System Description for BIT-SR System

Xinmei Su[1], *Qingran Zhan*[1], *Chenguang Hu*[1], *Xiang Xie*[1,2]

[1]School of Information and Electronics, Beijing Institute of Technology, Beijing, China
[2]Shenzhen Research Institute, Beijing Institute of Technology, Shenzhen, China
suxinmei2022@126.com

Abstract

Speaker retrieval (SR) is a task to select the enrolled speakers from a large amount of test utterances. Extracting speaker embeddings in retrieval tasks depends on deep neural networks in general. The Emphasized Channel Attention, Propagation and Aggregation in Time Delay Neural Network (ECAPA-TDNN), which is the state-of-the-art (SOTA) neural network in the field of speaker verification (SV), can also be used in solving SR problems. In this paper, we describe the Beijing Institute of Technology-Speaker Retrieval (BIT-SR) system of SR task and propose an extension of architecture based on ECAPA-TDNN that combines multiple embeddings in different layers. First, we replace the front TDNN layers in ECAPA-TDNN with multi-scale convolution layers that are adopted by multi-scale 1-D convolutional kernels. By applying multi-scale convolution, multiple scales of feature maps are extracted and multiple information is learned by the neural network. Second, skip connections in SE-Res2blocks are added to avoid overfitting. Third, a novel pooling method is employed and concatenated with the statistic attentive pooling to achieve better performances. Combination of multiple poolings can help the network get more spatial features. The proposed system obtains a relative improvement of 22.7% comparing with the SOTA model before. A further qualitative analysis shows that our proposed system can better cluster utterances from the same speaker.

1. Data

The datasets to train our proposed system are the CN-Celeb [1, 2] and Aishell [3]. CN-Celeb dataset includes two subsets: CN-Celeb1 and CN-Celeb2, except the CN-Celeb1 evaluation set. It contains speech from Chinese celebrities and covers 11 genres in real condition, including play, movie, interview, etc. The dataset focuses on large-scale and complex scenarios which is challenging for speaker retrieval. Aishell is an open-source Chinese Mandarin speech corpus which is recorded in a pure environment using high fidelity microphone. The total number of speakers is 400. The SR.dev and SR.test set are provided to evaluate our system by the competition organizer.

2. Models

2.1. Overview

The whole architecture of the retrieval system is depicted in Figure 1. First, the frame-level log Mel-filterbank features are fed into multi-scale convolution layers to produce multi-scale convolution features. The multi-scale filter embeddings are then concatenated on the dimension of channel features. Second, the concatenated embeddings go through into several SE-Res2Blocks with skip connections additionally added, and then

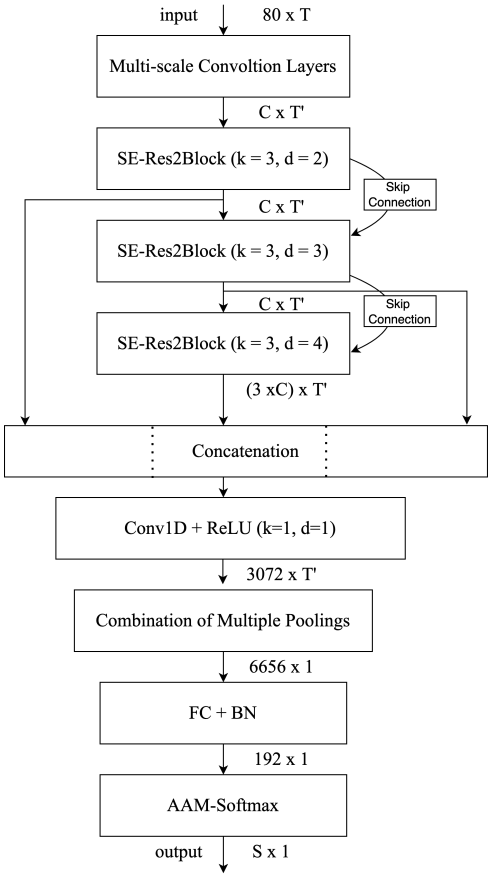

Figure 1: *The whole architecture of our proposed system.*

the statistic-attentive pooling and SPoC pooling are combined. Finally, the features are passed into batch normalization and fully-connected softmax layers. The output probabilities are considered as speaker embeddings. The details are described in the following subsections.

