ISBN 978-981-14-1684-2
Proceedings of 2019 the 9th International Workshop on Computer Science and Engineering
(WCSE 2019 SUMMER)
Hong Kong, 15-17 June, 2019, pp. 295-301
doi: 10.18178/wcse.2019.06.044

Speech Emotion Recognition using Convolutional Neural Networks and Recurrent Neural Networks with Attention Model

Xi He[+], Liyong Ren and Yongbin He

University of Electronic Science and Technology of China, Chengdu, China

Abstract. Speech emotion recognition is an essential step in advanced human-computer speech interaction. Most of researchers focus on the entire speech sequence without handling emotionally-irrelevant speech frames specifically. In this study, a novel deep recognition framework is proposed, which using attention mechanism to focus on speech segments with salient emotion. The framework involves two stages. In the first stage, the unconstrained sparse auto-encoder is used to learn the convolution kernel, and the local salient features are extracted using convolutional neural networks(CNNs). In the second stage, the local salient features are aggregated into a high-level representation using bidirectional long short-term memory(BLSTM) with attention model. The experimental results on different language data sets show that the framework leads to higher accuracy and outperformed the conventional methods by about 12.32%.

Keywords: convolutional neural networks, bidirectional long short-term memory, auto-encoder, attention model, speech emotion recognition

1. Introduction

In Human-Computer Interaction(HCI) area, how to recognize different emotions from human speech plays a significant role. Speech emotion recognition(SER) aims to accurately classify speech emotions into different categories , such as anger, happiness, sadness, surprise, ect.

Speech emotion recognition includes three key steps: feature extraction, feature selection and emotion classification. One of the key issue to improve performance is how to select an optimal feature sets. Meanwhile, how to achieve the aggregation of features that are more informative about emotions is still a problem worth studying[1].

2. Related Work

Speech emotion recognition is a challenging task mainly because different individuals express and perceive emotions in different ways [2]. In addition, There is no commonly agreed theoretical definition on emotion[2]. In this paper, we use the model of discrete classes[4] to represent emotion.

Recently, with the advent of deep neural networks, researches tend to automatically extract informative features that can efficiently characterize the emotional content of speech signals[5,6,7,8]. However, it is unclear if these methodologies can sufficiently represent the dynamic mood wings in different segments of speech sequences. Therefore, attention mechanism which makes it possible to focus on specific parts can be applied in speech emotion recognition.

Attention-based models have been successfully applied to many tasks, such as abstractive sentence summarization[9], machine translation[10], natural language inference[11], image caption generations[12] and speech recognition[13]. Mirsamadi et al [14] proposed a deep recurrent neural network with a feature weighted-pooling strategy which use local attention mechanism to concentrate on specific emotive regions of a speech signal. The weighted accuracy and unweighted accuracy are 63.4% and 53.8%, respectively. It has

[+] Corresponding author. Tel.: + 86 18380129535; fax: none.
E-mail address: 201721220209@std.uestc.edu.cn; 1719551483@qq.com

been shown that there is a large room for improvement in the accuracy of attention-based speech emotion recognition. However, there is little work exploring useful frameworks or architectures for this. In this paper, we proposed a deep recognition framework which combines CNN and attention-based BLSTM can effectively improve the accuracy.

3. Deep Recognition Framework

The deep recognition framework aims to extract high-level aggregation features from the raw speech signal and effectively classify emotions in speech signals. The proposed model is shown in Figure1, which has an input layer, one convolutional layer(with auto-encoder kernel), one BLSTM layer, attention mechanism, one pooling layer, one fully connected layer and a softmax layer.

