Joint Bottleneck Feature and Attention Model for Speech Recognition

Long Xingyan
Institution National Digital Switching System Engineering
& Technological Research Centre
Science Avenue 62, Zheng Zhou
China
86-18810683194
lxy120999@qq.com

Qu Dan
Institution National Digital Switching System Engineering
& Technological Research Centre
China
86-13938529655
qudanqudan@sina.com

ABSTRACT

Recently, attention based sequence-to-sequence model become a research hotspot in speech recognition. The attention model has the problem of slow convergence and poor robustness. In this paper, a model that jointed a bottleneck feature extraction network and attention model is proposed. The model is composed of a Deep Belief Network as bottleneck feature extraction network and an attention-based encoder-decoder model. DBN can store the priori information from Hidden Markov Model so that increasing convergence speed of and enhancing both robustness and discrimination of features. Attention model utilizes the temporal information of feature sequence to calculate the posterior probability of phoneme. Then the number of stack recurrent neural network layers in attention model is reduced in order to decrease the calculation of gradient. Experiments in the TIMIT corpus showed that the phoneme error rate is 17.80% in test set, the average training iteration decreased 52%, and the number of training iterations decreased from 139 to 89. The word error rate of WSJ eval92 is 12.9% without any external language model.

CCS Concepts

Computing methodologies → Speech recognition

Keywords

speech recognition; neural network; attention model; bottleneck feature;

1. INTRODUCTION

Acoustic Model (AM) is one of the most important module of speech recognition. Traditional acoustic model is based on Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM) for more than thirty years because HMM can describe the time-varying and non-stationary speech signals, and has a complete theoretical system and efficient algorithm for estimating and decoding model parameters. With the development of deep learning technology and artificial intelligence, scholars have replaced the GMM with the discriminative model represented by the Deep Neural Network[1] (DNN) to calculate the posterior probability of HMM state. The HMM acoustic model has the following defects: it assumes that the probability distribution of the current state is independent to other states except for previous states, so that it can't fully learn and utilize the temporal information of the feature sequence; it decomposes the acoustic model into state recognition and phoneme recognition, resulting in structural complexity of the acoustic model.

Recently sequence-to-sequence model have been used for acoustic model for speech recognition. The sequence-to-sequence model abandoned the assumption based on the HMM and directly transfer the feature sequences into phoneme sequences. There are two main sequence-to-sequence approaches: Connectionist Temporal Classification(CTC)[2][3] and attention model. The key point of CTC is to define a special loss function which can measure the similarity between input and output sequence directly as training[1][5] criterion. But it need to predict targets in every frame and still depends to the assumption that all the frame is conditionally independent to each other.

Another method is attention model[6] or LAS model[7] which directly learns the transformation from feature sequence to phoneme or character sequence[8][9][10]. Attention model is based on encoder-decoder architecture with an additional sub-attention network. Encoder transforms the feature sequence into high-level feature sequence. Attention sub-network can automatically learn the alignment between features and phonemes during training and combine features sequence as a context vector, so that the model does not need any priori alignments[11]. Decoder calculates the posterior probabilities of phones appear on every position of output sequence. The decoder can also calculate the probabilities of character to abandon pronunciation dictionary. However, due to the recursive structure of the recurrent neural network in encoder, the model can't process the multi-frame at the same time, resulting in waste of GPU parallel computing, and therefore causing the long training time problem. Attention model completely abandon the prior knowledge of linguistics and lacks effective initialization parameters, resulting in slow convergence of the model, which is another reason increasing the training time. Some methods[12][13][14] that combine the attention model with CTC are proposed to help train the encoder network and decoding, but it works little to accuracy. In addition, the attention shows poor robustness in noisy environments[12]. [13] utilizes deep convolution network to overcome the problem but it takes much more time and GPUs to converge.

To overcome all above problems, a deep belief network based bottleneck(BN) feature extraction(FA) network[15] is trained and used to append to the front end of attention model. First, a deep belief network is trained based on a tri-phone GMM-HMM model to extract bottleneck features. The key to our approach it that the

ICMAI '18, April 20–22, 2018, Chengdu, China

ACM ISBN 978-1-4503-6420-1/18/04.
https://doi.org/10.1145/3208788.3208798

bottleneck features trained by HMM model are more robust and discriminative and it have been applied in many application of speech recognition[16][17][18][19][20]. What's more, the features provide priori information of HMM model, result in converging rapidly. Then, the number of stack recurrent neural networks layers in encoder network is reduced, which effectively reduces the number of model parameters and training time. Finally, we evaluate our model on TIMIT and WSJ corpus. It shows that our model performs better than the original attention model in both accuracy and training speed.