2.2. Multi-scale convolution layers

The standard ECAPA-TDNN utilizes a 1-D CNN with the kernel size of 5 as the TDNN layer. However, single-scale convolution filtering limits the extraction of feature embedding and provide limited spatial features. By going through multi-scale convolution layers which are adopted by multi-scale convolution kernels, both the short term and high level features are preserved. This multi-scale filtering method is usually done on raw

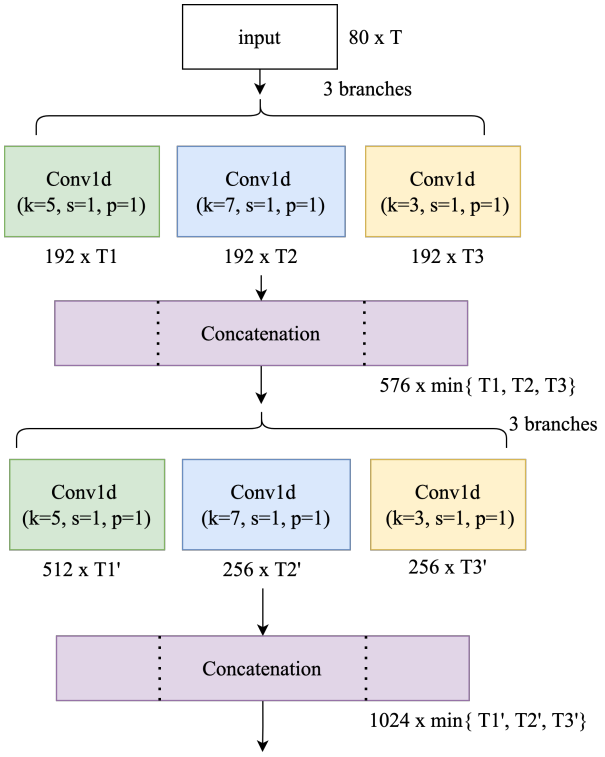

Figure 2: *The multi-scale convolution structures with three different scales of kernels, where k, s, p denotes kernel size, stride and padding respectively.*

waveform [4] which provides different scales of features at the same time.

In this work, we replace the simple 1-D CNN with three branches of multi-scale convolutions, where the kernel sizes are determined by convolution kernels. Each branch is composed of several layers of 1-D CNN as is shown in Figure 2. It should be noted that the number of layers for each branch is set to 2. Multi-scale convolution kernels with the size of $[5, 3, 1]$ are provided to extract different scales of embedding features. The stride and padding parameters of different kernels are all set to 1. After two layers of convolution, the features extracted from different branches are obviously different. Multiple feature maps are sliced to the minimum feature size and then all feature maps are concatenated along the dimension of channel. In this way, the proposed multi-scale convolution layers are able to provide multi-scale spatial information for a determined frame-level input.

2.3. Residual blocks in SE-Res2blocks

In the architecture of ECAPA-TDNN, the SE-Res2blocks consist of the SE blocks and Res2Net. These blocks are rather deep networks and composed of complex parameters for gradient propagation and backward computation. Thus, when training such deep networks, network non-convergence occurs frequently and affects feature extraction for speaker embeddings.