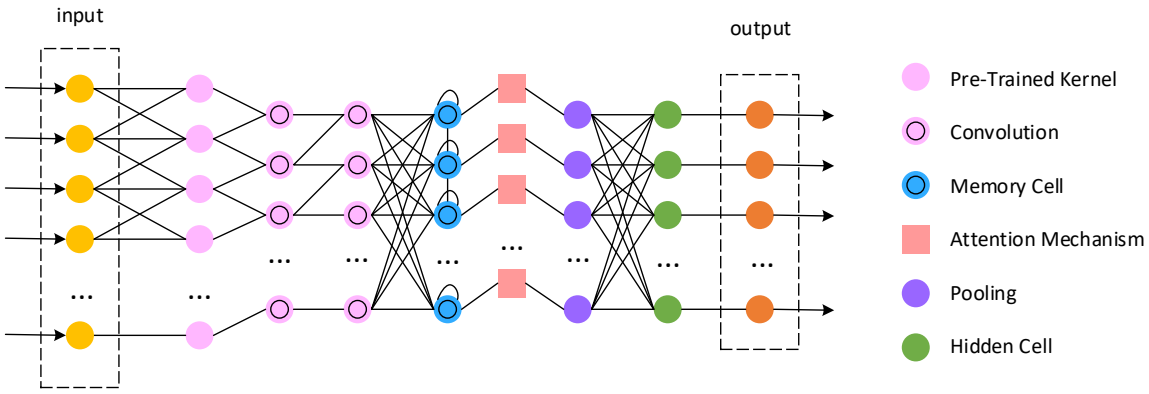

Fig. 1: The Architecture of Deep Recognition Framework

3.1. Learn salient feature representation

The convolutional neural network is intended to learn local salient features from spectrogram. A convolutional neural network consists of an input layer, an output layer and multiple hidden layers which consists of convolution layers, activate function, pooling layers, fully connected layers and normalization layers. Inspired by the work in [15], the convolution kernel is trained on randomly sampled spectrogram patches using unsupervised sparse auto-encoder. More specifically, the convolutional layer uses a total of 180 three fixed sized 3×60, 6×60 and 10×60 kernels.

An auto-encoder is generally used to learn the compressed representation of data, including the process of encoder and decoder [24]. The encoder learns to map the input to the feature vector, and the decoder reconstructs the input by minimizing the reconstruction error. It can be used as a tool for extracting common representation from the input in deep neural networks [16], usually by training each layer separately and passing the current layer output to the next layer.

We use the encoder function h to map the low-level features of the input x to a potential vector representation $\hat{x}$. The function is given as follows:

$$h(x) = f(Wx+b) \tag{1}$$

where function $f(z)=1/(1+e^{-z})$ is the non-linear activation function applied component-wise[19], $W \in m\times n$ is a weight matrix, $b \in R^n$ is a bias vector. m is the number of hidden units and n is the number of input units. The output value $\hat{x}$ is as the same formula as h with different values for W and b.

The training set T in the task is defined as consisting of a series records. The test set is consistent with the format of the training set, defined as Γ. Auto-encoder trains the data $x_i \in T, i=1,...,k$ first, then adjusts the parameters $\theta=(W,b)$ by back propagation to minimize the reconstruction error:

$$\min_{\theta} \sum_{x\in\Gamma} \|x_i - \hat{x}_i\|^2 \tag{2}$$

In order to maintain a low average activation for each unit, an additional sparse penalty $\Omega(h)$ is added to the reconstruction error. The penalty can be written in the form:

$$\Omega(h)=\mu\sum_{j=1}^{m}\left(\rho\log\frac{\rho}{\hat{\rho}_j}+(1-\rho)\log\frac{1-\rho}{1-\hat{\rho}_j}\right) \tag{3}$$

where $\hat{\rho}_j=1/k\cdot\sum_{i=1}^{k}h_j(x_i)$ is the average activation degree of the hidden unit j on the training set, the non-negative parameter ρ is the sparse level, and μ controls the weight of the penalty item.

Sparse self-encoder is used to learn convolution kernels of different sizes from the spectrogram. After pre-trained by the above method, each sparse self-encoder obtains a core (W_i,μ_i). The corresponding feature map l^i on the entire spectrogram is calculated for each block of the spectrogram with a different size convolution kernel [15].

$$l^i(x)=f(c(W^i,x)+\mu^i) \tag{4}$$

The function c represents a convolution operation. Subsampling is then performed, the number of sampling windows is N, and all sampling windows take the mean value to obtain a down-sampled value $L^i(x)$ of the block t in the convolution kernel i.

$$L_t^i(x)=[\bar{l}_1^i(x),\ldots,\bar{l}_N^i(x)] \tag{5}$$

After calculating the mean and variance of $L_t^i(x)$, a feature vector $L^i(x)$ of a relatively different size convolution kernel is obtained, and the output vector y is integrated. Assuming the total number of cores is k, the output of the convolution layer is given as follows:

$$y=[L^1(x),\ldots,L^k(x)] \tag{6}$$

The output is passed to the next layer of the deep recognition neural network. The process of obtaining intermediate feature representation through the CNN is shown in Figure 2.

