Multiple Enhancements to LSTM for Learning Emotion-Salient Features in Speech Emotion Recognition

Desheng Hu, Xinhui Hu, Xinkang Xu
Hithink RoyalFlush AI Research Institute, Zhejiang, China
{hudesheng, huxinhui, xuxinkang}@myhexin.com

Abstract

Emotion-relevant feature extraction is key to the speech emotion recognition (SER) task. Although neural network for extracting features has achieved excellent results, in particular long short-term memory (LSTM) based models, there is still ample space for improvement. In this paper, from the perspective of utilizing advantages of multiple models, we propose an approach of multiple enhancements for learning emotion-salient features in SER, which is based on the combination of LSTM, one-dimensional convolution and transformer networks. Firstly, we introduce residual-BLSTM (Bidirectional LSTM) module to make the network deeper and to increase the learning ability of the model by adding feed-forward network (FFN) to the output of BLSTM and building residual connections at the same time. Secondly, time pooling employed in residual-BLSTM module is proposed to reduce features redundancy and overcome training overfitting. Finally, we propose an E-transformer module by combining transformer and convolutional neural network. This approach enables it to learn local information while capturing global dependencies. We conduct evaluations on the IEMOCAP dataset using the proposed methods, and it shows the state-of-the-art performances.

Index Terms: speech emotion recognition, residual-BLSTM, time-pooling, E-transformer

1. Introduction

Speech is the most convenient and efficient way of human communication, and the emotion information contained in speech plays a vital role in communication. Enabling machines to speak, think, and feel like humans has long been pursued in the field of artificial intelligence. The research of SER will promote the realization of this goal.

The key to speech emotion recognition systems is how to extract emotion-relevant features from a speech signal [1]. While various short-term and long-term features have been proposed [2], traditional SER systems focus on features engineering [3]. However, there is still no consensus on the more effective emotion features. Most of these approaches extract frame-level features on speech signals and apply statistical functions to obtain utterance-level emotion features [4]. Then, the features are fed into machine learning algorithms such as Support Vector Machine (SVM) for emotion classification [5]. In recent years, deep learning methods have gained great attention in SER. For example, deep neural networks (DNN), convolutional neural networks (CNN), and recurrent neural networks (RNN) are successfully used to learn emotion-relevant features from mel-spectrogram or raw waveform [6, 7, 8].

Due to its ability to trace the time-series properties of speech signals, LSTM is used to model speech emotion in many studies [9, 10, 11]. [9] applied the attention mechanism to the output of the LSTM in order to focus on the emotion-salient regions of a speech signal. [11] added CNN as BLSTM’s front-end network and employed center-loss to learn discriminative emotion features. However, these methods are deficient for learning emotion-salient features since they all use the default LSTM cell. Therefore, to enhance the ability of LSTM in SER tasks, some studies improved LSTM from its internal structure [12, 13, 14]. [12] proposed a new variation of LSTM, advanced LSTM, for modeling temporal context in SER. [14] proposed a dual-sequence LSTM architecture to process two mel-spectrograms of different time-frequency resolutions from each utterance, and obtained the state-of-the-art SER performance. These variants enhance LSTM by improving the internal structure at the expense of complex calculations.

Since LSTM focuses on dynamic temporal change of time series, many researchers combine it with transformer, which can draw global dependencies between input and output in sequence modeling [15], to compensate for LSTM’s inability to learn global dependencies [16, 17, 18, 19]. In [17], an architecture of RNN with multi-head self-attention was proposed as the emotion classifier, and the pre-trained features from an end-to-end automatic speech recognition (ASR) model were input to the classifier. In [18], extractions of high-level features from audio and semantic information from text were both used to improve interactions by transformer in SER. However, these SER systems only utilized the original transformer or multi-head self-attention. While transformer is good at modeling long-range global context, it is less capable of extracting fine-grained local feature patterns [20].

