A COMPLETE END-TO-END SPEAKER VERIFICATION SYSTEM USING DEEP NEURAL NETWORKS: FROM RAW SIGNALS TO VERIFICATION RESULT

Jee-Weon Jung, Hee-Soo Heo, IL-Ho Yang, Hye-Jin Shim, and Ha-Jin Yu[†]

School of Computer Science, University of Seoul, South Korea

ABSTRACT

End-to-end systems using deep neural networks have been widely studied in the field of speaker verification. Raw audio signal processing has also been widely studied in the fields of automatic music tagging and speech recognition. However, as far as we know, end-to-end systems using raw audio signals have not been explored in speaker verification. In this paper, a complete end-to-end speaker verification system is proposed, which inputs raw audio signals and outputs the verification results. A pre-processing layer and the embedded speaker feature extraction models were mainly investigated. The proposed pre-emphasis layer was combined with a strided convolution layer for pre-processing at the first two hidden layers. In addition, speaker feature extraction models using convolutional layer and long short-term memory are proposed to be embedded in the proposed end-to-end system.

***Index Terms*—** speaker verification, end-to-end system, raw audio signal

1. INTRODUCTION

Conventional speaker verification systems are normally composed of the following four stages: pre-processing, acoustic feature extraction, speaker feature extraction, and binary classification. With recent advances in deep neural networks (DNNs), many researchers used DNN to replace individual processes of speaker verification [1, 2]. d-vector, and b-vector schemes were proposed to cover speaker feature extraction and binary classification, respectively [3, 4]. End-to-end systems were also proposed which cover from acoustic feature extraction to binary classification [5, 6, 7, 8].

However, although raw audio signals have been studied in other tasks, such as automatic music tagging [9] and speech recognition [10], a successful speaker verification system using raw audio signal itself has not yet been proposed. Unlike other domains, such as image and text, raw audio signals are difficult to use because they have highly fluctuating values, ranging from -32,768 to 32,767 in widely used 16 bit audio samples.

In this paper, an end-to-end speaker verification system that directly uses raw audio signals is proposed. The proposed system includes specially designed convolutional hidden layers that embed pre-processing into the DNN. An utterance-level speaker feature extraction model that conducts acoustic and speaker feature extractions is also proposed and embedded in the proposed end-to-end system. The proposed pre-processing layer, speaker feature extraction layers, and b-vector system together compose the proposed end-to-end system.

The remainder of this paper is organized as follows. Section 2 presents prior works related to our study. In Section 3, the embedding of raw audio signal processing is addressed, and Section 4 describes two DNN-based models for speaker feature extraction. The proposed complete end-to-end systems are presented in Section 5, and Section 6 delivers the experimental settings and the result analysis. Lastly, the paper is concluded in Section 7.

2. RELATED WORKS

The current study is influenced by two main fields of studies: end-to-end systems in speaker verification and raw audio signal processing in speech recognition and music auto-tagging task. In speaker verification, many studies have been conducted on end-to-end system using acoustic features such as mel-frequency cepstral coefficients, mel-filterbank energies, or spectrograms [8, 6, 11]. Bengio *et al.* and Heo *et al.* [6] used mel-filterbank energies as input features and extracted a speaker feature by using long short-term memory (LSTM) layers. Casper *et al.* [11] exploited spectrograms as input and used recurrent neural networks. For the back-end classifiers, cosine similarity scoring (CSS), and b-vector were embedded into the end-to-end systems. Our prior work concerning the end-to-end system, which is the most relevant to this paper, uses mel-filterbank energies as input, multilayer perceptrion (MLP) as a speaker feature extractor, and b-vector as a classifier [6].

