RawNet: Advanced end-to-end deep neural network using raw waveforms for text-independent speaker verification

Jee-ween Jung, Hee-Soo Heo, Ju-ho Kim, Hye-jin Shim, and Ha-Jin Yu†

School of Computer Science, University of Seoul, South Korea
jeewon.leo.jung@gmail.com, zhasgone@naver.com, wghll187@naver.com, shinhz6.6@gmail.com, hjyu@uos.ac.kr

Abstract

Recently, direct modeling of raw waveforms using deep neural networks has been widely studied for a number of tasks in audio domains. In speaker verification, however, utilization of raw waveforms is in its preliminary phase, requiring further investigation. In this study, we explore end-to-end deep neural networks that input raw waveforms to improve various aspects: front-end speaker embedding extraction including model architecture, pre-training scheme, additional objective functions, and back-end classification. Adjustment of model architecture using a pre-training scheme can extract speaker embeddings, giving a significant improvement in performance. Additional objective functions simplify the process of extracting speaker embeddings by merging conventional two-phase processes: extracting utterance-level features such as i-vectors or x-vectors and the feature enhancement phase, e.g., linear discriminant analysis. Effective back-end classification models that suit the proposed speaker embedding are also explored. We propose an end-to-end system that comprises two deep neural networks, one front-end for utterance-level speaker embedding extraction and the other for back-end classification. Experiments conducted on the VoxCeleb1 dataset demonstrate that the proposed model achieves state-of-the-art performance among systems without data augmentation. The proposed system is also comparable to the state-of-the-art x-vector system that adopts heavy data augmentation.

Index Terms: raw waveform, deep neural network, end-to-end, speaker embedding

1. Introduction

Direct modeling of raw waveforms using deep neural networks (DNNs) is increasingly prevalent in a number of tasks due to advances in deep learning [1][8]. In speech recognition, studies such as those of Palaz et al., Sainath et al., and Hoshen et al. deal with raw waveforms as input [1][3]. In speaker recognition, studies by Jung et al. and Muckenhirn et al. were the first to comprise systems that input raw waveforms [5][7]. Other domains such as spoofing detection and automatic music tagging are also adopting raw waveform inputs [4][8].

DNNs that directly input raw waveforms have a number of advantages over conventional acoustic feature-based DNNs. First, minimization of pre-processing removes the need for exploration of various hyper-parameters such as the type of acoustic feature to use, window size, shift length, and feature dimension. This is expected to lower entry barriers to conducting studies and lessen the burden of follow-up studies. Additionally, with recent trends of DNN replacing more sub-processes in various tasks, a raw waveform DNN is well positioned to benefit from future advances in deep learning.

Studies across various tasks have shown that an assembly of multiple frequency responses can be extracted when raw waveforms are processed by each kernel of convolutional layers [6][9]. Spectrograms, compared to raw waveform DNNs, have linearly positioned frequency bands, meaning the first convolutional layer sees only adjacent frequency bands (although repetition of convolutions can aggregate various frequency responses at deeper layers). In other words, spectrogram-based CNN can see fixed frequency regions depending on the internal pooling rule. This difference is hypothesized to increase the potential of the directly modeling raw waveforms; as increasing amounts of data become available, this data-driven approach can extract an aggregation of informative frequency responses appropriate to the target task.

In this study, we improve various aspects of the raw waveform DNN proposed by Jung et al., which was the first end-to-end model in speaker verification using raw waveforms [5][7]. This model extracts frame-level embeddings using residual blocks with convolutional neural networks (CNN) [10][11], and then aggregates features into utterance level using long short-term memory (LSTM) [12][13]. Key improvements made by our study include the following:

1. Model architecture: adjustments to various network configurations
2. CNN pre-training scheme: removal of inefficient aspects of the multi-step training scheme in [7][14]
3. Objective function: additional objective functions to incorporate speaker embedding extraction and feature enhancement phase
4. Back-end classification models: comparison of various DNN-based back-end classifiers and proposal of a simple, effective back-end DNN classifier

Through changing these aspects, performance is significantly enhanced. The equal error rate (EER) of utterance-level speaker embedding DNN with cosine similarity on the VoxCeleb1 dataset is 4.8 %, showing 44.8 % relative error rate reduction (RER) compared to the baseline [7]. An EER of 4.0 % was achieved for the end-to-end model using two DNNs, showing an RER of 46.0 %.