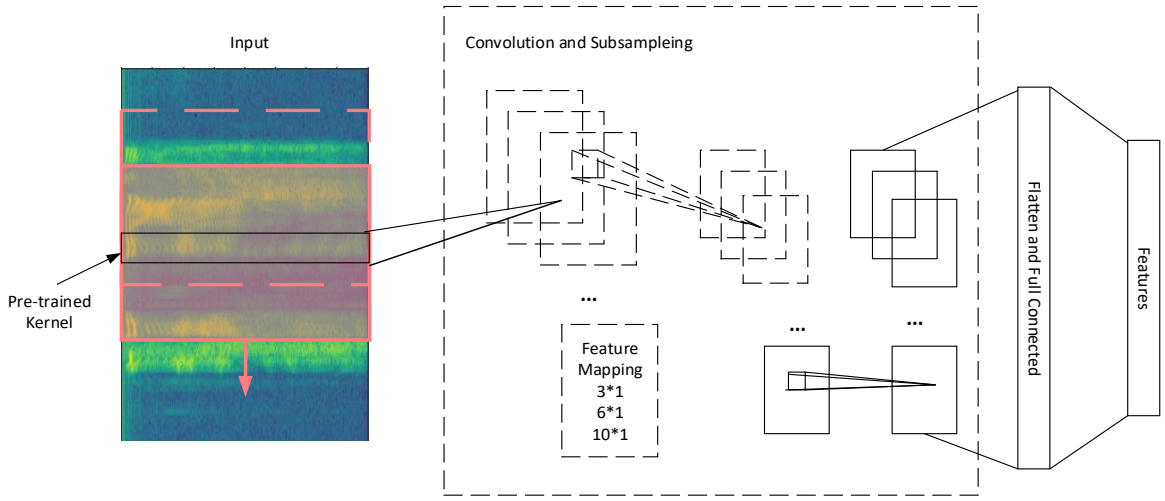

Fig. 2: The process of learning salient feature representation

3.2. Attention-based pooling recurrent neural networks

The deep feedforward network is a deep learning model for out-of-order data. A number of neighborhood frames can be overlapped with the current frame to extend for sequential data. However, for speech emotion data, context effect won't modeled well by a fixed window. So we use recurrent neural network as the next step and take the output of CNN as its input. It has been proven that RNN can be extended to longer sequences through feedback connections.

LSTM is a special model of RNN proposed by Hochreiter and Schmidhuber[16]. They proposed to replace hidden neurons with one or more memory blocks to effectively learn long-term dependence information. Every LSTM memory block consists of self-connected memory cells, an input gate, a forget gate and an output gate. The input gate adjusts cell input, the output gate controls cell output, and the forget gate changes the self-loop weight. The calculation process for different state transitions are as follows:

$$i_t = \sigma\left(W_{xi} \cdot x_t + W_{hi} \cdot h_{t-1} + b_i\right) \tag{7}$$

$$f_t = \sigma\left(W_{xf} \cdot x_t + W_{hf} \cdot h_{t-1} + b_f\right) \tag{8}$$

$$o_t = \sigma\left(W_{xo} \cdot x_t + W_{ho} \cdot h_{t-1} + b_o\right) \tag{9}$$

where i_t, f_t, o_t denote the output states of the input gate, the forget gate and the output gate at time t, respectively. W_* is a weighted matrix connecting different gates; x_t is the input of the cell at time t and h_t represents the output state of the cell at time t. and b_i, b_f, b_o denote the block bias of the input gate, forget gate and output gate,respectively. The weight of self-connected memory cell is controlled by the forget gate. The calculation is as follows:

$$c_t = f_t c_t + i_t \tanh\left(W_{xc} \cdot x_t + W_{hc} \cdot h_{t-1} + b_c\right) \tag{10}$$

where c_t represents the state of the cell at time t, and b_c represents the corresponding bias vector. Therefore the current cell output h_t is:

$$h_t = o_t \tanh\left(c_t\right) \tag{11}$$

Although the LSTM network can effectively model speech signals, it only uses historical information and does not consider future information. Therefore, the proposed model use bidirectional LSTM to represent different long-term integration over the intermediate features.