Systems that use raw audio signals exist in other tasks such as music auto-tagging and speech recognition. In music auto-tagging, there was a study on end-to-end systems using

[†] Corresponding author

This work was supported by the Technology Innovation Program (10076583, Development of free-running speech recognition technologies for embedded robot system) funded by the Ministry of Trade, Industry Energy(MOTIE, Korea)

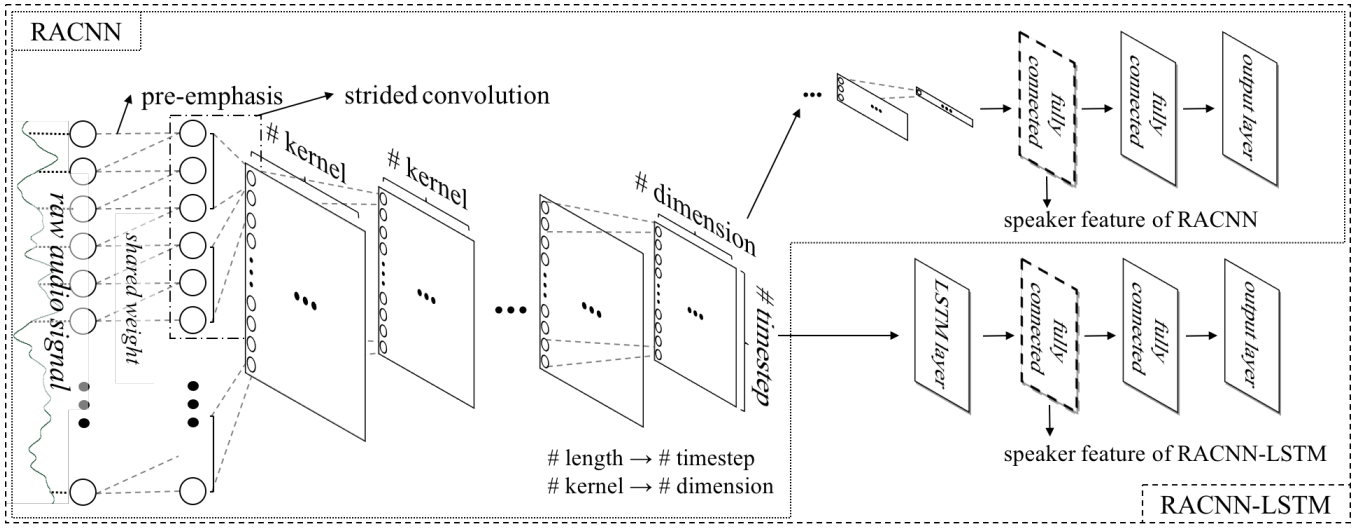

Fig. 1. Illustration of the proposed pre-processing layers and speaker-feature-extraction models.

raw audio signals [9]. In [9], the concept of 'strided convolution' was proposed to process raw audio signals at the first hidden layer, which is also used in this paper. In speech recognition, the work of Palaz *et al.* [10] is one of the most works relevant to ours. It also uses an end-to-end system based on raw audio signals and exploits convolutional layers. The proposed final end-to-end systems of this paper differs with their works [10] mainly in three aspects:

1. Specially designed pre-processing layers improve the performance
2. Convolutional layers and LSTM layers are sequentially used, each with different objectives.
3. B-vectors are composed of the speaker model and test utterance before classification.

3. RAW AUDIO SIGNAL PROCESSING

The processing of raw audio signals by using DNN is difficult mainly because of the fluctuating scales in these signals. In this study, the proposed pre-emphasis layer, depicted in Figure 1, was used to solve the scale problem of raw audio signals.

Pre-emphasis is a widely used pre-processing technique for audio signals [12]. It was originally used to emphasize high frequency signals. In addition, it stabilizes the scale of raw audio signals. Therefore, we embedded pre-emphasis into the proposed systems by designing a pre-emphasis layer. Pre-emphasis is represented as $p(t) = s(t) - \alpha \cdot s(t-1)$, where $s(t)$ refers to the input signal value at time t and α refers to the pre-emphasis coefficient, normally 0.97, and $p(t)$ refers to the pre-emphasized signal.

The pre-emphasis layer is designed using a convolutional layer, and has a kernel of length 2, in which two weights are initialized as -0.97 and 1. This layer is placed at the very first hidden layer and is fine-tuned along with other hidden layers. Note that the pre-emphasis layer should be fine-tuned using a relatively lower learning rate than that used for other layers to prevent its weights from changing too rapidly.