In this study, we consider LSTM-based methods to improve the learning ability of LSTM from the perspective of network depth and ease of training rather than improving the internal structure of the LSTM. Inspired by the winner scheme of Google Brain-Ventilator Pressure Prediction Competition[21], we introduce a residual-BLSTM module to SER that is to add a FFN to BLSTM output and a residual connection between BLSTM input and FFN output. This architecture is similar to ResNet [22] in image classification. It is a residual learning framework of CNN with enhanced network depth and ease of training. Furthermore, by borrowing the idea of pooling operation in CNN, we apply time pooling in the last residual-BLSTM module to deal with overfitting problem, filter out redundant features and make the network thinner. Motivated by conformer model in ASR [20], we further adopt an E-transformer architecture to combine CNN and transformer to model both local and global dependencies of feature sequence. E-transformer module is realized by adding a one-dimensional convolution layer to the input and output of the transformer, respectively, and establishing a residual connection between the two one-dimensional convolution layers.

The structure of this paper is organized as follows: Section 2 describes our proposed system; Section 3 presents experiments and analyzes these experimental results; Finally, section 4 summarizes the whole paper.
Learn the long-time dependence of sequences. BLSTM is composed of both forward LSTM and backward LSTM to learn not only historical information but also future information concerning current time. As a result, BLSTM can model time series better than a simple LSTM. For LSTM-based methods in SER, BLSTM is mainly used for temporal sequence modeling [13]. However, since the LSTM structure is complex, simply stacking LSTM tends to cause overfitting and limit network depth.

The modeling of residual-BLSTM module is shown in Equ.(1), and it consists of a BLSTM and a FFN. Here, \( X = [x_1, x_2, x_3, ..., x_t] \) represents the input to the BLSTM. The size of hidden units of BLSTM is 128, the same as the dimension size of the input. BLSTM is regularized using dropout with a probability of 0.2. The FFN contains two dense layers, with a relu activation function in the middle. The input dimension of FFN is 256, the middle dimension is 512, the output dimension is 128, and residual connections are used on the FFN output and BLSTM input. FFN can increase the nonlinear capability of the network and make it easy to construct residual connections.

\[
H = FFN(\tilde{H}) + X \quad (1)
\]

Where,
\[
\tilde{H} = BLSTM(X) \quad (2)
\]

\[
FFN(\tilde{H}) = \max(0, \tilde{H}W_1 + b_1)W_2 + b_2 \quad (3)
\]

Where, \( W_1 \) and \( W_2 \) are the weights of FFN’s two dense layers, \( b_1 \) and \( b_2 \) are the bias of FFN’s two dense layers. These parameters are obtained by random initialization.

2.3. Time pooling

By borrowing the idea of pooling operation in CNN, we apply time pooling (TP) across time dimensions to the last layer of residual-BLSTM module so that it can filter out redundant features. Since we use E-transformer module after residual-BLSTM module, we can also reduce the parameter number. The kernel size of time pooling is 2, and stride is 2. We also apply dropout with a probability of 0.2 before time pooling, and help regularize the network.

\[ T = TP(H) \quad (4) \]

where, \( H \) is the output of the residual-BLSTM module.

2.4. E-transformer module

Transformer is an attention approach based on encoder-decoder architecture. Although transformer can learn global dependencies on input sequences, it is less capable of extracting fine-grained local features [20]. E-transformer module is a new architecture that combines transformer with CNN so that it can focus on local features while learning about global dependencies.

The right part of Figure 1 illustrates E-transformer module. It composes of two Conv1d layers and transformer encoder layers. Two Conv1d layers are added before and after the transformer’s encoder layer, respectively, and we add residual connections between its inputs and outputs. The Conv1d layer includes one-dimensional convolution, relu activation function, and layer normalization. The convolution kernel size is 3, the stride is 1, and the output channel is 128. Layer normalization can speed up network convergence.

The architecture of transformer encoder layers are shown in Figure 2, including multi-head self-attention mechanism (MHSA) sub-layers and FFN sub-layers. A residual connection
is employed around two sub-layers, followed by layer normalization. The number of heads is 8, and the size of $d_{model}$ is 128. For input $T$ to E-transformer module, the output $Y$ of this module is formalized as:

$$\tilde{T} = \text{Conv1d}(T)$$  \hfill (5)

$$T' = \text{MHSA}(\tilde{T})$$  \hfill (6)

$$T'' = \text{FFN}(T')$$  \hfill (7)

$$Y = \text{Conv1d}(T'') + T$$  \hfill (8)

where, $T$ is the output of time pooling.