The rest of this paper is organized as follows. Section 2 describes the front-end speaker embedding extraction model. Section 3 addresses various back-end classification models. Experiments and results are in Sections 4 and 5 and the paper is concluded in Section 6.

† Corresponding author
2. Front-end: RawNet

We propose a model (referred to as “RawNet” for convenience) that is an improvement of the CNN-LSTM model in [5, 7] by changing architectural details (Section 2.1.), proposing a new pre-training scheme (Section 2.2.), and incorporating a speaker embedding enhancement phase (Section 2.3.).

2.1. Model architecture

The DNN used in this study comprises residual blocks, a gated recurrent unit (GRU) layer [15, 16], a fully-connected layer (used for extraction of speaker embedding), and an output layer. In this architecture, input features are first processed using the residual blocks [10] to extract frame-level embeddings. The residual blocks comprise convolutional layers with identity mapping [11] to facilitate the training of deep architectures. A GRU is then employed to aggregate the frame-level features into a single utterance-level embedding. Utterance-level embedding is then fed into one fully-connected layer. The output of the fully-connected layer is used as the speaker embedding and is connected to the output layer, where the number of nodes is identical to the number of speakers in the training set. The proposed RawNet architecture is depicted in Table 1. It includes a number of modifications to the CNN-LSTM model in [5, 7] allowing for further improvement. First, activation functions are changed from rectified linear units (ReLU) to leaky ReLU. Second, the LSTM layer is changed to a GRU layer. Third, the number of parameters is significantly decreased, including lower dimensionality of speaker embedding (from 1024 to 128).

2.2. CNN pre-train scheme

Extracting utterance-level speaker embeddings directly from raw waveforms often leads to overfitting toward the training set [7]. In [7], multi-step training proposed in [14] was used to avoid such phenomenon. This training scheme first trains a CNN (for frame-level training), and then expands to a CNN-LSTM (for utterance level). This scheme demonstrates significant improvement compared to training a CNN-LSTM with random initialization.

However, the multi-step training approach in [7] is inefficient because after training 9 residual blocks, 3 residual blocks which contains a number of layers are removed when expanding the trained CNN model to a CNN-LSTM model. In our study, a new approach of interpreting the CNN global average pooling layer. After training the CNN, only the global average pooling layer is removed. The objective is to consider the number of convolutional blocks appropriate for training with the recurrent layer and not remove any parameters. This modification enables more efficient and faster training. Application of model architecture modifications detailed in Section 2.1, and the CNN pre-training scheme exhibited an RER of 26.4 % (see Table 2).

2.3. Additional objective functions for speaker embedding enhancement

For speaker verification, a number of studies enhance extracted utterance-level features through an additional process before back-end classification. Linear discriminant analysis (LDA) in i-vector/PLDA systems is one example [17, 18]. Well-known methods such as LDA or recent DNN-based deep embedding enhancement, including discriminative auto-encoder (DCAE), have been applied for this purpose in feature enhancement. In such approaches, one of the main objectives is to minimize intra-class covariance and maximize inter-class covariance of utterance-level features. In this study, we aim to incorporate two phases of speaker embedding extraction and feature enhancement into a single phase, using two additional objective functions.

To consider both inter-class and intra-class covariance, we utilize center loss [19] and speaker basis loss [20] in addition to categorical cross-entropy loss for DNN training. We adopt center loss [19] to minimize intra-class covariance while the embedding in the last hidden layer remains discriminative. To achieve this goal, center loss function was proposed as

$$L_C = \frac{1}{2} \sum_{i=1}^{N} \|x_i - c_{y_i}\|^2,$$

where $x_i$ refers to embedding of the $ith$ utterance, $c_{y_i}$ refers to the center of class $y_i$, and $N$ refers to the size of a mini-batch.

Speaker basis loss [20], aims to further maximize inter-class covariance. This loss function considers a weight vector between the last hidden layer and a node of the softmax output layer as a basis vector for the corresponding speaker and is formulated as:

$$L_{BS} = \sum_{i=1}^{M} \sum_{j=1, j \neq i}^{M} \cos(W_i, W_j),$$

Table 1: RawNet architecture. For convolutional layers, numbers inside parentheses refer to filter length, stride size, and number of filters. For gated recurrent unit (GRU) and fully-connected layers, numbers inside the parentheses indicate the number of nodes. An input sequence of 59,049 is based on the training mini-batch configuration. At the evaluation phase, input sequence length differs. Center loss and between-speaker loss is omitted for simplicity. For residual blocks, layers under the dotted line are conducted after residual connection.

