Attribute Knowledge Integration for Speech Recognition Based on Multi-task Learning Neural Networks

Hao Zheng[1], *Zhanlei Yang*[1], *Liwei Qiao*[2], *Jianping Li*[2], *Wenju Liu*[1]

[1]National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, P.R.China
[2]Electric Power Research Institute of ShanXi Electric Power Company, China State Grid Corp
[1]{`hzheng, zhanlei.yang, lwj`}`@nlpr.ia.ac.cn,`
[2]{`lwqiao.csgd, jpli.csgd`}`@gmail.com`

Abstract

It has been demonstrated that the speech recognition performance can be improved by adding extra articulatory information, and subsequently, how to use such information effectively becomes a challenging problem. In this paper, we propose an attribute-based knowledge integration architecture which is realized by modeling and learning both acoustic and articulatory cues simultaneously in a uniform framework. The framework promotes the performance by providing attribute-based knowledge in both feature and model domains. In model domain, the attribute classification is used as the secondary task to improve the performance of an MTL-DNN used for speech recognition by lifting the discriminative ability on pronunciation. In feature domain, an attribute-based feature is extracted from an MTL-DNN trained with attribute classification as its primary task and phonetic/tri-phone state classification as the secondary task. Experiments on TIMIT and WSJ corpuses show that the proposed framework achieves significant performance improvements compared with the baseline DNN-HMM systems.

Index Terms: multi-task learning, automatic attribute transcription, deep neural networks

1. Introduction

In recent years, the knowledge integration architectures for automatic speech recognition (ASR) have attracted a lot of research attentions [1–5]. These studies aim at knowledge-rich ASR systems and have developed several novel ASR paradigms typically by employing a bottom-up knowledge integration framework to assemble speech cues at various levels [6]. These cues include speech production cues [3], acoustic-phonetic cues [7], and mainly focused attribute cues [8]. The work in [3] provides a comprehensive overview of the usage of speech production knowledge in ASR systems. Some recent works [9–11] explore a bottom-up knowledge integration framework for ASR by employing a two-step approach: firstly detect events or attributes of speech, and then integrating the detected cues into ASR system using evidence mergers [12] or lattice rescore techniques [9]. The knowledge based features, which are known to be linguistically and phonetically relevant, promote the discrimination performance on accent recognition [13] and language identification [14]. Although the effectiveness of using rich knowledge for ASR has been demonstrated under the bottom-up detect-and-merge architecture, how to integrate these cues and knowledge in conventional ASR architecture is still an open issue.

Besides the related detect-and-merge approaches, multi-task learning seems to provide a different framework for knowledge integration. In the conventional deep neural network - hidden Markov model (DNN-HMM) hybrid systems, neural networks are typically trained for one task of senones classification. Recent works try to take advantage of multi-task learning approach to improve the generalization performance of senones classification by jointly learning related tasks, such as grapheme classification [15], phoneme/state context learning [16], and so on. These studies show that when the classifier uses the same architecture of network to perform more than one related task, it learns the shared structure of tasks. In addition, multi-task learning architecture promotes the primary classification accuracy by simultaneously training a related secondary classification task.

In this paper, we describe an attribute-based knowledge integration paradigm which simultaneously models and learns both acoustic and articulatory cues in a uniform framework. Multi-task learning deep neural networks (MTL-DNNs) with shared input and hidden layers and individual output layer for each task are jointly trained on a uniform loss criterion. To integrate the attribute knowledge to speech recognition task, we consider the attribute classification as the secondary task to refine the primary senones classification. Considering that the output of hidden layers in MTL-DNN is a more powerful representation of input features compared with conventional DNN, the output of last hidden layer of MTL-DNN trained with attribute classification as the primary task and phoneme recognition as the secondary task is further concatenated with the original acoustic feature to train the MTL-DNN for phoneme recognition.

The rest of the paper is organized as follows. In Section 2, we review the attributes of speech and describe how to take use of them in our work. Multi-task learning, along with the implementation of attribute classification as the secondary task, is introduced in Section 3. The proposed attribute-feature extractor is illustrated in Section 4. In Section 5, we evaluate the performances of baseline DNNs and our framework on phoneme recognition and large vocabulary continuous speech recognition (LVCSR) tasks. Finally, the conclusion and outlook on future work are given in Section 6.