2. ATTENTION MODEL

The attention-based sequence-to-sequence model was first applied on machine translation[20]. This model implements the directly conversion between sentences in different languages using recurrent neural network. Speech recognition can be viewed as the "translation" of features sequence to phonemes sequence so that the attention model can be utilized to construct acoustic model.

The structure of attention model is shown in Figure 1, which consists of encoder, decoder and attention sub-network. Given a features sequence $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, ..., \boldsymbol{x_T})$, it will generate the output vector sequence $\boldsymbol{y} = (y_1, y_2, ..., y_O)$ whose element record the probabilities of all phonemes.

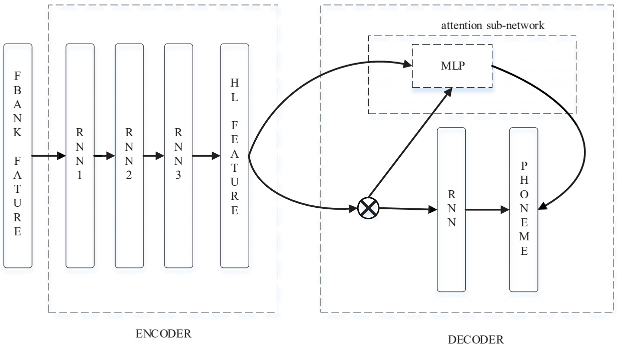

Figure 1. Attention model used for speech recognition

Encoder is to transfer input feature sequence $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, ..., \boldsymbol{x_T})$ to high-lever features sequence $\boldsymbol{h} = (\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_T)$. Therefore, enhance the expressive power and discrimination of feature sequences. Encoder is often constructed by multi-layer recurrent neural network which brings complex gradient computing during training stage.

$$\boldsymbol{h}_t = multilayer - RNN(\boldsymbol{x}_t) \tag{1}$$

The attention sub-network is the core part of attention model. It is a multilayer perceptron with only one hidden layer. When model need to produce vector $\boldsymbol{y}_o$, The sub-network uses previous output vector $\boldsymbol{y}_{o-1}$, together with the high-level vector $\boldsymbol{h}_t$ as the network input to calculate the association mark. Then, the association mark is exponentially normalized value as the weight. Sum the high-level vector according to the weight and finally get the context vector $\boldsymbol{ct}_o$:

$$e_{o,t} = MLP(\boldsymbol{y}_{o-1}, \boldsymbol{h}_t) \tag{2}$$

$$\alpha_{o,t} = \frac{\exp(e_{o,t})}{\sum_{t=1}^{\mathrm{T}} \exp(e_{o,t})} \tag{3}$$

$$\boldsymbol{ct}_o = \sum_{t=1}^{T} \alpha_{o,t} \boldsymbol{h}_t \tag{4}$$

The decoder contains a neural network to calculates the state vector $\boldsymbol{s}_o$ using previous output vector $\boldsymbol{y}_{o-1}$ and context vector $\boldsymbol{ct}_o$. Finally, the output vector $\boldsymbol{y}_o$ is acquire through a softmax layer, each of whose dimension corresponds to the probability that one phoneme will appear at the position o :

$$e_{o,t} = MLP(\boldsymbol{y}_{o-1}, \boldsymbol{h}_t) \tag{5}$$

$$\boldsymbol{y}_o = softmax(\boldsymbol{s}_o) \tag{6}$$

3. METHOD

3.1 DBN based BN FE network

We use a deep belief network as the bottleneck feature extraction network. Figure 2 shows a brief architecture of our bottleneck feature extraction network. The input features to the network are filter-bank(FBANK) feature. The network has 5 or more hidden layers: input layer (123), sigmoid based hidden layers (1024), bottleneck layer (42), other sigmoid based hidden layers (1024) and a softmax based output layer (1236). The number in braces means the dimension of the layers. We use 40 Mel-scale FBANK features together with its energy, then append first and second order difference in every frame, making 123-dimensional vector to input layer. The 1236-dimension output of the network correspond to the tied-state of all tri-phone.