In response to this problem, we use the skip connections between SE-Res2blocks to prevent network degradation caused by complex parameters and deep layers in training. The building block is shown in Figure 3 and defined as:

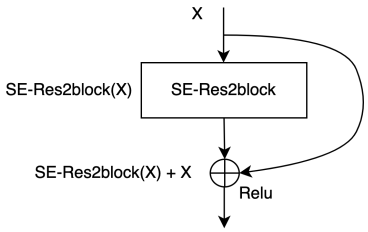

Figure 3: *The residual block of the SE-Res2block.*

$$Y = \mathcal{F}_{SE-Res2block}(X, W_i) + X \tag{1}$$

where X and Y represent the input and the output vectors of few stacked layers. $\mathcal{F}_{SE-Res2block}(X, W_i)$ denotes the learning layers of the SE-Res2blocks. $\mathcal{F}_{SE-Res2block}(X, W_i) + X$ of the building block represents the skip connection and the element-wise addition. Skip connection introduces no additional parameters as it has the same computation cost as the network before.

It should be noted that the dimension of x and $\mathcal{F}_{SE-Res2block}(X, W_i)$ must be equal. By going through several SE-Res2layers, the input and output vectors have the same dimension. The residual network can efficiently solve the model degradation problem of deep neural network.

2.4. Combination of multiple poolings

The basic ECAPA-TDNN model adopts channel-wise and context-wise dependent statistics pooling based on attention mechanism. This pooling method outperforms the traditional statistic pooling as it applies various temporal attention to each channel [5]. However, combination of multi-scale pooling methods can achieve better performances than single-scale pooling [6], which is widely applied in the filed of image retrieval.

In our work, two branches of different kinds of poolings are used after the last convolution layer of the network. One pooling method is the statistical attentive pooling method which focuses on channel-wise regions of utterance representation. Another pooling method is the SPoC [7] which activates larger regions on utterance representation and presents global descriptions of convolution features.

Given a fixed-length utterance U, the output of the hidden layer h is $2D$ tensor with a dimension of $C \times T$, where C denotes the number of feature maps and T denotes the length of frames. Let h_t $(C \times 1)$ and h_c $(1 \times T)$ denotes the hidden tensor of frame t and channel c. Thus, the hidden layer h can be a set of h_c defined as:

$$h = [h_1, h_2, ..., h_C]^T, c \in \{1, 2, ..., C\} \tag{2}$$

Then, the sum pooling method can be defined as:

$$\psi_{sumpool}(U) = \frac{1}{|h_c|} \sum_{t=1}^{T} h_t \tag{3}$$

The SPoC pooling vector is then produced by a fully-connected layer and an 1-D batch normalization layer represented as follows:

$$P_{SPoC} = f_{1d-BN}(W \cdot \psi_{sumpool}(U)) \tag{4}$$

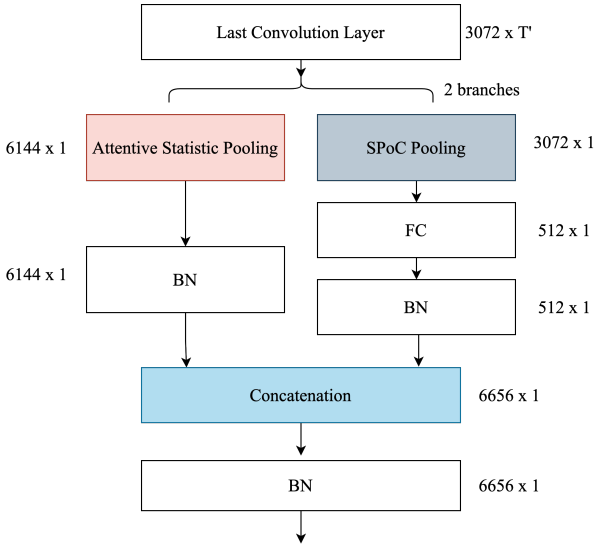

Figure 4: *The combination of attentive statistic pooling and SPoC pooling, where the last convolution layer denotes the 1-D CNN after concatenation of multiple SE-Res2blocks.*

where P is the pooling function of SPoc, f_{1d-BN} represents the batch normalization function and W represents the weight of fully-connected layer.

