Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 107 (2017) 842 – 847

Procedia Computer Science

International Congress of Information and Communication Technology (ICICT 2017)

Multilingual Convolutional, Long Short-Term Memory, Deep Neural Networks for Low Resource Speech Recognition

Danish bukhari[a], Yutian Wang[a], Hui Wang[a,*]

Key Laboratory of Media Audio & Video (Communication University of China), Ministry of Education

** Corresponding author：hwang@cuc.edu.cn Tel：+8615801362563*

Abstract

Stand-alone and the combined model of Convolutional Neural networks (CNNs) and Long Short-Term Memory (LSTM) and Deep neural Networks (DNNs) have shown great improvements in a variety of Speech Recognition tasks. In this paper we also combined these networks but in this paper we used them for multilingual speech recognition, for the prediction and correction (PAC) architecture, in order to calculate the state probability. Our proposed model is known as PAC-MCLDNN. In this paper, we present experiment results for multilingual training on AP16-OLR task. Furthermore, cross-lingual model transfer and multitask learning for under resourced languages such as Uyghur and Vietnam are also performed which further improved the recognition results.

Keywords: Multilingual CLDNN, LSTM, CNN, Cross-lingual

1. Introduction

Human perception on understanding what other person says depends upon their own assumption at the next utterance and the conformation of the utterance after it has been uttered. This kind of behavior in human speech recognition has been observed in [10] named as prediction, adaptation and correction (PAC).

Deep neural networks (DNN) [5][6][7][8] have overcome the pervious techniques of HMM/GMM [1][2][3][4] in multilingual speech recognition. Recently, Long short-term memory recurrent neural networks (LSTM-RNNs) [10] and Convolutional neural networks (CNNs) [9] have shown quite a lot of improvements on the multilingual speech recognition task. A combined model for all of these three techniques is shown in [11][12]. None of these jointly trained models used the multilingual data for acoustic model training using the PAC architecture. This work is different in a way that a combined model of CNN-RNN-DNN is used for multilingual speech recognition and also favorable for low resource languages.

Peer-review under responsibility of the scientific committee of the 7th International Congress of Information and Communication Technology

doi:10.1016/j.procs.2017.03.179

Prediction and Correction (PAC) previously used the LSTM RNN and DNN technique to predict the posterior probability by using the stack bottleneck (BN) features from the prediction DNN and used it as an input to the correction DNN [10]. In our work the difference is that the input features are multiple languages provided to the prediction frame to the convolutional neural network. The convolutional layers are stacked with 2 fully connected layers which are further stacked with 2 LSTM layers and later on with multilingual deep neural network layers (MDNN). In order to reduce the dimensions of the last fully connected (FC) layer we used a linear layer followed by [11] and add it as an input to the LSTM layer. The prediction information which is the hidden layer to the correction frame was taken from both the MDNN layers and FC CNN layers. Both the observations were observed and reported in section 4. The correction frame only comprises of the 2 FC CNN along with 2 LSTM layers and MDNN layers with softmax layer at the end of the MDNN layer.

Multitasking technique is applied to acoustic modeling at several occasions [13][14]. The related languages are jointly trained in order to improve the recognition of the target language. The target language should be similar to the related languages for the better accuracy of the system. Previously DNN shared layers are used to improve the accuracy of the target language but no one never used the combined model to improve the accuracy of the system. In this paper, in order to reduce the computational load we adopt the multitasking technique from [10] and shared the hidden layers of the model by keeping the output layers distinct.

As in [9], IARPA-Babel corpus is used entirely focusing on the low resource languages. For our case we used AP16-OLR corpus [15] particularly focusing on multilingual speech recognition tasks. We use Uyghur and Vietnam as our target language to improve the accuracy. The reason of choosing Uyghur as a target language is because it is considered very close to Oriental languages. To our best knowledge for the first time Uyghur language is considered for multi-tasking and multilingual speech recognition.