| Layer      | Input: 59,049 samples | Output shape |
|------------|------------------------|--------------|
| Strided -conv | Conv(3,3,128) | (9683, 128) |
|             | LeakyReLU              |              |
| Res block  | Conv(3,1,128) BN      |              |
|            | LeakyReLU              |              |
|            | Conv(3,1,128) ×2      | (2187, 128) |
|            | BN                     |              |
|            | LeakyReLU              |              |
|            | MaxPool(3)             |              |
| GRU        | GRU(1024)              | (1024,)      |
| Speaker embedding | FC(128)   | (128,)      |
| Output     | FC(1211)               | (1211,)     |
of speakers within the training set. Hence, the final objective function used in this study is

\[ \mathcal{L} = \mathcal{L}_{CE} + \lambda \mathcal{L}_{C} + \mathcal{L}_{BS}, \]  

(3)

where \( \mathcal{L}_{CE} \) refers to categorical cross-entropy loss and \( \lambda \) refers to the weight of \( \mathcal{L}_{C} \) (10^{-3} in our study).

3. DNN-based back-end classification

In speaker verification, cosine similarity and PLDA are widely used for back-end classification to determine whether two speaker embeddings belong to the same speaker [18]. Although PLDA has shown competitive results in a number of studies, DNN-based classifiers have also shown potential in previous researches [5]. A number of DNN-based back-end classifiers have been explored: concatenation of speaker embeddings, and b-vector and rb-vector systems [22]. A novel back-end classifier using DNN is introduced based on analysis of a b-vector system.

The b-vector approach uses various element-wise binary operations to derive the relationship between the speaker embeddings. However, because weighted summation operations in a DNN can replace (or even find better combinations of) addition and subtraction, we hypothesized that the core term contributing to the success of the b-vector is the multiplication operation. Therefore, we propose an approach using the concatenation of the speaker embedding, test utterance, and their element-wise multiplication. Experimental results show that by only adding element-wise multiplication, performance exceeds that of the b-vector (see Table 4, ‘concat&mul’).

4. Experimental settings

Experiments in this study were conducted using Keras, a deep learning library in Python with Tensorflow back-end [27–29]. Code used for experiments is available at https://github.com/Jungjee/RawNet.

4.1. Dataset

We use VoxCeleb1 dataset which comprises approximately 330 hours of recordings from 1251 speakers in text-independent scenarios and has a number of comparable recent studies in the literature. All utterances are encoded at a 16 kHz sampling rate with 16-bit resolution. As the dataset comprises various utterances of celebrities from YouTube, it includes diverse background noise and varied durations. We followed the official guidelines which divide the dataset into training and evaluation sets of 1211 and 40 speakers respectively.

4.2. Experimental configurations

We didn’t apply any pre-processing, such as normalization, except pre-emphasis [30] to raw waveforms. For mini-batch construction, utterances were either cropped or duplicated into 59049 samples (\( \approx 3.59s \)) in the training phase, following [5, 7]. In the evaluation phase, no adjustments were made to length; the whole utterance was used.

RawNet comprises onestrided convolutional layer, six residual blocks, one GRU layer, one fully-connected layer, and an output layer (see Table 1). Residual block comprises two convolutional layers, two batch normalization (BN) layers [31], two leaky ReLU layers, and a max pooling layer as shown in Table 1. Residual connection adds the input of each residual block to the output of the second BN layer. A GRU layer with 1024 nodes aggregates frame-level embeddings into an utterance-level embedding. One fully-connected layer is used to extract
In this paper, we propose an end-to-end speaker verification system using two DNNs, for extracting speaker embedding extraction and back-end classification. The proposed system has a simple, yet efficient, process pipeline where speaker embeddings are extracted directly from raw waveforms and verification results. Such a simplified process pipeline is expected to lower barriers to research and provide opportunities for many researchers to apply new techniques.


7. References

[1] 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.

[2] T. N. Sainath, R. J. Weiss, A. Senior, K. W. Wilson, and O. Vinyals, “Learning the speech front-end with raw waveform cnns,” in Sixteenth Annual Conference of the International Speech Communication Association, 2015.

[3] Y. Hoshen, R. J. Weiss, and K. W. Wilson, “Speech acoustic modeling from raw multichannel waveforms,” in Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on. IEEE, 2015, pp. 4624–4628.