2.5. Statistics pooling

To get an utterance-level feature representation $P$ for emotion classification, we apply the statistics pooling layer for feature sequence over all frames. It is performed first by computing the mean and standard deviation, and then concatenating the mean and standard deviation to obtain the second-order statistics of the feature sequence:

$$P = \text{Concat}(\text{mean}(Y), \text{std}(Y))$$  \hfill (9)

where, $Y$ is the output of E-transformer module.

3. Experiments and analysis

3.1. Experimental dataset and settings

The dataset IEMOCAP (Interactive Emotional Dyadic Motion Capture Database) [23] is used for experiments. It contains 12 hours of emotional speech performed by 10 actors. The dataset is divided into two parts, improvised part and scripted part. That is, the actors perform according to the script and improvisation. In general, the accuracy classification of the improvised part is higher than that of the scripted part because actors do not need to pay attention to the content of the words and express emotions more naturally. This study utilizes both scripted and improvised data for training and evaluations. The utterances are labeled with 9 types of emotion, including anger, happiness, excitement, sadness, frustration, fear, surprise, other and neutral state. To make fair comparisons with previous studies, we also merge happy and excited, and we use four emotions: happy, sad, angry, and neutral, and a total of 5531 utterances are selected.

In experiments, mel-spectrogram is extracted from speech as input feature. We use a 25-msec Hamming window in the feature extraction, with a 10 msec shift. Each mel-spectrogram is constructed using the output of a 40-dimensional mel-scale. We set the maximal audio length to 4s. Longer ones are truncated at 4s, and shorter ones are padded with zeros.

For experiments, the PyTorch deep learning framework is used. Here, the loss function is cross-entropy, and the optimizer is Adam with a learning rate of 0.0001. The batch size is set to 32. We randomly shuffle the data and make 5 fold cross validation.

The unweighted accuracy (UA) and the weighted accuracy (WA) are used as evaluation criteria for system performances. The WA is the classification accuracy of all utterances, and UA is the average individual emotion classification accuracy.

3.2. Comparisons with previous work

Recently, an LSTM-based framework, CNN-BLSTM, has been actively used as a model for SER tasks, and CNN layers here are utilized as the front-end network for it [11] [14]. This paper compares the above model, some of its variants and CNN-based methods, with our proposed method. Table 1 shows these comparisons, including our method and four other representative methods.

Table 1: The results of comparative experiments on IEMOCAP

<table>
<thead>
<tr>
<th>model</th>
<th>WA(%)</th>
<th>UA(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS-LSTM [14]</td>
<td>69.40</td>
<td>69.50</td>
</tr>
<tr>
<td>ACNN [24]</td>
<td>67.28</td>
<td>67.94</td>
</tr>
<tr>
<td>Semi-supervised AAE[25]</td>
<td>-</td>
<td>68.80</td>
</tr>
<tr>
<td>our method</td>
<td>69.31</td>
<td>70.11</td>
</tr>
</tbody>
</table>

The table shows that our method improves by 3.91% on WA and 3.21% on UA, compared to the CNN-BLSTM (center-loss) model [11]. That demonstrates significant performance improvements with multiple enhancements to BLSTM. The recognition results of DS-LSTM model [14] are close to ours. However, while the LSTM is a four-gated RNN, the DS-LSTM is a six-gated RNN with computational complexity. Moreover, it needs two mel-spectrograms at different time-frequency resolutions.

We also compare with two CNN-based methods, ACNN model [24] and Semi-supervised AAE model [25]. The former proposed an improved mechanism head fusion based on CNN. The latter used two-dimensional convolution to build the basic model and employed a multi-task learning framework and semi-supervised technique to improve performance. As a result, our method improves UA by 1.31% compared with the latter one. In all, based on these comparisons, they demonstrate the excellence of our proposed method.

3.3. Component contribution analyses

For investigating component contributions, following models are also used for experiments. These models are realized by some operations on our proposed model.

- Base1: w/ BLSTM, employing BLSTM instead of residual-BLSTM module.
Base2: w/ transformer, employing transformer instead of E-transformer module.

Base3: w/ BLSTM and transformer: BLSTM and transformer are employed instead of residual-BLSTM and E-transformer modules.

Base4: w/o time pooling: time pooling is not used.

Base5: Conv-residual-BLSTM: both 3-Conv-Layers and residual-BLSTM modules are employed for learning emotion features.