After the pre-emphasis layer, a strided convolution layer was added, as in [9, 10, 13]. The strided convolution layer is also a specially designed hidden layer for processing raw audio signals; here, the stride size is the same as or half of the length of the kernel. Synnaeve *at el.* [13] used this layer with the kernel length set to approximately 25 ms (1 frame) and a stride size of half of the kernel length. Palaz *et al.* [10] shortened the kernel length to 10 ms. Lee *et al.* [9], compared various strided convolutions, and selected the strided convolution with both kernel length and stride size of 3. Similarly, in this study, the strided convolution with both kernel length and stride size of 3 at the second hidden layer was used.

4. SPEAKER FEATURE EXTRACTION

This section presents the two proposed speaker feature extraction models: the raw audio convolutional neural network (RACNN) model and RACNN model with LSTM (RACNN-LSTM). Each model inputs raw audio signals propagated through the pre-processing layers, and is embedded in the final end-to-end system. These two proposed models are depicted in Figure 1, where the small and wide boxes respectively refer to RACNN and RACNN-LSTM.

4.1. Raw audio CNN model

The raw audio CNN model (RACNN) comprises convolutional hidden layers, pooling layers, and fully connected

hidden layers, and simultaneously conducts acoustic and speaker feature extractions. The number of kernels in the convolutional layers increases as the network goes deeper. A max pooling layer is attached after each convolutional layer to reduce the output length of each layer. In this process, speech signals become an utterance-level speaker feature. Compared to the conventional d-vector systems that conduct element-wise averaging on frame-level features to compose an utterance-level feature, the proposed CNN model inherently models utterance-level features without averaging. In the RACNN model, speaker representation features are extracted from the fully connected layers that follow the last convolutional layer.

4.2. Raw audio CNN-LSTM model

The RACNN-LSTM model is an extension of the RACNN model. As described earlier, the RACNN model effectively extracts the utterance-level speaker features from raw audio. Additionally, the output of the pooling layer in the RACNN model can be interpreted as two-dimensional data (i.e. utterance feature map), which the LSTM can utilize as input. LSTM is a special type of recurrent layer, which can model sequential data [14, 15]. To extract speaker features more effectively, the LSTM layer is added next to the selected pooling layer.

The RACNN-LSTM model comprises two parts: utterance-feature map extraction using convolutional layers and LSTM modeling. An utterance feature map refers to the output of the selected pooling layer of the RACNN model, where the length of the input is reduced using pooling layers for LSTM modeling. An utterance feature map has two-dimensional data where the length and number of the kernels are interpreted as timesteps and dimensions, respectively. The utterance feature map is then transformed into a vector through LSTM modeling. Linear activations of the fully connected layer, following the LSTM layer, is used as the speaker feature in the RACNN-LSTM model.

When training the proposed RACNN-LSTM model, the joint-optimization approach in [6] was applied by fine-tuning with the losses from two output layers, one from the RACNN and the other from the RACNN-LSTM. The hidden layers after the selected pooling layer in RACNN are only used for joint training the RACNN-LSTM model and not for extracting the speaker features. These layers are not removed because the joint optimization was determined to enhance the training convergence speed and improve the speaker verification performance, as reported in [6].

5. CLASSIFIER IN THE END-TO-END SYSTEMS

The b-vector classifier [4] was used as the classifier embedded in our final end-to-end system. In the b-vector classifier, operations, such as element-wise summation, subtraction, and multiplication, are applied to the speaker model and test utterance, each in a form of a vector, to compose a b-vector for a given trial. The relationship between the speaker model and test vector is derived through these element-wise operations. The b-vector is then input to a binary classifier MLP to conduct the verification process.

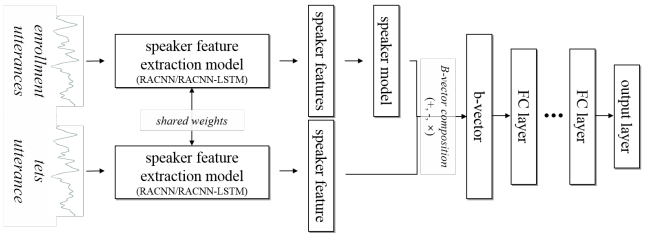

Fig. 2. Illustration of the overall process of the proposed end-to-end systems with embedded pre-processing layers and speaker feature extraction layers.