This research was supported in part by the China National Nature Science Foundation (No.91120303, No.61273267, No.61403370 and No.90820011).

Table 1: *Phonological features (attributes) and their associated phones used in this study.*

	Attribute	Phonemes
manner	Vowel	iy ih eh ey ae aa aw ay ah ao oy ow uh uw er
	Fricative	jh ch s sh z zh f th v dh hh
	Nasal	m n ng
	Stop	b d g p t k
	Approximant	w y l r
place	Coronal	d l n s t z
	High	ch ih iy jhy sh uh uw y ow g k ng
	Dental	dh th
	Glottal	hh
	Labial	b f m p v w
	Low	aa ae aw ay oy
	Mid	ah eh ey ow
	Retroflex	er r
	Velar	g k ng
others	Anterior	b d dh f l m n p s t th v z w
	Back	ay aa ah ao aw ow oy uh uw g k
	Continuant	aa ae ah ao aw ay dh eh er r ey l f ih iy oy ow s sh th uh uw v w y z
	Round	aw ow uw ao uh v y oy r w
	Tense	aa ae ao aw ay ey iy ow oy uw ch s sh f th p t k hh
	Voised	aa ae ah aw ay ao b d dh eh er ey g ih iy jh l m n ng ow oy r uh uw v x w y z
	Silence	sil

2. The attributes of speech

The attributes of speech can be comprehended by a collection of information from fundamental speech sounds [8]. The information of sounds contains speaker characteristics and speaking environment, including linguistic interpretations, a speaker profile encompassing gender, accent, emotional state etc. [8]. The 21 phonological features (attributes), which we used as labels of the secondary task for DNN training and labels of feature extractor to extract attribute-based features, are listed in Table 1 [12]. The usual set of 39 phone classes are then mapped to 21 pairs of 2-dim features, which is 42-dim in total. The value of each pair of the phonological features indicates the presence/absence of this attribute in the pronunciation of this phone, with $(1, 0)$ meaning the presence and $(0, 1)$ meaning the absence.

3. Multi-task learning

Multi-task learning (MTL) [17] is a machine learning technique that improves single-task learning (STL) by training the model with several related tasks using a shared representation. The effectiveness of MTL depends on the relations between each task and the shared learned structure across the tasks [17]. These secondary tasks are used for the training stage and are dropped while testing the unseen data.

3.1. Understanding multi-task learning

One aspect of the effectiveness of secondary task learning, which is similar to dropout strategy and sparse penalty in a sense, can be explained as a regularization to avoid over-fitting. As a result, the MTL is effective especially when the training data is limited, in which case the over-fitting problem is more likely to occur. By adding extra knowledge-based targets, secondary tasks weaken the excessive dependence between the model and the primary task.

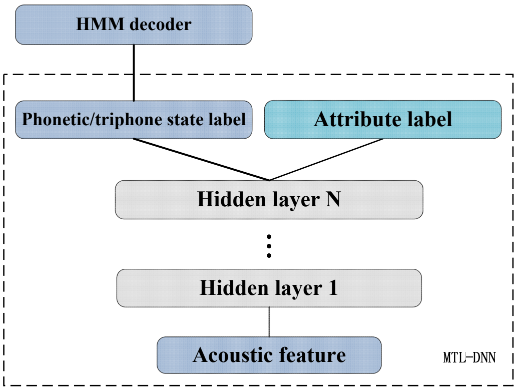

Figure 1: An MTL-DNN system for the joint training of phonetic state and attribute label.

The secondary task learning can also improve the model performance by applying extra information, including accent, speaker, lingual and so on. Taking MTL-DNN for example, the secondary task learning increases the discrimination of the hidden layer outputs on these extra areas, which leads to a more discriminative hidden layer for primary task classification.