Finally, the pooling produced by SPoC and statistic attentive pooling are concatenated along the dimension of channel. These concatenated poolings are then passed into a batch normalization layer to effectively avoid gradient vanishing and improve convergence speed. The structure of the combination of pooling layers is presented in Figure 4.

3. Results

3.1. Comparision with the baseline system

As seen in Table 1, our proposed system gets a better performance than the baseline system in terms of the mAP evaluation metric. The performance of our system gets 19.6% and 22.7% on dev set and test set respectively. Our proposed system as a whole is more effective than the state-of-the-art ECAPA-TDNN system.

Table 1: *Comparision of mAP for several different systems on dev and test set. MC denotes multi-scale convolution layers, MP denotes combination of multiple poolings and residual denotes the skip connection layers.*

System	mAP	
	Dev	Test
Baseline (ECAPA-TDNN)	0.726	0.5773
Proposed	**0.868**	**0.7085**
Proposed − residual	0.860	0.6844
Proposed − MP	0.840	0.6439
Proposed − MC	0.851	0.6755
ECAPA-2-layer-TDNN	0.765	0.6199

3.2. Ablation study

We also adopt ablation study to prove each part of our proposed system is effective. The whole architecture is an assemble of the multi-scale convolution layers, residual learning blocks and the combination of multiple pooling layers. We eliminate these three components one by one to see the performance of our model.

As seen in Table 1, first, the residual blocks are removed. The mAP can achieve to 0.860 in dev set and 0.6844 in test set, where the score descends a little. The residual blocks can slightly improve the network and have a good effect on preventing overfitting. Second, the combination of SPoC pooling and statistic attentive pooling is removed and the system is trained with single statistic attentive pooling. It is shown that our proposed system relatively outperforms the system with no SPoC pooling by 15.7% in dev set and 11.5% in test set. Third, the multi-scale convolution layers are replaced by the traditional TDNN layer which is the same as single 1-D convolution. The multi-scale convolution layers lead to a relative improvement of 17.2% and 17.1% in dev and test set. One may argue that two layers of convolutions can definitely outperform the 1-layer TDNN. Therefore, we also experiment the baseline system with a TDNN of two layers. The performance of ECAPA-2-layer-TDNN is not as good as the the proposed multi-scale convolution layers. However, it also achieves a slightly relative improvement of 5.3% and 7.4% in dev and test sets. Overall, the results represent that the variant architectures proposed by us perform great improvements and the most effective method in the whole proposed architecture is the combination of multiple poolings.

4. References

[1] Yue Fan, JW Kang, LT Li, KC Li, HL Chen, ST Cheng, PY Zhang, ZY Zhou, YQ Cai, and Dong Wang, "CN-Celeb: a challenging chinese speaker recognition dataset," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7604–7608.

[2] Lantian Li, Ruiqi Liu, Jiawen Kang, Yue Fan, Hao Cui, Yunqi Cai, Ravichander Vipperla, Thomas Fang Zheng, and Dong Wang, "Cn-celeb: multi-genre speaker recognition," 2020.

[3] Xingyu Na Bengu Wu Hao Zheng Hui Bu, Jiayu Du, "Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline," in *Oriental COCOSDA 2017*, 2017, p. Submitted.

[4] Ge Zhu, Fei Jiang, and Zhiyao Duan, "Y-vector: Multiscale waveform encoder for speaker embedding," *arXiv preprint arXiv:2010.12951*, 2020.

[5] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[6] HeeJae Jun, Byungsoo Ko, Youngjoon Kim, Insik Kim, and Jongtack Kim, "Combination of multiple global descriptors for image retrieval," *arXiv preprint arXiv:1903.10663*, 2019.

[7] Artem Babenko and Victor Lempitsky, "Aggregating local deep features for image retrieval," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1269–1277.