Improve Data Utilization with Two-stage Learning in CNN-LSTM-based Voice Activity Detection

Tianjiao Xu, Hao Li, Hui Zhang* and Xueliang Zhang
Department of Computer Science, Inner Mongolia University, Hohhot, China
E-mail: xtj@mail.imu.edu.cn lihao.0214@163.com alzhu.san@163.com cszxl@imu.edu.cn
Tel/Fax: +86-0471-4993132

***Abstract*—Voice activity detection (VAD) is essential for the speech signal processing system. Convolutional long short-term memory deep neural network (CLDNN), which consists of a CNN and an LSTM, has shown excellent improvement in VAD. However, the training data of the CLDNN must be sequence data because of the LSTM. To improve data utilization, we proposed a two-stage training strategy. Specifically, the first stage trains the CNN on shuffled frame-level data to get high-level feature expression, individually. The second stage trains the LSTM to model the speech continuity. We show that our method has obvious advantages in discriminative ability and generalization ability than compared approaches in different scale of training data, especially in small datasets. The proposed method achieves over 2.89% relative improvement than the original CLDNN on noise matched condition and over 1.07% on unmatched condition.**

I. Introduction

Voice activity detection (VAD) is an important pre-processing step of the speech signal processing system, such as speech enhancement, voice wake-up and speech recognition. The task of VAD is to detect the speech or non-speech events in an audio signal, which is pretty simple for clean speech. However, for noisy speech, especially in low signal-to-noise ratio (SNR) scenario, VAD is a challenge. To cope with the challenge, in recent years, most researches focused on deep learning methods, which take VAD as a binary-class classification problem and train model on pre-marked corpora.

Deep neural networks (DNNs) are commonly used in VAD, e.g. [1, 2]. DNNs are simple and powerful, and are good at solving classification problems. However, DNNs are not good at modeling the sequential information. To provide more context information in the time sequences, *Zhang and Wang* [3] proposed to apply multi-resolution cochleagram feature (MRCG) and boosted deep neural network (bDNN) to explore contextual information. MRCG concatenates multiple cochleagram features calculated at different spectral and temporal resolution. bDNN generates multiple different predictions from a single DNN by boosting contextual information. In [4], *Zhang and Wang* further proposed an ensemble learning framework named multi-resolution stacking for VAD, which is a stack of ensemble classifiers.

In computer vision and other fields, convolutional neural networks (CNNs) have become a popular deep learning model, which is also effective in VAD [5]. CNNs are good at modeling local and shift-invariant patterns. The weights sharing technology makes CNN can build a large model with few trainable parameters. Some effective methods like dilating and gating improve the CNNs' performance further, and make it more suitable for the temporal sequence modeling tasks, e.g. VAD [6, 7]. Another popular deep learning model, recurrent neural networks (RNNs) [8], for example, the long short-term memory (LSTM), is good at modeling the temporal dependence in long sequences, such as the speech signal. However, the LSTM focus more on the underlying differences of each frame, but the feature expression ability is its weakness. Therefore, use the high-level feature instead of the original ones have been proved more efficiently [9], [10]. In [11], it proved that combining DNN, CNN, LSTM can take each one's advantage: CNNs are good at extracting features, LSTMs are good at processing sequence data, and DNNs are good at mapping features into a more separable space. With these observations, *Sainath et al*. [12] proposed a hybrid model called convolutional long short-term memory deep neural networks (CLDNNs), which consist a CNN and an LSTM, for speech recognition tasks. After that, *Zazo et al*. [13] further employed the same model for VAD.

The CLDNN architecture has shown excellent improvement in VAD problem. CNNs are make up for the lack of feature expression of LSTM. In CNN, it is beneficial for the layers to receive various order of data. Shuffle the data provide improvement in robustness ability for CNN and remain the model general and overfit less. *Sainath et al* proposed ShuffleNode [14], which shuffles feature map elements to achieve regularization functions during model training. However, the input of CLDNN must be sequence data because of the LSTM, while CNN located in the bottom layer of CLDNN. So, training the whole networks of CLDNN directly limits the feature expression of CNN.

As a solution, we propose a two-stage training strategy. At the first stage, the CNN is trained on frame-level data to get high-level feature expression, individually. At the second stage, the LSTM receives the high-level feature expression and trained with sequence data. This two-stage training strategy

is similar to greedy layer-wise unsupervised training strategy or layer-by-layer discriminative pretraining in [12, 13]. But differs in that, training CNN on frame-level data benefit a lot in feature expression by shuffle training data, which improve the performance of LSTM, simultaneously.