In the final end-to-end system proposed in this study, the b-vector network was attached to the trained RACNN-LSTM model and the whole network was fine-tuned. The overall proposed end-to-end system is depicted in Figure 2. Identical with the training process of the RACNN-LSTM, the end-to-end system was trained by jointly optimizing the three output layers, one each from RACNN, RACNN-LSTM, and speaker verification result. After the training, convolutional and fully connected layers in the latter part of the RACNN model were removed because these layers are not necessary for speaker verification process. In the evaluation phase, the utterances which compose the speaker model and one test utterance are input to the end-to-end system and the verification result is directly seen at the output layer.

6. EXPERIMENTS AND RESULTS

6.1. Dataset

The evaluation is conducted using part 1 short utterances of RSR 2015 [16]. RSR 2015 comprises 300 speakers, including both genders, and there are 270 utterances of average 3.2 s (9 sessions $\times$ 30 different phrases) for each speaker. Utterances from 100 males and 94 females were used as the development set and the rest was utilized as the evaluation set following [17]. The trial composition in this paper follows RSR 2015's guideline where three utterances from the same speaker with identical phrase is used for composing the speaker model. The trials where the identities of both speaker and phrase are identical was considered as client trial. Cross-gender trials were excluded in examining the performance.

6.2. Experimental configurations

The baseline d-vector system uses mel-filterbank energies as input and is composed of a seven-layer MLP, where each hid-

den layer has 1024 nodes. The length of the input layers of the raw audio systems, i.e., RACNN, RACNN-LSTM, and end-to-end systems was set to 59,049(=3^{10}). All the convolutional layers, except the pre-emphasis and strided convolutional layers, comprise kernels of length 3 and stride 1. Each fully connected layer, except the baseline d-vector system, has 512 nodes. One pre-emphasis layer with one kernel of length 2 and one strided convolutional layer with 128 kernels of length and stride of 3 were placed immediately after the input layer in all the systems, except the baseline d-vector model. After every convolutional layer, a max pooling of 3 was applied to decrease the input length by one-thirds.

The RACNN comprises nine convolutional layers, each with a pooling layer, and two fully connected layers. In the RACNN, the linear activations of the first fully connected layer are used as the speaker features. One LSTM layer and two fully connected layers were attached to the RACNN to compose the RACNN-LSTM model. The LSTM was attached to the output of the 5^{th} pooling layer, where the input length of an utterance is decreased from $59,049$ to $81(=3^4)$. The linear activations of the first fully connected layer were used as speaker features in the RACNN-LSTM. In the end-to-end systems, the b-vectors were calculated by applying element-wise summation, subtraction, and multiplication between a speaker model and a test utterance. The b-vector networks comprise input layer of 1,536 nodes (512×3) and five fully connected layers, each with 1024 nodes, and one output layer with two nodes. Dropout [18], and batch normalization [19] were used in all systems inputting raw audio signals.

6.3. Result analysis

The effectiveness of the proposed pre-emphasis layer was evaluated first, the results of which are shown in Table 1. By applying the pre-emphasis layer, a relatively 24% lower EER was measured. All systems in Table 2 comprise a pre-emphasis layer, except the baseline model that uses mel-filterbank energies as input. The two weights of the pre-emphasis layer initialized to -0,97 and 1 (widely used pre-emphasis coefficients) were fine-tuned to -0.83 and 1.12, respectively. The effectiveness of the strided convolution layer at the second hidden layer has already been experimented in [9]. Therefore, no further evaluation regarding strided convolution layer was made in this study, and the strided convolution layer with kernel length and stride of 3 was used.

Next, the speaker feature extractors were evaluated, and the results are described in Table 2. Our baseline system, with d-vector as the speaker feature extractor and CSS as the back-end classifier, showed EER of 4.89%, which is consistent with the results in [17]. The EER of RACNN with CSS was worse than the baseline. In contrast, the RACNN-LSTM model performed better, implying that relying entirely on convolutional layers to extract speaker features from raw audio signals is less appropriate. This is supported by the fact that the RACNN-LSTM model, which divided the speaker-feature-extraction process into two steps, showed lower EER. The two steps in RACNN-LSTM model refer to the convolutional layers extracting appropriate feature format for the LSTM layer and the LSTM layer modeling sequential data provided by convolutional layers.