One important issue of this structure is how to train the parameters. This paper introduces a new weighted-pooling strategy used in [14], which is generated from Attention-based Recurrent Sequence Generator(ARSG). The strategy focuses on special parts of an utterance which contains strong emotional characteristics. We add a weighted-pooling layer on top of the LSTM layer, and the weights of different frames are determined by the results of the parameter generation according to the attention model. The attention mechanism randomly selects an input sequence for updating the hidden state of the BLSTM and predicting the next output value. Finally, the sequence weights are weighted average in time to obtain a high-level representation y.

$$y = \sum_t \alpha_t h_t \tag{12}$$

where α_t demotes the attention wight. The inner product between attention model parameter ϑ and the output of BLSTM h_t represent the proportion in the emotional expression. A softmax function is applied to make all the frame ratios equal to one.

$$\alpha_t = \frac{\exp\left(\vartheta^T h_t\right)}{\sum_{\Gamma=1}^{T} \exp\left(\vartheta^T h_\Gamma\right)} \tag{13}$$

The weighted results are ultimately passed to the softmax layer of the network to obtain the posterior probability of each type of emotion. Both the parameters of the attention mechanism and BLSTM network are trained by backpropagation.

4. Experiments

4.1. Datasets and experimental setup

To evaluate the performance of the proposed deep recognition framework and its applicability to different languages, we apply the model to three different language standard databases in speech emotion recognition tasks: the CASIA Chinese Emotional Speech Database [18], the Berlin German Emotional Speech Database(EMO-DB) [19] and the IEMOCAP English Emotional Speech Database [20].

Each speech emotion database was processed separately in the experiment. In the phase of unsupervised learning convolution kernel, we randomly select one-third of the data to train the convolution kernel in the training set of each speech emotion database.

In our experiment, the spectrogram of speech signals is first drawn. The spectrogram has a 20 ms window size with a 10 ms overlap. The 256-dimensional Fast Fourier Transform (FFT), fundamental frequency, formant, frame energy, zero-crossing rate and 13-dimensional Mel-Frequency Cepstral Coefficient (MFCC) are extracted as the low-level descriptors(LLDs) of speech frames. These features are normalized by global mean and standard deviation based on neutral speech characteristics in the training set.

Then, we conducted four comparative experiments on each database to compare the performance of proposed model with the previous model.

Experiment one: After extracting the LLDs characteristics of the data set samples, the higher function calculation is performed. Then use the support vector machine with radial basis function(RBF) kernel to realize multi-classification through libsvm [20] experiment library. The complexity parameter is optimized by logarithmic grid in the range of 10^{-6}-10^{0}.

Experiment two: A Rectified Linear(ReLU) full connection layer with 512 nodes is used for LLD learning, and then a 128-cell BLSTM recurrent layer for learning temporal aggregation.

Experiment three: The dimensionality of spectrogram is reduced by principal component analysis(PCA), and finally 60 channels of one-dimensional vectors of length n is obtained, which is fed into the convolutional neural network to learn intermediate feature representation. The auto-encoder is pre-trained with patches randomly extracted from different locations, using three different sizes of convolution kernels: 3×60, 6×60, and 10×60. Then use a 128-cell BLSTM recurrent layer to learn temporal aggregation.

Experiment four: The weighted-pooling strategy with attention mechanism was added into the former experiment, and two hidden layers were selected, each consisting of 30 memory cells. During the network training, a gradient of learning rate of 10^{-6} and a momentum of 0.9 is achieved. Zero-mean Gaussian noise with a standard deviation of 0.1 was added to the input activation during the training phase. The weights in all BLSTM and attention mechanisms were randomly initialized from -0.1 to 0.1.

The parameters of the four experiments were optimized on the validation set with 50% dropout on all layers to avoid over-fitting. In order to speed up the training process, the network weights are updated after running every 10 sequences to achieve parallel computing.