The rest of this paper constructed as follows. Section 2 reviews the architecture of CLDNN and compared with two-stage learning architecture. Then the experimental details are described in section 3. The results and analysis are presented in section 4. Finally, section 5 concludes the paper.

II. Methods

A. CLDNN

The architecture of CLDNN is shown in Figure 1(a), where X_t represents the feature at frame t. The input is denoted as $[X_1, ..., X_m]$, which means the network processing m frames each time. Thus, the target of this network is the VAD labels of each frame. CLDNN applies CNN at the bottom to reduce frequency variance in the inputs, then passes this to LSTM to perform temporal modeling. Finally, the DNN mapping features into a more separable space. The input is must be organized as a temporal sequence because of the LSTM [12], [13].

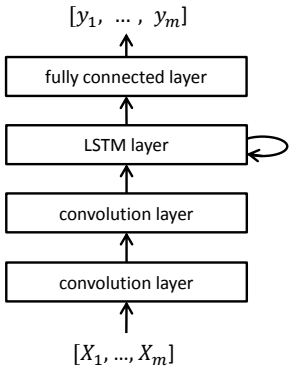
B. Two-stage Learning Architecture

CLDNN architecture combine CNN, LSTM and DNN structure into one unified framework that is trained jointly. To improve data utilization and get a better feature expression of CNN, we propose a two-stage training strategy, which is shown in Figure 1(b). At the first stage, the CNN aims to get a better feature map which trained on frame-level rather than sequence data, which is denoted as X_t and to estimate the corresponding VAD label y_t. So we can feed the frame-level features disorderly. At the second stage, the input is denoted as $[X_1, ..., X_m]$. The LSTM receives the high-level features which produced by the pre-trained CNN. The output of the second stage is $[y_1, ..., y_m]$, which is corresponding with the input. The two-stage learning strategy improves the data utilization, which use the frame-level features at the first stage and sequence data at the second.

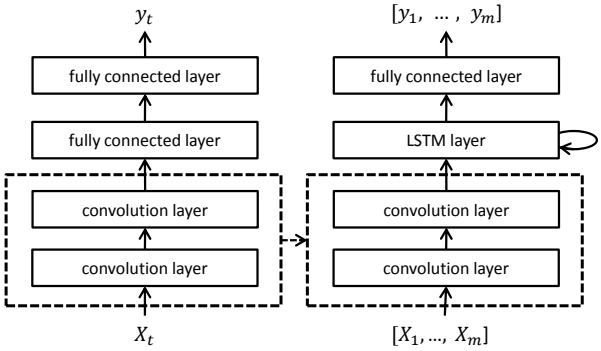
III. Experimental Details

A. Dataset

All experiments are conducted on TIMIT database [17]. We randomly selected 2000 clean utterances from training set, and use the TIMIT core test set as our test utterances. The TIMIT core test set contains 192 utterances, 8 from each of 24 speakers. We concatenate the selected train utterances with some silence segments of random length, which makes the ratio of speech frames account for around 60%. Then mixed with a speech shape noise (SSN) and 4 other types of noise from the NOISEX-92 dataset [18]: babble noise, factory noise, destroy engine noise, and destroyer operations room noise at SNRs of -5 and 0 dB for training. Each noise is divided into two non-overlapping segments for training and testing respectively. To make the sample more generally and multiply, we intercept noise segments from long noise randomly. Besides these four types of noise, another four types of noise are used for noise unmatched test, which includes an unseen factory noise, buccaneer noise from NOISEX-92 and bus noise, street noise from CHiME-4 dataset [19]. The testing SNR are -5, 0 dB and an unmatched 5 dB. All signal is resampled to 16 kHz before mixing. Finally, we conduct experiments on about 30.81 hours of noisy training data.



(a) the architecture of CLDNN



(b) Two-stage leaning strategy

Fig. 1. Network architecture for voice activity detection.

B. Features and Labels

To extract the features, we divided the speech signal into frames using 20 ms hamming window with 10 ms overlap. For all experiments, we use 40-dimensional log-mel filterbank energies as features, which exhibit more temporal and spectral smoothness than MFCC features [20, 21].