Uyghur is the southeastern Turkic language which is spoken by ten million people in China and the neighboring countries such as Kazakhstan, Kirghizstan [23]. It is influenced primarily by Persian and Arabic and recently by Mandarin Chinese and Russian

The rest of the paper is structured in a way that in section 2 we gives us the combined multilingual model. Section 3 shows our PAC-MCLDNN architecture. Section 4 shows Experimental setup and Section 5 explains our results followed by conclusion and references.

2. Combined Multilingual Model

The network we use here is a combination of Convolutional, long short term memory and deep neural network in the multilingual framework known as MCLDNN. The MCLDNN model is adopted from the single language input featured CLDNN model in [11].

2.1. Deep shared DNN and CNN

Convolutional neural networks after being widely used in computer vision [17][18] made their way towards speech recognition [19][20]. Our model is adopted from the multilingual VBX network defined earlier in [9]. The difference is that in our work two untied FC layers are used combined with the convolutional layer (CV). Frames of input features along with the contextual vectors are applied as an input to the network. Each frame is 40 dimensional log-mel feature and the kernel size is set to 3*3. The stride is set as similar to the pooling size. We use convolutions to reduce the size of the feature maps hence the padding is applied in the highest layers of the network. The weights and biases for all the languages are not the same. They are all concatenated in the fully connected layers. For the MCLDNN model these both FC layers act as the multilingual shared hidden layers. For just the multilingual convolutional network, we untie the FC layers except the last two layers and combine the last two layers with the convolutional layers with max-pooling after every two convolutional layer.

Another difference is that we concatenated the LSTM layers with the FC layers of CNN. As mentioned in earlier work [10] that two layers of LSTM give better performance. We also stick with the same and used two layers of LSTM. The framework of LSTM is followed from [22].

In the end, the output of the LSTM is passed to the MDNN layers. The MDNN from [21] having 1024 hidden units, shows that multilingual training can give an additional gain which tends to be larger when the amount of data is

small. The conv layer in the convolutional network is passed to the linear layer, which reduces its feature dimension with no loss in the accuracy. This is passed on to the MDNN layers. As observed in [21] the higher layers are appropriate to give high order feature representation.

2.2. Multi-scale additions

Our aim is to add more information from all multiple languages and use it for further processing without increasing the computation cost. In order to fulfill this desire we create different strides on the input window with the help of down sampling. This process is only required at the first conv layer. The parameters are small for the rest of the conv layers so this technique is not required at the other steps.

As for the combination of the conv layers with the DNN layers we add a linear layer to reduce the parameters. The addition of linear layer is seen in [11] but in that it concatenates CNN with LSTM but in our case it is used to combine shared CNN with shared DNN layers.

3. PAC-MCLDNN Architecture

As for the overall architecture of our PAC-MCLDNN architecture shown in Figure 1, we adopted the PAC model from [10]. The major difference from [10] is that in our work inside each prediction and correction frame we use MCLDNN.

The prediction MCLDNN hidden layer information to be used as an input for correction MCLDNN are the FC layers. Experiments were carried out by using the MDNN and LSTM layers.

The correction MCLDNN calculates the state posterior probability [10]. The same input features are used for prediction MCLDNN. The FC layer of the correction MCLDNN model depends on the FC layer of the prediction MCLDNN model that creates the recurrent loop. The contextual window size is adopted from [10] and they are also set to 10 for the correction MCLDNN and 1 for the prediction MCLDNN. As in [10], the frame cross-entropy (CE) criterion is used. As proposed in [24] for the prediction MCLDNN we used the phoneme label for prediction targets. For the MCNN model the VBX network in [9] is adopted which outperforms with less amount of data.