4.2. Experimental results

There are several ways to evaluate performance of the proposed model. In this paper, we use the confusion matrix to demonstrate the experimental results of the deep recognition framework. The results of the three data sets trained using the proposed model are shown as confusion matrices in Figure 3. The row represents the standard emotion and the column represents the recognition emotion from the model. The darker the color, the higher the corresponding recognition rate.

	Happy	Sad	Angry	Scared	Surprised	Neutral
Happy	73.1	0.9	5.1	4.4	13.4	3.1
Sad	0.8	74.9	1.4	7.6	3.4	11.9
Angry	2.0	0.0	81.7	6.4	8.8	1.1
Scared	3.8	11.1	3.2	69.6	8.2	4.1
Surprised	4.4	1.4	6.6	15.1	72.3	0.2
Neutral	1.2	10.9	2.7	9.4	3.2	72.6

(a)CASIA

	Happy	Sad	Angry	Neural
Happy	82.4	4.2	7.8	5.6
Sad	2.7	80.4	7.4	9.5
Angry	9.8	3.2	84.1	2.9
Neural	4.4	7.4	3.3	84.9

(b)IEMOCA

	Anger	Joy	Sadness	Fear	Disgust	Boredom	Neutral
Anger	82.1	8.1	2.4	4.5	2.5	0.0	0.4
Joy	5.6	77.5	0.0	7.7	2.1	5.2	1.9
Sadness	0.8	0.0	87.8	3.4	0.0	3.8	4.2
Fear	0.0	3.2	4.0	76.5	6.3	6.9	3.1
Disgust	0.0	4.3	2.1	10.7	75.2	3.4	4.3
Boredom	2.3	4.6	0.0	3.7	5.7	71.6	12.1
Neutral	2.0	4.8	2.3	2.7	3.2	9.9	75.1

(c)EMO-DB

Fig. 3: Confusion matrix using Deep Recognition Framework on three different language datasets

According to the confusion matrix, the weighted average values of the accuracy of the proposed model applied in the CASIA, the EMO-DB and the IEMOCAP are 74.03%, 77.97% and 82.95%, respectively. The accuracy is calculated by dividing the number of correctly classified in all categories by the total number. It

can be seen from the results that the proposed model performs best in identifying anger and happy emotions, which means that the extracted features can more accurately identify the emotions with greater fluctuations.

The comparison of the four experiments on the three data sets is measured by the weighted accuracy(WA) and shown in Figure 4. Compared to the low-level descriptor and support vector machine used in experiment one, the BLSTM used in experiments two can improve the weighted accuracy by about 4.72%, and after replacing the LLDs with local salient features, the performance can be improved by 4.2%. The attention mechanism improved performance by approximately 3.4% in experiment four, which was increased to 75.82% compared to the 63.5% accuracy in [14].

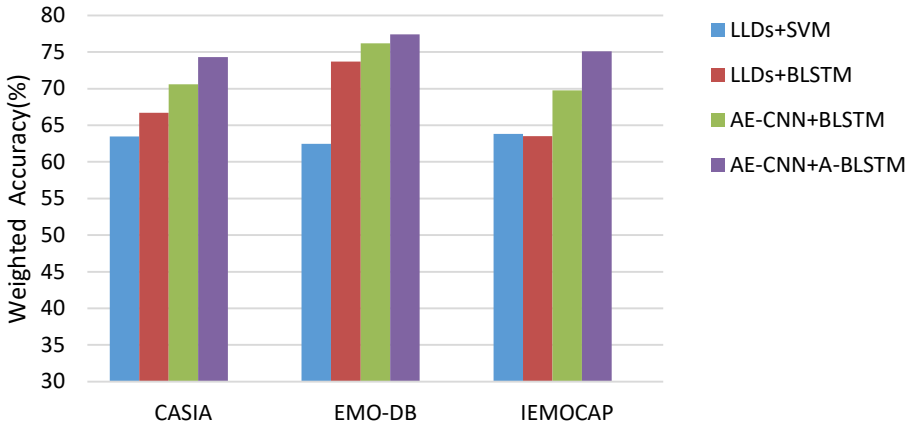

Fig. 4: The recognition accuracy in CASIA,EMO-DB and IEMOCAP dataset by four experiments

We also compare the differences between LSTM and BLSTM in this model. The unweighted average recall(UAR) can be significantly improved by increasing the number of context frames, but as the number of frames exceeds the length of the training frame, the performance improvement tends to be flat.