The training can be divided into two stages. The first stage is pretraining in which the parameters of network are initialized by treating every two adjacent layers as an RBM. This training strategy help deep belief network move the weights to good initial values.

The second stage is fine-tuning in which all the weights and bias are trained by a framed based back propagation algorithm with cross-entropy criterion. Training labels are obtained from a trained tri-phone GMM-HMM system.

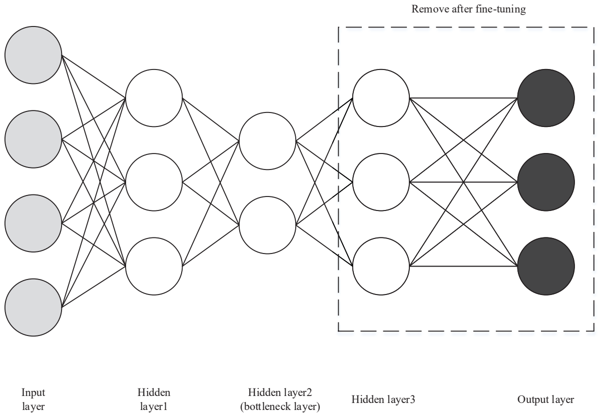

Figure 2. Bottleneck feature extraction network: Train a multi-layer neural network with a relatively small hidden layer (also called bottleneck layer). After finishing training the whole network, layers after bottleneck layers are all removed to construction the bottleneck feature extraction network

3.2 Proposed model: Joint BN FE network and attention-based model

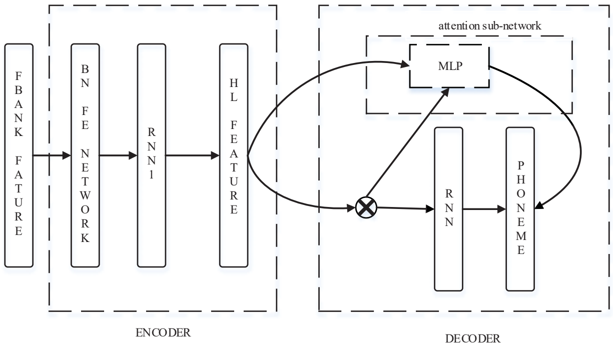

Figure 3. Our proposed model Combined bottleneck feature attention based end-to-end framework: the encoder is added a BNF network to transform the FBANK feature into high-level feature. The decoder included attention sub-network generates the phoneme sequence.

The idea of our model is to use a deep belief network as bottleneck extraction network to extract the high-level features and provide prior knowledge from HMM model for attention model. Figure 3 illustrates the overall architecture of our model, where the encoder network is connected to the BNF extraction network. The BNF extraction network can enhance the distinction of the feature and we therefore expect that it helps attention model acquire faster convergence. Another advantage of using bottleneck feature is that the model gets higher accuracy as a result of the BNF feature is robust to different speaker, accent and noisy. And furthermore, we attempt to reduce the layers of stacks recurrent networks in encoder in order to lower training time and model scale. The encoder of the proposed model is represented as follows:

$$\hat{\boldsymbol{h}}_t = BN - FE - NETWORK(\boldsymbol{x}_t) \tag{7}$$

$$\boldsymbol{h}_t = RNN(\hat{\boldsymbol{h}}_t) \tag{8}$$

4. EXPERIMENT

4.1 Data

We performed experiment on two most commonly used English datasets in speech recognition field: TIMIT and WSJ. TIMIT corpus contain contains 6,300 utterances of spoken English speech from different speakers and accents, from where select 3296 utterances as the training set, 192 sentences as the test set and 400 statements as the development. WSJ is relatively larger corpus and the trainset of it contains 80 hours read sentences collected from Wall Street Journal under clean conditions. We choose eval92 as test set and dev93 as develop set.

To preprocess the data, we choose 16kHZ sampling frequency with 16 sampling bits. Hamming window processing is adopted. The frame length is 25ms, the frame shift is 10ms and the pre-emphasis coefficient is 0.97. The 40-mels FBANK features together with energy are spliced first and second order differences to yield a 123- dimension features. The features in train set are first normalized to standard normal distribution. Mean and variance of trainset are applied to normalize the test set and development set.