Base6: Conv-BLSTM: both 3-Conv-Layers module and BLSTM are employed for learning emotion features.

Base7: BLSTM (baseline): only BLSTM is employed for learning emotion features.

To measure the contribution of residual-BLSTM, E-transformer, and time pooling module to the whole system, we conduct ablation experiments using Base1, Base2, Base3, and Base4.

**Table 2: Comparisons among different models**

<table>
<thead>
<tr>
<th>model</th>
<th>WA(%)</th>
<th>UA(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base1: w/ BLSTM</td>
<td>67.21</td>
<td>68.21</td>
</tr>
<tr>
<td>Base2: w/ transformer</td>
<td>68.04</td>
<td>68.79</td>
</tr>
<tr>
<td>Base3: w/ BLSTM and transformer</td>
<td>67.01</td>
<td>67.99</td>
</tr>
<tr>
<td>Base4: w/o time pooling</td>
<td>68.61</td>
<td>69.70</td>
</tr>
<tr>
<td>Base5: Conv-residual-BLSTM</td>
<td>67.48</td>
<td>67.88</td>
</tr>
<tr>
<td>Base6: Conv-BLSTM</td>
<td>65.38</td>
<td>66.08</td>
</tr>
<tr>
<td>Base7: BLSTM (baseline)</td>
<td>61.48</td>
<td>62.83</td>
</tr>
<tr>
<td>our method</td>
<td>69.31</td>
<td>70.11</td>
</tr>
</tbody>
</table>

The experimental results using these models are shown in Table 2. This table shows that applying residual-BLSTM module, our method achieves 2.1% and 1.9% absolute improvements in WA and UA, respectively, compared with Base1. This indicates the effectiveness of applying residual-BLSTM module in deeper network for SER. When applying E-transformer module, our method improves WA and UA by 1.27% and 1.32% compared with Base2. It clearly shows the importance of combining convolution network with transformer for learning features in SER. We also experiment with adding the number of Conv1d layers before and after the transformer encoder layers, but we don’t show results because of the number of pages. When applying both E-transformer and residual-BLSTM modules, our method improves 2.30% and 2.12% in WA and UA compared with Base3. When applying time pooling, our method does not improve performances so largely compared with Base4. However, it significantly reduces the parameters by two in E-transformer module.

To verify the contribution of the residual-BLSTM module in the absence of transformer or E-transformer module, we built another two models, Base5 and Base6. The difference between these two models is whether there is a connection between the residual module and the FFN module. Except for this difference, the other parameters are the same for these two models. By comparing Base5 and Base6 in Table 2, it is found that the Conv-residual-BLSTM model improves WA and UA with 2.1% and 1.8%, respectively, which confirms the benefit of applying residual connection to BLSTM and residual-BLSTM module is still effective in a shallower network.

The results show that our methods improve by 7.83% and 7.28% absolute improvement in WA and UA over Base7 in table 2, proving that our multiple enhancements to BLSTM can learn emotion-salient features. By comparing Base6 and Base7, the Conv-BLSTM model achieves 3.9% and 3.25% improvement in WA and UA. This indicates that adding CNN layer as the front-end network of BLSTM can significantly improve the recognition performance.

### 3.4. Confusion matrices

To compare the recognition results of different emotions, four confusion matrices are shown in Figure 3. It shows that our approach outperforms all other models and performs best in three emotions except for happy ones. It was generally accepted that the classification of happy emotion in IEMOCAP was difficult due to the dataset annotation problems and other relevant factors.

### 4. Conclusions

This study proposes a method for learning emotion-salient features using multiple enhancements in speech emotion recognition. These enhancements are embodied in the following respects. First, with the residual-BLSTM module, it is proven that the module is easy to train, the network depth can be deeper, and more emotion-salient features can be obtained. Second, employing time pooling in the last residual-BLSTM module can filter out redundant features and reduce overfitting. Third, the network can model both local and global dependencies of feature sequence with E-transformer module. We conduct experiments to investigate the proposed model’s effectiveness by comparing it with the other seven models. For the dataset IEMOCAP, our proposed model outperforms the other previous state-of-the-art SER models achieving a WA of 69.31%, a UA of 70.11%. Compared with the BLSTM model, a 7.83% improvement in WA and a 7.28% improvement in UA are achieved.
5. References