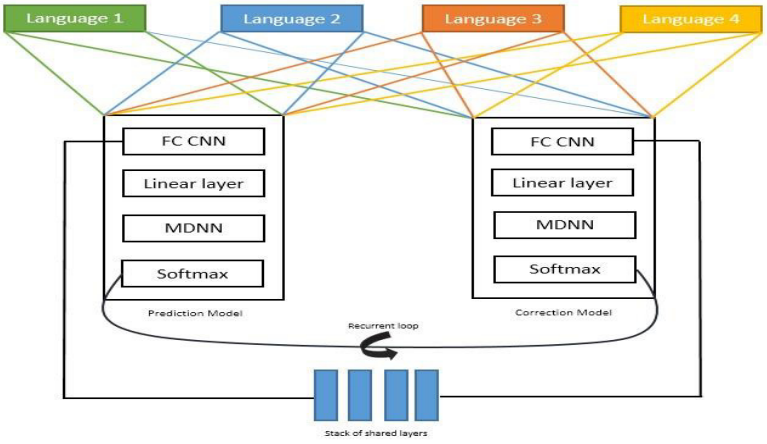

Figure 1: Overview of PAC-MCLDNN Architecture

4. Experimental setup

4.1. Database and Task

AP16-OL7 database comprises of seven different languages from East, Northeast, and Southeast Asia with the main focus on the multilingual Speech Recognition of the Oriental languages [15]. The database is combined with the help of the collaboration between the center of speech and language technologies (CSLT) at the Tsinghua

University and Speechocean. For our work we are going to consider the 10 hours of training set all are reading style, recordings from the mobile phones with sampling rate as 16 kHz and sample size as 16 bits. In addition to that we used THUYG-20 database [16] for Uyghur training. This database consists of 10 hours of training data approx. All are sampled at 16 kHz with sampling size of 16 bits.

For our experiment we used MDNN and MCNN. For the multilingual data we used the universal phone-set and then we performed cross language model transfer by retraining the output layer on target language data.

4.2. Universal phone set:

In order to train the MDNN and MCNN we created a universal phone-set, and we merge all the monolingual phones which share the same symbols. The tied-state targets for the training of the MDNN, we used the Kaldi toolkit. We trained Multilingual HMM/GMM systems and build multilingual decision trees to generate tied-state alignments.

4.3. Cross language model transfer

In order to generate the acoustic model for the new language using MCNN and MDNN, the hidden layers of the MDNN and MCNN are transferred to a new language. The MCNN softmax layer is replaced with the new output layer corresponding to the target language. All the weights which connect the neurons of the FC layer are randomly initialized.

5. Setup

We conducted three different sets of experiments among them first is by analyzing the relation between the source and target languages on MDNN, MCNN and combined MDCNN. The second set of experiments was performed on the concatenation of LSTM/RNN with MDCNN in the prediction and correction architecture i.e. PAC-MCLDNN. Followed by the third set of experiments in which we cross lingual training for Vietnam and multitasking for the Uyghur language. In the first set of experiments we experimented with eight languages. Seven languages were used to train the MDNN and MCNN which is then adapted to the Vietnamese. As for multitasking on Uyghur, the target language is related to the source language and the training is faster as data is comparatively low.

For the second set of experiments speech data remains the same. MDNN hidden layers, MCNN and the last softmax layer are used to form prediction and correction models. Y(t) The output information from the correction model depends on the input information from the prediction model and vice versa so a recurrent loop is generated among them.

6. Results

This section presents the experimental results of our study. We trained standalone models of DNN and CNN with different initializations. MDCNN models and a combined PAC-MCLDNN model are also trained at the same platform with merging the phone-sets which share the same symbols in the IPA table.

The experiments were carried out on seven languages. The DNN model is trained using the first implementation of kaldi. Vietnamese (Vi) data was set up in different amounts. Full set is 8 h, 5h and 1h subsets. The results are summarized in table 1.