3.2. Using attribute classification as the secondary task

In order to improve the performance of phonetic/triphone state classification task, we apply attributes as the secondary task to integrate articulatory knowledge to conventional STL-DNN. As shown in Figure 1, the MTL-DNN uses shared input layer (The proposed framework uses combined feature, which will be described in section 4, as its input layer) and hidden layers but individual output layers for each task. Both of the labels are used to update the weights, including shared weights and individual weights, for the training stage. And at the test stage, only the output layer of phonetic/triphone state classification task is used to calculate the posteriors for the HMM decoder. Given an input vector $\mathbf{x}$, the posterior probability of the ith phonetic state $s_i^{(p)}$ from primary task output layer is computed using the softmax function as follows:

$$P(s_i^{(p)}|\mathbf{x}) = \frac{\exp(y_i^{(p)})}{\sum_{j=1}^{N^{(p)}} \exp(y_j^{(p)})}, \forall i = 1, ..., N^{(p)}, \quad (1)$$

where $y_i^{(p)}$ is the ith output of phonetic/triphone state classification task, and $N^{(p)}$ is the number of phonetic/triphone state, which is 183/3453 in this paper. The posteriors of attribute labels are calculated in pairs. For the ith attribute label, which mapped to the ith pair of outputs, the corresponding posterior is computed using softmax function as follows:

$$P(s_i^{(a)}|\mathbf{x}) = \frac{\exp(y_{pi}^{(a)})}{\exp(y_{pi}^{(a)}) + \exp(y_{ni}^{(a)})}, \quad (2)$$

where $y_{pi}^{(a)}$ and $y_{ni}^{(a)}$ are positive and negative outputs of the ith pair of attribute classification task, separately.

We use cross-entropy as the training criteria. The cross-entropy of phonetic/triphone state classification task is calculated as follows:

$$E^{(p)} = \sum_{\mathbf{x}} [\sum_{i=1}^{N^{(p)}} d_i^{(p)} \log(P(s_i^{(a)}|\mathbf{x}))], \tag{3}$$

where $d_i^{(p)}$ denotes the target values of the ith phonetic/triphone state label, which is 1 when $\mathbf{x}$ belongs to the ith state and is 0 otherwise. The cross-entropy of the ith pair of attribute label is:

$$E_i^{(a)} = \sum_{\mathbf{x}} \{d_i^{(a)} P(s_i^{(a)}|\mathbf{x}) + (1-d_i^{(a)})[1-P(s_i^{(a)}|\mathbf{x})]\}, \tag{4}$$

where $d_i^{(a)}$ denotes whether the ith attribute presents in the pronunciation of the input frame, with $(1, 0)$ denoting the presence and $(0, 1)$ denoting the absence. Then the cross-entropy of attribute classification task is calculated as the summation of all pairs of attribute labels:

$$E^{(a)} = \sum_{i=1}^{N^{(a)}} E_i^{(a)}, \tag{5}$$

where $N^{(a)}$ is the number of attributes, which is 21 in this article. Finally, the MTL-DNN is trained by minimizing the weighted summation of $E^{(p)}$ and $E^{(a)}$:

$$E = (1-\alpha)E^{(p)} + \alpha E^{(a)}, \tag{6}$$

where α is the weight that controls the proportion of gradient which is calculated from the secondary task. We use the factor $(1-\alpha)$ to scale $E^{(p)}$ so that the summation of the two factors is 1, in which case there's no need for a scaler of the learning rate. When α is greater than 0.5, the attribute classification, whose proportion is larger than the phonetic/triphone state classification task, can be seen as the primary task and the phonetic/triphone state classification as the secondary task.

4. Attribute-based features

To further improve the performance of the MTL-DNN, we extract attribute-based features, which can be seen as a knowledge integration feature, to be a part of input features. Considering the powerful representation ability [18] of DNNs, we use the DNN trained with attribute labels to be our feature extractor. Figure 2 illustrates how features are derived. Firstly, we train an MTL-DNN with attribute classification as its primary task. The phonetic/tri-phone state classification task is now used as the secondary task to promote the discriminative capability of the model. This is achieved by setting α in Eq. (6) to a value that is greater than 0.5 (We use $\alpha = 0.8$ in this paper). Feature extraction is achieved via forward-propagation from input layer to the linear outputs of last hidden layer with a back-end Linear Discriminate Analysis (LDA) projection to reduce the feature dimension. Finally we append the the attribute-based feature with the original mel filter-bank (FBANK) feature to obtain the combined feature.