5. Conclusion and Future Work

This paper proposes a speech emotion recognition framework using convolutional neural networks and recurrent neural networks with attention model. This framework can effectively extract the emotional features of speech signals and solve the problem of emotional label uncertainty in speech. The experimental results show that compared with several typical models, the framework has better feature extraction ability and can achieve higher emotional recognition accuracy. In the future, we can expand the proposed method to achieve domain adaptation.

6. References

[1] Björn W. Schuller. Speech emotion recognition: two decades in a nutshell, benchmarks, and ongoing trends[J]. Communications of the Acm, 2018, 61(5):90-99.

[2] Anagnostopoulos C.N , Iliou T , Giannoukos I . Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011[J]. Artificial Intelligence Review, 2015, 43(2):155-177.

[3] Kleinginna P.R , Kleinginna A M . A categorized list of emotion definitions, with suggestions for a consensual definition[J]. Motivation and Emotion, 1981, 5(4):345-379.

[4] Swain M , Routray A , Kabisatpathy P . Databases, features and classifiers for speech emotion recognition: a review[J]. International Journal of Speech Technology, 2018.

[5] André Stuhlsatz, Meyer C , Eyben F , et al., “Deep neural networks for acoustic emotion recognition: raising the benchmarks,” in 2011 IEEE International Conference on Acoustics, Speech and Signal Pro- cessing (ICASSP). IEEE, 2011, pp. 5688–5691.

[6] Lee J. and Tashev I., “High-level feature representation using recurrent neural network for speech emotion recognition,” in Interspeech, 2015.

[7] Zhang S. , Huang T. and Gao W. , "Speech Emotion Recognition Using Deep Convolutional Neural Network and Discriminant Temporal Pyramid Matching," in IEEE Transactions on Multimedia, vol. 20, no. 6, pp. 1576-1590, June 2018.

[8] Trigeorgis G , Ringeval F , Brückner, R, et al. Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network[C]// IEEE International Conference on Acoustics. IEEE, 2016.

[9] Rush, Alexander M. , Chopra S. , and Weston J.. "A Neural Attention Model for Abstractive Sentence Summarization." Computer Science (2015).

[10] Bahdanau D. , Cho K., and Bengio Y.. Neural machine translation by jointly learning to align and translate. In Proc. Of the 3rd ICLR, 2015.

[11] Parikh A P , Täckström, Oscar, Das D , et al. A Decomposable Attention Model for Natural Language Inference[J]. 2016.

[12] Xu K , Ba J , Kiros R , et al. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention[J]. Computer Science, 2015.

[13] Chorowski J , Bahdanau D , Serdyuk D , et al. Attention-Based Models for Speech Recognition[J]. Computer Science, 2015, 10(4):429-439.

[14] Mirsamadi S , Barsoum E , Zhang C . Automatic Speech Emotion Recognition Using Recurrent Neural Networks with Local Attention[C]// ICASSP. IEEE, 2017.

[15] Mao Q , Dong M , Huang Z , et al. Learning Salient Features for Speech Emotion Recognition Using Convolutional Neural Networks[J]. IEEE Transactions on Multimedia, 2014, 16(8):2203-2213.

[16] Goodfellow I.J , Le Q.V , Saxe A.M , et al. Measuring Invariances in Deep Networks.[C]// International Conference on Neural Information Processing Systems. Curran Associates Inc. 2009.

[17] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735-1780.

[18] Institute of Automation Chinese Academy of Sciences. The selected Speech Emotion Database of Institute of Automation Chinese Academy of Sciences (CASIA) [DB/OL].2012/5/17.

[19] Burkhardt, Felix, et al. "A database of German emotional speech." Ninth European Conference on Speech Communication and Technology. 2005.

[20] Busso, Carlos, et al. "IEMOCAP: Interactive emotional dyadic motion capture database." Language resources and evaluation42.4 (2008): 335

[21] Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST), 2(3):27.