[4] H. Dinkel, N. Chen, Y. Qian, and H. Yu, “End-to-end spoofing detection with raw waveform cnns,” in Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. IEEE, 2017, pp. 4860–4864.

[5] J. Jung, H. Heo, I. Yang, H. Shim, and H. Yu, “A complete end-to-end speaker verification system using deep neural networks: From raw signals to verification result,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 5349–5353.

[6] H. Muckenhirn, M. Doss, and S. Marcell, “Towards directly modeling raw speech signal for speaker verification using cnns,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 4884–4888.

[7] J. Jung, H. Heo, I. Yang, H. Shim, and H. Yu, “Avoiding speaker overfitting in end-to-end dnn using raw waveform for text-independent speaker verification,” in Proc. Interspeech 2018, 2018, pp. 3583–3587.

[8] 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.

[9] M. Ravanelli and Y. Bengio, “Speaker recognition from raw waveform with sincnet,” arXiv preprint arXiv:1808.00158, 2018.

[10] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[11] ———, “Identity mappings in deep residual networks,” in European conference on computer vision. Springer, 2016, pp. 630–645.

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

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

[14] 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.” in INTERSPEECH, 2017, pp. 1532–1536.

[15] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” arXiv preprint arXiv:1406.1078, 2014.

[16] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” arXiv preprint arXiv:1412.3555, 2014.

[17] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, “Front-end factor analysis for speech verification,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 19, no. 4, pp. 788–798, 2011.

[18] P. Kenny, “Bayesian speaker verification with heavy-tailed priors,” in Odyssey, 2010, p. 14.

[19] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, “A discriminative feature learning approach for deep face recognition,” in European conference on computer vision. Springer, 2016, pp. 499–515.

[20] H.-s. Heo, J.-w. Jung, I.-h. Yang, S.-h. Yoon, and H.-j. Shim, and H.-j. Yu, “End-to-end losses based on speaker basis vectors and all-speaker hard negative mining for speaker verification,” arXiv preprint arXiv:1902.02455, 2019.

[21] S. Shon, H. Tang, and J. Glass, “Frame-level speaker embeddings for text-independent speaker recognition and analysis of end-to-end model,” arXiv preprint arXiv:1809.04437, 2018.

[22] 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.

[23] H.-s. Heo, I.-h. Yang, M.-j. Kim, S.-h. Yoon, and H.-j. Yu,”Advanced b-vector system-based deep neural network as classifier for speaker verification,” in 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016, pp. 5465–5469.

[24] M. Hajibabaei and D. Dai, “Unified hypersphere embedding for speaker recognition,” arXiv preprint arXiv:1807.08312, 2018.

[25] K. Okabe, T. Koshinaka, and K. Shinoda, “Attentive statistics pooling for deep speaker embedding,” arXiv preprint arXiv:1803.10963, 2018.

[26] A. Nagrani, J. S. Chung, and A. Zisserman, “Voxceleb: A large-scale speaker identification dataset,” in Interspeech, 2017.

[27] F. Chollet et al., “Keras,” https://github.com/keras-team/keras 2015.

[28] A. Martín, A. Ashish, B. Paul, B. Eugene et al., “Tensorflow: Large-scale machine learning on heterogeneous distributed systems,” 2015. [Online]. Available: http://download.tensorflow.org/paper/whitepaper2015.pdf

[29] A. Martin, B. Paul, C. Jianmin, C. ZhiFeng, D. Andy, D. Jeffrey, D. Matthieu, G. Sanjay, I. Geoffrey, I. Michael, K. Manjunath, L. Josh, M. Rajat, M. Sherry, M. G. Derek, S. Benoit, T. Paul, V. Vijay, W. Pete, W. Martin, Y. Yuan, and Z. Xiaoqiang, “Tensorflow: A system for large-scale machine learning,” in 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 2016, pp. 265–283. [Online]. Available: https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf

[30] R. Vergin and D. O’Shaughnessy, “Pre-emphasis and speech recognition,” in Electrical and Computer Engineering, 1995. Canadian Conference on, vol. 2. IEEE, 1995, pp. 1062–1065.

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

[32] S. J. Reddi, S. Kale, and S. Kumar, “On the convergence of adam and beyond,” 2018.

[33] Y. Gal and Z. Ghahramani, “A theoretically grounded application of dropout in recurrent neural networks,” in Advances in neural information processing systems, 2016, pp. 1019–1027.

[34] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, “X-vectors: Robust dnn embeddings for speaker recognition,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 5329–5333.