In order to analyze the discrimination of attribute-based features in feature domain, we display the first two principal components of original FBANK features and proposed attribute-based features, as shown in Figure 3. We select 5 phones and show features that belong to their phonetic states (We use 3 states to model each phone/triphone) from TIMIT [19] corpus.

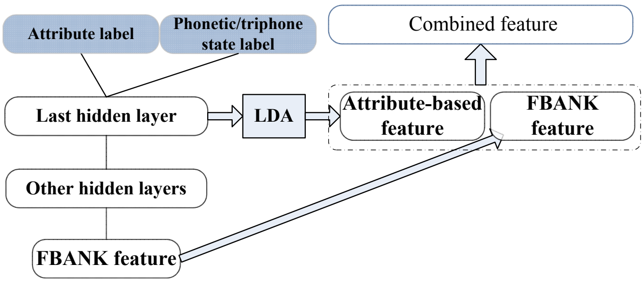

Figure 2: The attribute-based feature extractor.

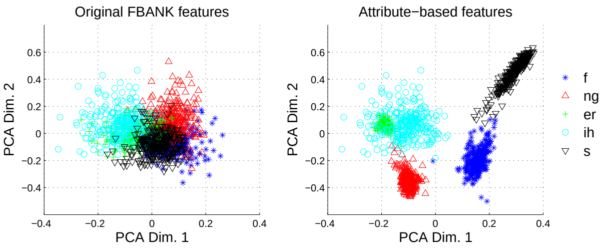

Figure 3: Scatter plot of first two dimensions of PCA projection for features. The plot of left uses original FBANK features and the right one uses proposed attribute-based features.

It can be clearly seen that, compared with the original FBANK features, the attribute-based features have a much stronger discriminative ability on the five states, which leads to a better recognition result as a matter of course.

5. Experiments

5.1. Corpus and configuration

We conduct phoneme recognition on the TIMIT [19] corpus and LVCSR on Wall Street Journal [20] (WSJ) speech corpus to systematically evaluate the performance of the proposed approach and other comparison methods. The training set of TIMIT consists of 3696 utterances spoken by 462 speakers from 8 kind of different dialects. For validation purpose, a dataset of 400 utterances spoken by 50 speakers is chosen to be the development set. We also use the test set with 192 utterances spoken by 24 speakers to examine each of the approaches. The 5000-word speaker independent WSJ0 task [20] is also used to evaluate the performances of proposed approach on LVCSR. The training set used in this article is an 14-hour subset train-si84 (7138 utterances from 83 speakers), rather than the full 81-hour set. We use the dataset dev93 (503 utterances from 10 speakers) as the development set. Evaluation is carried out on the eval92 evaluation data with 330 utterances from 8 speakers. Conventional 40-dimension log mel filter-banks (FBANK), along with their first and second order derivatives, are used as features of each frame. For TIMIT, the features of 11 consecutive frames are combined as input features of the baseline DNN and feature extractor described in section 4, and for WSJ the number of consecutive frames is 15. Each dimension of the input features is normalized to have zero mean and unit variance over the whole training set. Mini-batches of size 256 are used for all batch-based training procedure. For TIMIT a bi-phone language model trained on training utterances is used. For WSJ we use the big dictionary setup in kaldi [21] that adds common pronunciation variants to the default dictionary.

5.2. Greedy layer-wise supervised training

We initialize DNNs with greedy layer-wise supervised training [22]. The DNNs are first initialized randomly with one hidden layer and trained with single-task learning for only one epoch, then we remove the output layer (including the softmax layer) and add a new hidden layer and a new output layer that initialized randomly, and train again. The procedure is repeated several times until the number of hidden layers reaches the target.

5.3. The fine-tune stage

DNNs, including proposed feature extractors and DNNs for phonetic/triphone state classification, are trained with 2048 sigmoid nodes in each layer. For TIMIT, the number of DNN layer is 4 and that for WSJ is 6. The cross-entropy discriminative training is done by back-propagation algorithm to fine-tune the parameters. For TIMIT, each of the 61 phonetic labels is mapped to 3 HMM states, making the dimension of output layer to be 183. For WSJ, the tri-phone states are clustered into 3453 classes by a decision tree. Softmax function is used as the activation of output units. The learning rate is initialized with 0.008 and scaled by a factor of 0.5 if the increase of frame accuracy on validation set is less than 0.5%. The maximum number of training iteration is set to 15. We choose 0.2 as the secondary task weight (α in Eq. (6)) after having tested a list of values. As the momentum is very tricky which may affect the final results significantly [23], and we just intend to make a fair comparison between different approaches in the experiment, the momentum is not used as a part of our experiments.