Table 1: The word error rate (WER) on the Vietnamese (Vi-Vn) data

Amount of Vi-Vn data	8 hour	5 hour	1 hour
MDNN	12.6	14.5	18.2
MDNN-IPA	12.5	12.8	18.1
MCNN	11.1	10.8	11.2
MCNN-IPA	11.0	10.9	11.4

MCLDNN	10.9	10.7	11.0

The system DNN was pretrained on the (Vi) data. For all the other systems, we used multilingual data for the pretraining. Afterwards, to obtain the VN DNN cross language model transfer is applied. All the DNN configured in this set of experiments had 3 hidden layers, each consisting of 2000 units and were trained from 9 consecutive frames. System DNN corresponds to the baseline system that only uses the Vi data. The VN DNN was trained to give us the estimation of the posterior probability of 2252 tied state triphone targets. We than also evaluated cross language model transfer by bootstrapping the DNN with hidden layers trained on 7 languages using the IPA universal phoneset. MDNN-IPA were trained to estimate the posterior probability of 3,338 and 3,139 tied states targets, respectively extracted from the multilingual decision trees. IPA seems to be favorable choice for us because of the fewer amount of the training data available to us. Crosslingual model transfer consistently improves the WER compared to the baseline system and fine-tuned only with monolingual data. Using IPA to merge the phone-set of the multilingual DNN seems to improve the ASR system.

Table 2: ASR performance for the closest languages (Zh, Id, Ct) performs multitasking on Viatnam (Vi) and Closest language (Zh, Ru, Id) perform multitasking on the target language Uyghur (Uy)

Systems	Vi	Uy
MDNN	63.4	53.1
MDNN-IPA	62.8	53.0
MCNN	61.6	52.8
MCNN-IPA	61.4	52.9
MCLDNN	59.9	50.1
MCLDNN-IPA	59.8	49.1
PAC-MCLDNN	59.6	48.2

The second experiment performed the multitasking technique in which we trained the rich resourced closest languages to train the MCLDNN and PAC-MCLDNN model and then adopt them to the target language. Table 2 summarizes the ASR results. As seen from the table that MCNN performs significantly better than the DNN-IPA. Using a MCLDNN model yields a noticeable improvement over the MCNN network. Similarly, PAC-MCLDNN model can further improve the results.

Acknowledgment

This research is supported by the National Natural Science Foundation of China (Grant No. 61231015，61501410), The State Administration of Press, Publication, Radio, Film and Television of the People's Republic of China scientist research project (Grant No. 2015-53) and the Engineering Planning Project of Communication University of China (Grant No.3132014XNG1425).

References

1. L. Burget, P. Schwarz, M. Agarwal, et al, “Multilingual acoustic modelling for speech recognition based on subspace Gaussian mixture models,” in Proc. ICASSP, pp. 4334-4337, 2010.
2. A. Mohan, S. H. Ghalehjegh, and R. C. Rose, “Dealing with acoustic mismatch for training multilingual subspace Gaussian mixture models for speech recognition,” in Proc. ICASSP, pp. 4893-4896, 2012.
3. L. Lu, A. Ghoshal, and S. Renals, “Regularized subspace Gaussian mixture models for cross-lingual speech recognition,” in Proc. ASRU, pp. 365-370, 2011.
4. L. Lu, A. Ghoshal, and S. Renals, “Maximum a posteriori adaptation of subspace Gaussian mixture models for cross-lingual speech recognition,” in Proc. ICASSP, pp. 4877-4880, 2012.
5. J.T. Huang, J. Li, D. Yu, L. Deng, and Y. Gong, “Cross language knowledge transfer using multilingual deep neural network with shared hidden layers,” in Proc. Of ICASSP, 2013.
6. G. Heigold, V. Vanhoucke, A. Senior, P. Nguyen, M. Ranzato, M. Devin, and J. Dean, “Multilingual acoustic models using distributed deep neural networks,” in Proc. of ICASSP, 2013.