In both RACNN and RACNN-LSTM models, CSS was replaced with the b-vector system, and the network was extended to the end-to-end systems. Both systems showed more than the relative 10% improved performance. However, the dynamic changes in the classifier described in [17] were not shown.

Table 1. Comparison of the effectiveness of the proposed pre-emphasis layer.

Model	EER (%)
RACNN (CSS)	7.61
RACNN + pre-emphasis layer (CSS)	**5.22**

Table 2. Performance evaluation of the proposed systems.

Input feature	Model	EER (%)
mel-filterbank energies	d-vector (CSS) (baseline)	4.89
Raw audio signal	RACNN (CSS)	5.22
	RACNN (e2e)	3.94
	RACNN-LSTM (CSS)	3.82
	RACNN-LSTM (e2e)	**3.63**

7. CONCLUSIONS

This paper described a novel complete end-to-end speaker verification system that inputs raw audio signals and outputs verification results. This study was concentrated on the pre-processing of raw audio signals by using embedded layers and the extraction of speaker features from raw audio signals. To facilitate raw audio signal processing by using DNN, specially designed layers were suggested. The proposed speaker feature extraction model is a natural utterance-level system, in which raw audio signals are directly mapped to the utterance-level speaker features by using convolutional and LSTM layers. Therefore, the proposed models facilitated in avoiding the necessity to change frame-level features into utterance-level features. Our future work will include the embedding of various classifiers into the raw audio end-to-end systems.

8. REFERENCES

[1] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1695–1699.

[2] Zeinali H, L. Burget, H. Sameti, O. Glembek, and O. Plchot, "Deep neural networks and hidden markov models in i-vector-based text-dependent speaker verification," in *Odyssey-The Speaker and Language Recognition Workshop*, 2016, pp. 24–30.

[3] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 4052–4056.

[4] H. S. Lee, Y. Tso, Y. F. Chang, H. M. Wang, and S. K. Jeng, "Speaker verification using kernel-based binary classifiers with binary operation derived features," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1660–1664.

[5] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5115–5119.

[6] H. S. Heo, J. W. Jung, I. H. Yang, S. H. Yoon, and H. J. Yu, "Joint training of expanded end-to-end dnn for text-dependent speaker verification," *Proc. Interspeech 2017*, pp. 1532–1536, 2017.

[7] D.Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," *Proc. Interspeech 2017*, pp. 999–1003, 2017.

[8] J. Rohdin, A. Silnova, M. Diez, Q. Plchot, P. Matejka, and L. Burget, "End-to-end dnn based speaker recognition inspired by i-vector and plda," *arXiv preprint arXiv:1710.02369*, 2017.

[9] J. Lee, J. Park, K. Kim, Luke, and J. Nam, "Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms," *arXiv preprint arXiv:1703.01789*, 2017.

[10] D. Palaz, M. Doss, and R. Collobert, "Convolutional neural networks-based continuous speech recognition using raw speech signal," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4295–4299.

[11] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, et al., "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[12] R. Vergin and D. O'Shaughnessy, "Pre-emphasis and speech recognition," in *Electrical and Computer Engineering, 1995. Canadian Conference on*. IEEE, 1995, vol. 2, pp. 1062–1065.

[13] R. Collobert, C. Puhrsch, and G. Synnaeve, "Wav2letter: an end-to-end convnet-based speech recognition system," *arXiv preprint arXiv:1609.03193*, 2016.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[15] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," 1999.

[16] A. Larcher, K. A. Lee, B. Ma, and H. Li, "Text-dependent speaker verification: Classifiers, databases and rsr2015.," *Speech Communication*, vol. 60, pp. 56–77, May 2014.

[17] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang, and K. Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–15, October 2015.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

[19] I. Sergey and S. Christian, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015, pp. 448–456.