The TIMIT corpus includes a time-aligned word transcription file associated with each utterance, in which the word boundaries were aligned with the phonetic segments in time domain. We convert it to the frequency domain labels which can correspond with features for each frame.

TABLE I
AUC(%) COMPARISON BETWEEN THE CONVENTIONAL APPROACHES AND THE PROPOSED TWO-STAGE LEARNING APPROACHES.

SNR	Methods	nosie matched					nosie unmatched			
		babble	engine	factory	op	ssn	bucc	factory2	bus	street
-5dB	SOHN	63.37	58.74	49.94	62.60	45.98	56.41	69.45	73.02	61.68
	CNN	70.52	79.41	74.77	77.99	75.88	68.46	71.22	68.66	76.13
	LSTM	69.37	80.05	79.26	82.30	83.61	82.77	72.04	75.59	77.16
	CLDNN	80.88	89.85	88.08	88.51	89.82	86.13	74.12	75.69	**82.81**
	Proposed	**85.34**	**92.96**	**89.83**	**90.22**	**91.04**	**86.15**	**74.96**	**78.20**	81.88
0dB	SOHN	69.29	68.48	56.35	70.18	49.83	62.76	74.69	78.07	68.34
	CNN	78.26	86.84	81.83	83.90	83.37	75.20	77.52	75.34	82.41
	LSTM	76.37	86.09	85.51	86.16	87.45	87.50	84.39	82.11	85.87
	CLDNN	86.91	93.01	91.97	91.46	92.30	90.56	**87.25**	83.42	90.40
	Proposed	**91.82**	**95.68**	**94.17**	**94.37**	**94.81**	**91.74**	87.08	**84.69**	**91.62**
5dB	SOHN	72.63	74.88	62.77	74.29	57.97	70.98	79.58	81.67	74.86
	CNN	84.03	89.53	86.64	87.59	87.41	78.77	87.40	82.29	86.37
	LSTM	84.83	88.10	87.50	87.97	88.28	88.29	87.97	87.56	87.76
	CLDNN	89.79	93.50	93.53	93.37	93.80	92.57	93.27	92.16	92.84
	Proposed	**94.55**	**96.48**	**95.47**	**95.94**	**95.92**	**95.23**	**95.55**	**94.22**	**95.41**

C. Comparison Methods

We compare the proposed method with four VAD methods: a *statistical method* proposed in [22] by *Sohn et al*, a pure CNN, a pure LSTM, and the original CLDNN methods.

Sohn's method is a state-of-the-art VAD algorithm based on a statistical model in the time-frequency domain for the derivation of the *Likelihood Ratio Test* (LRT).

The CNN has two 1-D convolutional layers with 32 and 64 kernels. The filter size is set to 3. The inputs are frame-level features. Each frame is surrounded by 2 contextual vectors to the left and 2 to the right. For the CNN baseline, a fully connected layer with 64 hidden nodes was connected to the convolutional layer, then a softmax layer is applied to obtain the output.

The LSTM baseline has three LSTM layers, where the first layer contains 128 memory cells, and the second contains 64 memory cells. Finally, the last layer has only one memory cells to get the output. In the training stage, we limit the length of sequences input to 100 frames.

The CLDNN baseline combines the above CNN and a single-layer LSTM which training with sequence data. This configuration makes the parameters number is comparable with the LSTM baseline. From the second convolution layer, we get a 64-dimensional output for each frame. The outputs are feed into an LSTM layer with 64 cells. After that, a fully connected layer with 64 hidden units is used to compress the information to make the feature map is easy to separation. Finally, the last layer output the probability for each frame.

For the two-stage training, the model is identical to the CLDNN. In the first stage, we only train the CNN as we did in the CNN baseline, the inputs of this network is frame-level features rather than a sequence. Then the output layer is removed from the CNN and connected to the LSTM. In the second stage, we only train the LSTM. Its input is the output from the trained CNN in sequences.