4.2 Training

We utilize the kaldi-pdnn toolkit to build and train the bottleneck extraction network and the attention model is implement by Theano.

4.2.1 Bottleneck feature extraction network

The BN FA network contains 7 layers and the dimensions correspond to each layer is ”123-1024-1024-42-1024-1024-1236”. In stage 1, the network is trained layer-by-layer with a contrastive divergence (CD) based on the mini-batch stochastic gradient descent method. The batch size is 128, the number of iterations per layer is 5, and the momentum of each epoch is increased 0.5 to 0.9 by 0.1.

In stage 2, back propagation algorithm based on the small batch stochastic gradient descent algorithm is utilized to perform fine-tuning. The batch size is 256, and the momentum factor is fixed at 0.5. The initial learning rate is 0.08. If the increase of frame rate in development set is less than 0.2% after a training epoch, the learning rate is halved. The training will be stop when the learning rate is lower than 0.02. After finishing training, reserve the parameters from input layer to the bottleneck layer as the BN FA network. Finally, when gets other input features network will calculate the states in bottleneck layer is taken as the 42-dimension bottleneck features. In TIMIT corpus, the dimension of output vector in decoder is 63, corresponding to the posterior probabilities of 61 phonemes, <spc>(space) and <eos> (sequence terminator). In WSJ corpus, the number is 28, corresponding to the posterior probabilities of 26 English characters, <spc> and <eos>.

4.2.2 Attention Model

In attention model, both encoder and decoder are based on Gated Recurrent Unit (GRU) with 256-dimenssion hidden layer. The weight matrix is initialized using a standard orthogonal matrix, the initial value of the bias vector is 0, and the internal state are initialized with an independent standard Gaussian distribution.

Taking the formula (12) as the objective function, the model parameters were trained with stochastic gradient descent using the Adadelta learning rule. The training process was also divided into two stages. In the first stage, the batch size is 8, so that improving the training efficiency and making model converge efficiently; In the second stage, the batch size is 1, and adding random Gaussian noise to all the parameters of the model before calculating the gradient. The purpose is to enhance robustness of the model to noise. However, as the WSJ corpus is larger will spend more time in training, stage 2 won’t be utilized when training WSJ. During the training process, if the minimal phoneme error rate of the development set did not decrease in five consecutive epochs, the training process will automatically enter the next stage or terminate.

4.2.3 Evaluation

In order to evaluate the performance of acoustic model recognition, phone error rate (PER) is taken as the evaluation index in TIMIT corpus. Character error rate(CER) and word error rate(WER) are taken in WSJ corpus. To evaluate and compare the training speed of attention model, we choose the epoch during which all the utterances in train set are used to refresh model parameters.

4.3 Results

4.3.1 *TIMIT corpus*

Table 1 lists the PER and average epoch in stage 2 to attention models using different structured encoder networks. According to the data from the line 1 to the line 3, it's shown that increasing the stack number of recurrent unit can significantly improve the accuracy of model. It means that the encoder extracts higher level feature which is more distinguishable through the deep structure, so the recognition performance is effectively improved. However, the increase in the number encoder layers also leads to a rapid increase in the training epoch.

Line 4 to line 6 in Table 1 shows that the joint bottleneck feature and attention model reduce the PER by 2% to 4% than original attention model. The 1-layer encoder is merely 0.24% higher than 3-layer recurrent network encoder. This proves that the bottleneck features have stronger discrimination and robustness. Therefore the structure based on multilayer recurrent neural network can be replaced. Line 7 in the table gives the performance of DBN-HMM. Its phoneme error rate is obviously higher than most of the attention models in the table, demonstrating that the attention model has a stronger timing modeling capability than the HMM.

Table 1. Performance comparison between acoustic model in Phone Error Rate(PER) and average epoch. Number in Models row refer to the number of layer in encoder

Models	PER	Average Epoch
Attention 1-RNN	21.83	36.2
Attention 2-RNN	21.41	54.8
Attention 3-RNN	19.57	76.3
BN Attention 1-RNN	17.80	37.5
BN Attention 2-RNN	17.75	56.4
BN Attention 3-RNN	17.56	78.1
DBN-HMM	21.6	-

Figure 4 shows the curve of PER of attention model and our bottleneck attention model in development set during training. According to the figure, it is find that the proposed model not only shows a significantly faster decrease of the PER the original attention model, but also reduces the number of required training epoch from 139 to 89. It proves that the HMM based priori information which bottleneck feature extraction network carry can efficiently make attention model converge rapidly.