5.4. Results on TIMIT corpus

The experimental results of phone error rates (PERs) on TIMIT corpus are shown in Figure 4. We can observe that the MTL-DNNs outperform the STL-DNNs in all conditions of input features. Recognition performance on PERs is improved obviously by augmenting all dimensional attribute-based features to STL-DNNs than that trained with original FBANK features. However, some results of MTL-DNN (i.e., results with 20-dimension and 180-dimension extra-added features) become worse after applying attribute-based features. For the MTL-DNN trained with 20-dimension attribute-based feature, it is caused by that the information contained in extra feature is so little that is covered by the secondary task training. For the one trained with 180-dimension attribute-based feature, the dimension of additional feature is so large that the attribute-based feature dilutes the original feature, resulting in an over-fitting. The lowest PER of 21.77%, which is a 4.1% relative reduction to the result 22.66% of the baseline DNN, is achieved by the MTL-DNN trained with combined feature which contains FBANK feature and 120-dim attribute-based feature.

5.5. Results on WSJ corpus

The experimental results on WSJ corpus in terms of word error rates (WERs) are demonstrated in Table 2. Experiments show that both the MTL-DNN with original FBANK feature and the DNN with combined feature achieve significant WER reductions compared with DNN baseline, and the combination attains a further improvement compared with each individual approach. The best result in WERs achieved by the proposed framework has a 14.75% relative reduction to the baseline DNN on test set and a 6.75% relative reduction on development set.

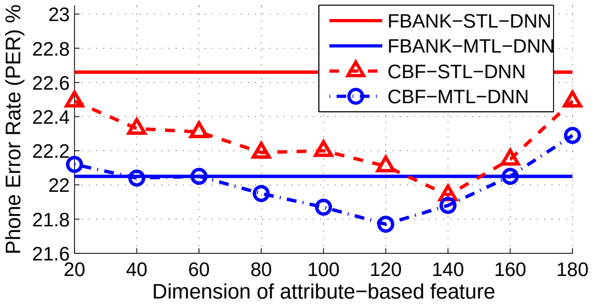

Figure 4: Comparison of different systems using single-task and multi-task learning with FBANK feature and combined feature (CBF). The performance is given in PER (%) for all test conditions.

Table 2: *WERs (%) of WSJ experimental results. The dimension of extra-added features is* 120*.*

	dev	test
STL-DNN	12.30	3.66
STL-DNN + CBF	11.78	3.46
MTL-DNN	11.53	3.22
MTL-DNN + CBF	**11.47**	**3.12**

Although both the attribute-based feature and attribute-based secondary task produce knowledge from attribute labels, the integration framework achieves a better result than each individual approach. The reason is that the attribute-based feature provides discriminative information in feature domain, while the MTL-DNN applies the discriminative rule in model domain. Therefore, as shown in Figure 4, the knowledge integration architecture performs discriminative tuning in both domains and achieves a further improvement in performance.

6. Conclusions

In this paper, we propose an attribute-based knowledge integration architecture which is realized by simultaneously modeling and learning both acoustic and articulatory cues in a uniform framework. In feature domain, an attribute-based feature is extracted by an MTL-DNN trained with attribute classification as the primary task and senones recognition as the secondary task. In model domain, the attribute classification is used as the secondary task to improve the performance by lifting the discrimination ability of an MTL-DNN used for phoneme recognition. We evaluate our framework on TIMIT context-independent phoneme recognition task and WSJ LVCSR task. Experiments on both tasks show that DNNs with modifications on both model and feature domains achieve significant improvements compared with the baseline DNN-HMM system, and the combination of them achieve a further improvement to each of them. As future directions, we plan to investigate our architecture on other tasks to explore the generalization, such as mandarin recognition and accented speech recognition.