D. Optimization and Evaluation Metrics

All models are trained using adaptive moment estimation (Adam) optimizer [23] with a mini-batch size of 256. As a typical classification problem, the loss function is binary cross entropy, which is given by:

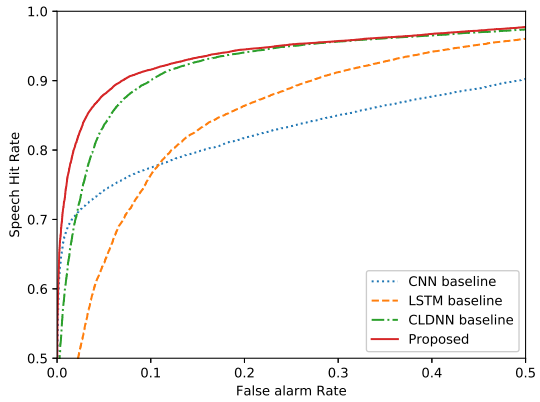
$$L_{vad} = -\sum_{t=1}^{N} \left(Y_t \log \hat{Y}_t + (1 - Y_t) \log(1 - \hat{Y}_t) \right) \quad (1)$$

where N is the number of frame, Y_t and $\hat{Y}_t$ represent the VAD label and the estimated label of the t-th frame, respectively.

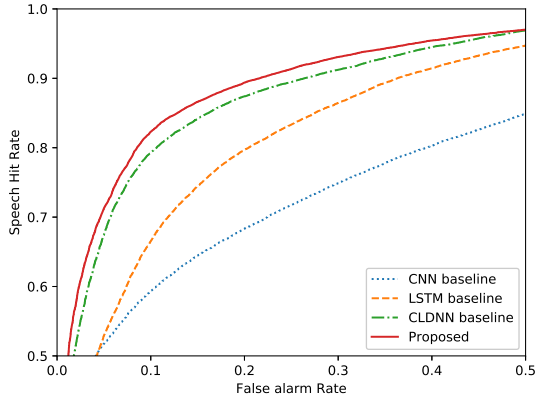
In order to evaluate the performance of the class imbalance problem like VAD, we use the area under the curve (AUC) as the evaluation metrics, which is the area under the receiver operating characteristic (ROC) curve [24]. AUC is considered as an overall metric of the VAD performance rather than the detection accuracy [4]. Higher value means better performance.

IV. RESULTS

First, we conducted all experiments on about 3.07 hours of training set, which is quite small. Table 1 lists the comparison between the four baselines approaches and the proposed method under four seen and four new background noises at various SNRs. The values of each approach indicate the best results use the same testing set under the same conditions. SOHN denotes the method proposed by *Sohn*. CNN, LSTM, and CLDNN denote the other comparison methods.



(a) noise-matched condition



(b) noise-unmatched condition

Fig. 2. ROC curves for the proposed method and baselines.

As can be seen from Table 1, LSTM shows better performance than CNN in almost all matched and unmatched conditions. Combining both advantages of CNN and LSTM, the CLDNN provides over 9.38% and 7.20% relative improvement than CNN and LSTM under the matched condition while 11.99% and 6.87% under unmatched condition, respectively. Comparing with CLDNN baseline, the proposed architecture achieves over 2.89% relative improvement on noise matched condition and over 1.07% on unmatched condition.

Figure 2 shows the ROC curves comparison among each approach under the matched and unmatched test condition, where the SNR is 0 dB. False alarm rate denotes the rate of the non-speech frame which misclassified to speech frame, and the speech hit rate denotes the rate if the speech frame which was classified correctly. This figure shows that the two-stage method achieved a better performance obviously.

A. Comparison among different scale of training data

Table 2 lists the AUC(%) comparison between CLDNN model and two-stage learning strategy using the small and big scale of training set. Under the small training set, the proposed method achieves over 2.89% relative improvement than the original CLDNN on noise matched condition and over 1.07% on unmatched condition.

While under the larger training set, both the CLDNN baseline and the proposed method get improvement on the matched condition. When the training data increases by more than 10 times, CLDNN is comparable in performance to the proposed method in matched condition.

On the unmatched condition, the performance of both approaches implies that it's still a challenge for the model to deal with variable environment, especially the noise is unseen in the training set.

