysis for speaker verification," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[14] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of inter-speaker variability in speaker verification," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 16, no. 5, pp.980 – 988, 2008.

[15] S. Ioffe*, "Probabilistic linear discriminant analysis," in *Proc. of ECCV-2006*, pp.531-542, 2006.

[16] Z. J. Yan, Q. Huo, and J. Xu, "A scalable approach to using DNN-derived features in GMM-HMM based acoustic modeling for LVCSR," in *Proc. of Interspeech*, pp.104-108, 2013.

[17] K. Vesel´y, M. Karafi´at, and F. Gr´ezl, "Convolutive bottleneck network features for LVCSR," in *Proc. of ASRU*, pp. 42–47, 2011.

[18] Y. Zhang, E. Chuangsuwanich, and J. Glass, "Extracting deep neural network bottleneck features using lowrank matrix factorization," in *Proc. of ICASSP*, pp. 185–189, 2014.

[19] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: a review and new perspectives," *IEEE Trans. on Pattern Analysis and Machine Intelligence 35*, pp. 1798–1828, 2013.

[20] M.L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. of ICASSP*, pp. 7398-7402, 2013.

[21] R. Giri, M. L. Seltzer, J. Droppo and D. Yu, "Improving speech recognition in reverberation using a room-aware deep neural network and multi-task learning", in *Proc. of ICASSP*, pp. 5014-5018, 2015.

[22] M.L. Seltzer and J. Droppo, "Multi-task learning in deep neural networks for improved phoneme recognition," in *Proc. of ICASSP*, pp.6965-6969, 2013.

[23] J.T. Huang, J. Li, D. Yu,, Deng, L., Gong, Y., 2013. Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers. *In Proc. of ICASSP*, pp. 7304–7308, 2013.

[24] K. Vesely, M. Karafiat, F. Grezl, M. Janda and E. Egorova, "The language-independent bottleneck features", In *Proc. of Workshop on SLT*, pp. 336-341, 2012.

[25] 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. Vesel, "The Kaldi speech recognition toolkit," in *Proc. of ASRU*, 2011.

[26] D. Yu, A. Eversole, M.L. Seltzer, K. Yao, B. Guenter, O. Kuchaiev, Y. Zhang, F. Seide, G. Chen, H. Wang, J. Droppo, A. Agarwal, C. Basoglu, M. Padmilac, A. Kamenev, V. Ivanov, S.Cyphers, H. Parthasarathi, B. Mitra, Z. Huang, G. Zweig, C. Rossbach, J. Currey, J. Gao, A. May, B. Peng, A. Stolcke, M. Slaney, X. Huang, "An introduction to computational networks and the computational network toolkit", *Microsoft Technical Report MSR-TR-2014-112*, 2014.

[27] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Proc. of ASRU*, 2011.