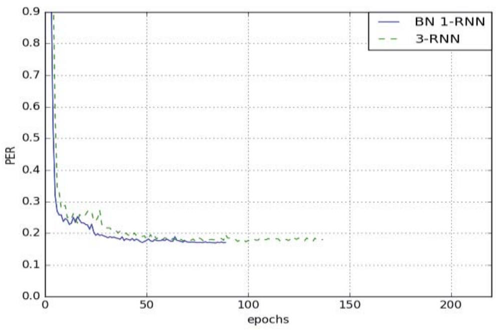

Figure 4. PER of attention model and proposed model in validation set.

4.3.2 *WSJ corpus*

Table 2 shows the results of proposed model in large vocabulary system. Note that we did not utilize any external language model during decoding and both of the encoder are 4-layer recurrent neural network. Our model outperforms than both original attention model and CTC-attention model in WER. The Attention model combined with deep convolution network is better than our model in WER. However, it takes 10 GPU workers and spend O(5) days to converge[12]. Our model uses only 1 GPU worker and converge in 4 days.

Table 2. Word Error Rate (WER) of our model on the Wall Street Journal Corpus eval92 set in comparison with attention model.

Models	WER
attention(Bahdanau et al., 2016)[6]	18.6
CTC-attention(Kim et al. 2017)[13]	18.2
deep CNN attention(Zhang et al., 2017)[12]	10.5
Our Model	12.9

5. CONCLUSION

In this work we showed that a deep belief network based bottleneck feature extraction network can be combined with attention model can to compose a speech recognition system. Through training a deep belief network as bottleneck feature-extraction network and appending it to the encoder of attention model, prior information based on HMM model can be transfer to attention model. As a result, bottleneck extraction network provide more discriminative and robust feature and attention model gets better initialization. That's the reason why reducing the number of recurrent neural network's layer can acquire the comparative word error rate and speed up the training rate at the same time. What's more, the deep belief network is more compatible with GPU parallel computing than recurrent network and it will significantly reduce the time in gradient calculation. The results of experiments in both TIMIT and WSJ corpus show that the proposed model can effectively improve the accuracy and decrease the training time. In future, we will combine our model with efficient compression algorithm[21] to implement an online speech recognition system as well as apply the model in low-resource language the transfer learning.

6. REFERENCES

[1] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A., and Jaitly, N., et al. 2012. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. IEEE Signal Processing Magazine, 29(6), 82-97. DOI= https://doi.org/10.1109/msp.2012.2205597

[2] Miao, Y., Gowayyed, M., and Metze, F. 2015. EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding. 2015 IEEE Workshop on Automatic Speech Recognition and Understanding. ASRU 2015. IEEE, Piscataway, NJ, 2015, 167-174. DOI= https://doi.org/10.1109/asru.2015.7404790

[3] Miao, Y., and Metze, F. 2017. End-to-End Architectures for Speech Recognition. In New Era for Robust Speech Recognition, 299-323. DOI= https://doi.org/10.1007/978-3-319-64680-0_13

[4] Kang, J., Zhang, W. Q., and Liu, J. 2016. Lattice based transcription loss for end-to-end speech recognition. In 2016 10th International Symposium on Chinese Spoken Language Processing. ISCSLP 2016. IEEE, Piscataway, NJ ,1-5. DOI= https://doi.org/10.1109/iscslp.2016.7918455

[5] Kanda, N., Lu, X., and Kawai, H. 2017. Minimum Bayes risk training of CTC acoustic models in maximum a posteriori based decoding framework. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2017. IEEE, Piscataway, NJ, 2017, 4855-4859. DOI= https://doi.org/10.1109/icassp.2017.7953079

[6] Bahdanau, D., Chorowski, J., Serdyuk, D., Brakel, P., and Bengio, Y. 2015. End-to-end attention-based large vocabulary speech recognition. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2015. IEEE, Piscataway, NJ, 2015, 4945-4949. DOI= https://doi.org/10.1109/icassp.2016.7472618