7. A. Ghoshal, P. Swietojanski, and S. Renals, “Multilingual training of deep neural networks,” in Proc. Of ICASSP, 2013.
8. Dongpeng Chen and Brian Kan-Wing Mak "Multitask Learning of Deep Neural Networks for Low-Resource Speech Recognition", ACM Trans ASLP 2015
9. Tom Sercu, Christian Puhrsch, Brian Kingsbury, Yann Lecun, “Very deep multilingual convolutional neural networks for LVCSR,” in Proc. ICASSP 2016.
10. Y. Zhang, D. Yu, M. Seltzer, and J. Droppo, “Speech recognition with prediction-adaptation-correction recurrent neural networks,” in Proc. ICASSP, 2015.
11. T. N Sainath, O. Vinyals, A. Senior, and H. Sak, “Convolutional, long short-term memory, fully connected deep neural networks,” Proc. ICASSP, 2015.
12. L. Deng and J. Platt, “Ensemble Deep Learning for Speech Recognition”, in Proc. Interspeech, 2014.
13. S. Thomas, S. Ganapathy, and H. Hermansky, “Crosslingual and multi-stream posterior features for low resource lvcsr systems,” in Prof. of Interspeech, 2010,pp. 877–880.
14. N.T Vu, F. Metze, and T. Schultz, “Multilingual bottleneck features and its application for under-resourced languages,” Proc. of SLTU, vol. 12, 2012.
15. Dong Wang, Lantian Li, Difei Tang and Qing Chen, “AP16-OL7: A multilingual Database for Oriental Languages and A Language Recognition Baseline,” in Proc. APSIPA 2016.
16. Askar Rozi, Dong Wang, Zhiyong Zhang, “An Open/Free Database and Benchmark for Uyghur Speaker Recognition,” in Proc. O-COCOSDA, 2015.
17. K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” Proc. ICLR, 2015.
18. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Proc. NIPS, 2012, pp. 1097–1105.
19. G. Saon, H.-K. Kuo, S. Rennie, and M. Picheny, “The IBM 2015 english conversational telephone speech recognition system,” Proc. Interspeech, 2015.
20. M. Bi, Y. Qian, and K. Yu, “Very deep convolutional neural networks for LVCSR,” in Proc. Interspeech, 2015.
21. G. Heigold, V. Vanhoucke, A. Senior, J. Dean,“Multilingual acoustic models using distributed deep neural networks,” Proc. ICASSP 2013
22. Hasim Sak, Andrew Senior, Francoise Beaufays, “Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling,” Proc. Interspeech 2014
23. Dimitri Palaz, Ronan Collobert, Mathew Magimai.-Doss, “End-to-end Phoneme Sequence Recognition using Convolutional Neural Networks,” Proc. IJCNN 2013
24. M. D. Zeiler, “ADADELTA: An adaptive learning rate method,” Technical Report, arXiv: 1212.5701, 2012.
25. D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in Proc. International Conference on Learning Representations (ICLR), 2015.
26. F. Guenther and J. Perkell, “A neural model of speech production and its application to studies of the role of auditory feedback in speech,” Speech Motor Control in Normal and Disordered Speech, 2004.
27. J. Huang, J. Li, D. Yu, L. Deng, and Y. Gong, “Cross language knowledge transfer using multilingual deep neural network with shared hidden layers,” in Proc. ICASSP, 2013.
28. A. Mohan and R. Rose, “Multi-lingual speech recognition with low-rank multi-task deep neural networks,” in Proc. ICASSP, 2015.
29. D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motl´ıˇcek, Y. Qian, P. Schwarz, J. Silovsk´y, G. Stemmer, and K. Vesel´y, “The Kaldi speech recognition toolkit,” in Proc. ASRU, 2011.
30. Yu Zhang, Ekapol Chuangsuwanich, James Glass, Dong Yu, “Prediction-Adaptation-Correction Recurrent Neural Networks for Low-Resource Language Speech Recognition,” in Proc. ICASSP, 2016.
31. Dongpeng Chen, Brian Mak, Cheung-Chi Leung, and Sunil Sivadas, “Joint acoustic modeling of triphones and trigraphemes by multi-task learning deep neural networks for low-resource speech recognition,” in International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014.