TABLE II
AUC(%) COMPARISON BETWEEN CLDNN AND THE TWO-STAGE LEARNING APPROACHES UNDER DIFFERENT SCALE OF TRAINING DATA

Scale	Model	Matched	Unmatched
3.07 hours	CLDNN	90.45	87.05
	Proposed	**93.07**	**87.98**
30.81 hours	CLDNN	93.17	**87.99**
	Proposed	**94.04**	**87.99**

V. Conclusions

To cope with the challenge of VAD task in low SNR, we propose a two-stage learning strategy to the CLDNN to improve the data utilization. At the first stage, the CNN is trained on frame-level data to get high-level feature expression, individually. At the second stage, the LSTM receives the high-level feature expression and trained with sequence data. Comparing with conventional methods, the bottom layers can be trained better and obtain a more robust feature expression. And the next layers can obtain a better estimation so that the VAD can be more accurate. We compared the proposed method with the conventional CLDNN under various noisy conditions. Experimental results show that the proposed method has obvious advantages in discriminative ability and generalization ability. Using the proposed method, we can obtain an accurate VAD system trained with very limited training data.

VI. Acknowledgments

This research was supported in part by the China National Nature Science Foundation (No. 61876214, No. 61866030).

References

[1] X. L. Zhang and J. Wu, "Deep belief networks based voice activity detection," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 4, pp. 697–710, April 2013.

[2] L. Mateju, P. Cerva, and J. Zdansky, "Study on the use of deep neural networks for speech activity detection in broadcast recordings," *International Conference on Signal Processing and Multimedia Applications*, pp. 45–51, 2016.

[3] X. L. Zhang and D. L. Wang, "Boosted deep neural networks and multi-resolution cochleagram features for voice activity detection," *Proceedings of Interspeech*, pp. 1534–1538, 2014.

[4] X. L. Zhang and D. L. Wang, "Boosting contextual information for deep neural network based voice activity detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 2, pp. 252–264, Feb 2016.

[5] S. Thomas, S. Ganapathy, G. Saon, and H. Soltau, "Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 2519–2523.

[6] J. Kim, H. Choi, J.Park, J. Kim, and M. Hahn, "Voice activity detection based on multi-dilated convolutional neural network," in *Proceedings of the 2018 2Nd International Conference on Mechatronics Systems and Control Engineering*, ser. ICMSCE 2018. New York, NY, USA: ACM, 2018, pp. 98–102. [Online]. Available:

[7] S. Chang, B. Li, G. Simko, T. N. Sainath, A. Tripathi, A. van den Oord, and O. Vinyals, "Temporal modeling using dilated convolution and gating for voice-activity-detection," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018, pp. 5549–5553.

[8] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, "Real-life voice activity detection with lstm recurrent neural networks and an application to hollywood movies," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, May 2013, pp. 483–487.

[9] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 3156–3164.

[10] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," *Computer Science*, pp. 338–342, 2014.

[11] L. Deng and J.C.Platt, "Ensemble deep learning for speech recognition," *Proceedings of Interspeech*, pp. 1915–1919, 2014.

[12] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2015, pp. 4580–4584.

[13] R. Zazo, T. N. Sainath, G. Simko, and C. Parada, "Feature learning with raw-waveform cldnns for voice activity detection," in *Interspeech*, 2016, pp. 3668–3672.

[14] Y. Chen, H. Wang, and Y. Long, "Regularization of convolutional neural networks using shufflenode," in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, July 2017, pp. 355–360.

[15] F. Seide, L. Gang, C.Xie, and Y. Dong, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Automatic Speech Recognition and Understanding*, 2011.

[16] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring strategies for training deep neural networks," *Journal of Machine Learning Research*, vol. 1, no. 10, pp. 1–40, 2009.

[17] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "Timit acoustic-phonetic continuous speech corpus," in *Linguistic Data Consortium*, 1993.

[18] A. Varga, Steeneken, and J. Herman, "Assessment for automatic speech recognition ii: Noisex-92: a database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Communication*, vol. 12, no. 3, pp. 247–251, 1993.

[19] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Computer Speech and Language*, vol. 46, pp. 535 – 557, 2017.

[20] A. Mohamed, "Deep neural network acoustic models for asr," *Doctoral*, 2014.

[21] M. Valenti, A. Diment, G.Parascandolo, S.Squartini, and T. Virtanen, "Dcase 2016 acoustic scene classification using convolutional neural networks," *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE), Budapest, Hungary*, 2016.

[22] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Processing Letters*, vol. 6, no. 1, pp. 1–3, Jan 1999.

[23] D. P. Kingma and J.Ba, "Adam: A method for stochastic optimization," *Computer Science*, 2014.

[24] J.A. Hanley, and B.J. Mcneil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve,"*Radiology*, vol. 143, pp. 29 – 36, 1982.