[7] Chan, W., Jaitly, N., Le, Q., and Vinyals, O. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2016. IEEE, Piscataway, NJ, 2016, 4960-4964. DOI= https://doi.org/10.1109/icassp.2016.7472621

[8] Lu, L., Zhang, X., and Renais, S. 2016. On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2017. IEEE, Piscataway, NJ, 2016, 5060-5064. DOI= https://doi.org/10.1109/icassp.2016.7472641

[9] Rosenberg, A., Audhkhasi, K., Sethy, A., Ramabhadran, B., and Picheny, M. 2017. End-to-end speech recognition and keyword search on low-resource languages. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2017. IEEE, Piscataway, NJ, 2017, 5280-5284. DOI= https://doi.org/10.1109/icassp.2017.7953164

[10] Kim, S., Lane, I., Kim, S., and Lane, I. 2017. End-to-End Speech Recognition with Auditory Attention for Multi-Microphone Distance Speech Recognition. INTERSPEECH 2017. ISCA, Grenoble, France, 3867-3871. DOI= https://doi.org/10.21437/interspeech.2017-1536

[11] Hou, J., Zhang, S., and Dai, L. 2017. Gaussian Prediction based Attention for Online End-to-End Speech Recognition. INTERSPEECH 2017. ISCA, Grenoble, France, 3692-3696. DOI= https://doi.org/10.21437/interspeech.2017-751

[12] Zhang, Y., Chan, W., and Jaitly, N. 2017. Very deep convolutional networks for end-to-end speech recognition. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2017. IEEE, Piscataway, NJ, 2016, 4845-4849. DOI= https://doi.org/10.1109/icassp.2017.7953077

[13] Kim, S., Hori, T., and Watanabe, S. 2017. Joint ctc-attention based end-to-end speech recognition using multi-task learning. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2017. IEEE, Piscataway, NJ, 2017, 4835-4839. DOI= https://doi.org/10.1109/icassp.2017.7953075

[14] Hori, T., Watanabe, S., Zhang, Y., and Chan, W. 2017. Advances in Joint CTC-Attention Based End-to-End Speech Recognition with a Deep CNN Encoder and RNN-LM. INTERSPEECH 2017. ISCA, Grenoble, France, 949-953. DOI= https://doi.org/10.21437/interspeech.2017-1296

[15] Mohamed, A. R., Dahl, G. E., and Hinton, G. 2011. Acoustic modeling using deep belief networks. IEEE Transactions on Audio Speech & Language Processing, 20(1), 14-22. DOI= https://doi.org/10.1109/tasl.2011.2109382

[16] Grezl, F., and Fousek, P. 2008. Optimizing bottle-neck features for lvcsr. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2008. IEEE, Piscataway, NJ, 2008, 4729-4732. DOI= https://doi.org/10.1109/icassp.2008.4518713

[17] Sainath, T. N., Kingsbury, B.,and Ramabhadran, B. 2012. Auto-encoder bottleneck features using deep belief networks. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2012. IEEE, Piscataway, NJ, 2012, 4153-4156. DOI= https://doi.org/10.1109/icassp.2012.6288833

[18] Veselý, K., Karafiát, M., and Grézl, F. 2011. Convolutive Bottleneck Network features for LVCSR. Automatic Speech Recognition and Understanding. 42-47. DOI= https://doi.org/10.1109/asru.2011.6163903

[19] Sui, C., Togneri, R., and Bennamoun, M. 2015. Extracting deep bottleneck features for visual speech recognition. In IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2015. IEEE, Piscataway, NJ, 2015, 1518-1522. DOI= https://doi.org/10.1109/icassp.2015.7178224

[20] Hartmann, W., Hsiao, R., Ng, T., Ma, J., Keith, F., and Siu, M. H. 2017. Improved Single System Conversational Telephone Speech Recognition with VGG Bottleneck Features. INTERSPEECH 2017. ISCA, Grenoble, France, 2017112-116. DOI= https://doi.org/10.21437/interspeech.2017-1513

[21] Venkateswaran, P., Sanyal, and A., Das, S., and Nandi, R. 2009. An efficient time domain speech compression algorithm based on lpc and sub-band coding techniques. Journal of Communications, 4(6). DOI= https://doi.org/10.4304/jcm.4.6.423-428