7. References

[1] K. Kirchhoff, "Robust speech recognition using articulatory information," 1999.

[2] E. Eide, "Distinctive features for use in an automatic speech recognition system." in *INTERSPEECH*, 2001, pp. 1613–1616.

[3] S. King, J. Frankel, K. Livescu, E. McDermott, K. Richmond, and M. Wester, "Speech production knowledge in automatic speech recognition," *The Journal of the Acoustical Society of America*, vol. 121, no. 2, pp. 723–742, 2007.

[4] C.-H. Lee, "From knowledge-ignorant to knowledge-rich modeling: a new speech research paradigm for next generation automatic speech recognition," in *Proc. ICSLP*, vol. 4, 2004.

[5] C.-Y. Chiang, S. M. Siniscalchi, S.-H. Chen, and C.-H. Lee, "Knowledge integration for improving performance in lvcsr." in *INTERSPEECH*, 2013, pp. 1786–1790.

[6] S. M. Siniscalchi, T. Svendsen, and C.-H. Lee, "Towards bottom-up continuous phone recognition," in *Automatic Speech Recognition & Understanding, 2007. ASRU. IEEE Workshop on*. IEEE, 2007, pp. 566–569.

[7] S. M. Siniscalchi and C.-H. Lee, "A study on integrating acoustic-phonetic information into lattice rescoring for automatic speech recognition," *Speech communication*, vol. 51, no. 11, pp. 1139–1153, 2009.

[8] C.-H. Lee, M. A. Clements, S. Dusan, E. Fosler-Lussier, K. Johnson, B.-H. Juang, and L. R. Rabiner, "An overview on automatic speech attribute transcription (asat)." in *INTERSPEECH*, 2007, pp. 1825–1828.

[9] S. M. Siniscalchi, J. Li, and C.-H. Lee, "A study on lattice rescoring with knowledge scores for automatic speech recognition." in *INTERSPEECH*, 2006.

[10] I. Bromberg, Q. Qian, J. Hou, J. Li, C. Ma, B. Matthews, A. Moreno-Daniel, J. Morris, S. M. Siniscalchi, Y. Tsao *et al.*, "Detection-based asr in the automatic speech attribute transcription project." in *INTERSPEECH*, 2007, pp. 1829–1832.

[11] C.-H. Lee and S. M. Siniscalchi, "An information-extraction approach to speech processing: Analysis, detection, verification, and recognition," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1089–1115, 2013.

[12] D. Yu, S. M. Siniscalchi, L. Deng, and C.-H. Lee, "Boosting attribute and phone estimation accuracies with deep neural networks for detection-based speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4169–4172.

[13] H. Behravan, V. Hautamauki, S. M. Siniscalchi, T. Kinnunen, and C.-H. Lee, "Introducing attribute features to foreign accent recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 5332–5336.

[14] S. M. Siniscalchi, J. Reed, T. Svendsen, and C.-H. Lee, "Exploring universal attribute characterization of spoken languages for spoken language recognition." in *INTERSPEECH*, 2009, pp. 168–171.

[15] D. Chen, B. Mak, and S. Sivadas, "Joint sequence training of phone and grapheme acoustic model based on multi-task learning deep neural networks," in *INTERSPEECH*, 2014.

[16] M. L. Seltzer and J. Droppo, "Multi-task learning in deep neural networks for improved phoneme recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6965–6969.

[17] R. Caruna, "Multitask learning: A knowledge-based source of inductive bias," in *Machine Learning: Proceedings of the Tenth International Conference*, 1993, pp. 41–48.

[18] 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 *INTERSPEECH*, 2013, pp. 104–108.

[19] J. S. Garofolo *et al.*, "Getting started with the darpa timit cd-rom: An acoustic phonetic continuous speech database," *National Institute of Standards and Technology (NIST), Gaithersburgh, MD*, vol. 107, 1988.

[20] D. B. Paul and J. M. Baker, "The design for the wall street journal-based csr corpus," in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 357–362.

[21] 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 *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011, iEEE Catalog No.: CFP11SRW-USB.

[22] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," *Advances in neural information processing systems*, vol. 19, p. 153, 2007.

[23] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, 2013, pp. 1139–1147.