Vol.2021-NL-251 No.7
Vol.2021-SLP-139 No.7
2021/12/1

Effective Integration of Transformer for Network-based Speech Emotion Recognition

Yurun He[1] Nobuaki Minematsu[1] Daisuke Saito[1]

Abstract: The performance of a speech emotion recognition (SER) system heavily relies on deep representations learned from training samples. Recently, transformer has exhibited outstanding properties in learning relevant representations for this task. However, to better fuse it with conventional models, experimental investigations are still needed. In this paper, we attempt to take advantage of several integrations of transformer with two most widely used deep learning models - CNN and BLSTM. Experiments on the IEMOCAP benchmark dataset demonstrate that the proposed approaches can make a promising improvement.

1. Introduction

Speech emotion recognition (SER), referring to the process of detecting the emotional state of a speaker, has become a very active research topic in the affective computing field and has had a wide range of applications which require human-computer interaction (HCI) such as call center conversation, in-car board system, and mobile communication [1]. Due to its practical importance, SER has received substantial attention from both academia and industry. However, as of now, it still remains a challenging technical problem due to the inherent subtlety of human emotions [2]. To successfully implement a speech emotion recognition system, emotion need to be defined and modeled carefully. Discrete and dimensional emotional models are two widely used approaches. The former is based on six basic emotions: sadness, happiness, fear, anger, disgust, and surprise, the latter uses valence, arousal, and dominance to describe emotion quantitatively. In this paper, we will only consider the discrete one.

SER aims to identify the high-level affective status of an utterance from the low-level features. It can be treated as a classification problem on sequences [3]. In the past, people have come with many different methods, most of which extract a large amount of complex low-level handcrafted features (such as pitch, Mel-frequency cepstrum coefficients (MFCC) and so on) out of the initial utterance and then apply conventional classification algorithms like Hidden Markov Model (HMM) [4] and support vector machines (SVM) [5]. In recent years, the boom of deep learning has exhibited outstanding performances in extracting discriminative features for SER. [3] proposed to use the segments with highest energy to train a Deep Neural Network (DNN) model to extract effective emotional information. [6] first used convolutional neural networks (CNN) to learn affective-salient features for SER and showed excellent performances on several benchmark datasets. [7] applied a long short-term memory (LSTM) to learn long-range temporal relationships for SER. In [8], they directly used raw audio samples to train a convolutional recurrent neural network (CRNN) to build a continuous arousal and valence space.

More recently, attention-based deep-learning approaches have started finding their application for SER. In [9], attention layers are used to focus on the emotional relevant parts and produce utterance-level affective-salient features for SER. In other researches, the authors showed the efficiency of transformer on the SER task [2], [10], [11]. However, the interaction between transformer and other deep learning structures is still needed to be investigated. In this paper, we propose two different approaches to utilize network-based structure aggregated with transformer. We first intergrate LSTM with transformer (ILT), trying to replace the function of positional encoding of transformer by LSTM. Then we utilize the cross attention transformer (CAT), which aims to interact and combine the information obtained from CNN and LSTM. Our experiments show both methods outperform our baseline system, indicating their strength for emotion recognition.

The remainder of this paper is organized as follows. We describe the transformer and our proposals in Section 2. In Section 3, we briefly introduce the IEMOCAP database used in the experiment and the experimental setup. Experiments are addressed and their results are analysed in Section 4. Finally, Section 5 presents the conclusions and future works.

2. Model Architecture

2.1 Transformer

Due to the poor ability of RNN families in solving the

[1] The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

phenomenon of special long-term dependence, [12] proposed transformer, which exploits the self-attention mechanism to reduce the distance between any two positions in the sequence to a constant. Given an input sequential matrix as $\mathbf{X} \in \mathbb{R}^{d_t \times d_f}$, by multiplying with three different trainable weight matrix $\mathbf{W}^Q \in \mathbb{R}^{d_f \times d_k}, \mathbf{W}^K \in \mathbb{R}^{d_f \times d_k}, \mathbf{W}^V \in \mathbb{R}^{d_f \times d_v}$, we can obtain the set of queries $\mathbf{Q} \in \mathbb{R}^{d_t \times d_k}$, the set of keys $\mathbf{K} \in \mathbb{R}^{d_t \times d_k}$, and the set of values $\mathbf{V} \in \mathbb{R}^{d_t \times d_v}$. Then the self-attention can be calculated as:

$$\mathbf{Z} = Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V}, \quad (1)$$

where variable $\mathbf{Z} \in \mathbb{R}^{d_t \times d_v}$ represents the attentional matrix.

Researchers found it beneficial to linearly project the queries, keys and values h times with different weight matrix $\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^V$ respectively, concatenate them together, and then multiply with another weight matrix $\mathbf{W}^O \in \mathbb{R}^{hd_v \times d_f}$ to obtain the final output. This is called Multi-Head Attention.

$$\mathrm{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = Concat(\mathbf{Z}_1, \ldots, \mathbf{Z}_h)\mathbf{W}^O \quad (2)$$

To address the crucial problem that transformer has little ability to capture sequential sentence, it is required to add a positional encoding (PE). This means summing a sinusoid function with a large period over the input before feeding it to the first encoder layer. The intuition here is that for any fixed offset k, PE_{pos+k} can be represented as a liner function of PE_{pos}, which provides great convenience for the model to capture the relative position relationship between sequential data.

$$PE(pos, 2i) = \sin(\frac{pos}{10000^{\frac{2i}{d_f}}}) \quad (3)$$

$$PE(pos, 2i+1) = \cos(\frac{pos}{10000^{\frac{2i}{d_f}}}), \quad (4)$$

where pos is the position of each sequential data in input features, i represents the i^{th} dimension of the input embedding of each data.

2.2 Intergrate LSTM with Transformer (ILT)

As the positional encoding for transformer is just a fixed positional representation of input features, initially we aim to replace it by connecting LSTM between CNN and transforer. However in our prilimary experiment, we find that this approach cannot work well. Therefore we propose to use a paralled combination of LSTM and transformer instead of cascaded ones, illustrated in **Fig. 1**. The input features are represented as a sequence of vectors $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$, where N is the length of frames. At first the input features are fed into the CNN, which is composed of serveral conv blocks. In each conv block, there is a convolutional layer, followed by batch normalization, average pooling, and dropout operation. After passing throuth CNN, the output $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_K]$ is fed into two flows to the subsequential models respectively. The first flow is to the LSTM, which is applied to learn long-term dependencies and contextual information by introducing the gating mechanism. We utilize the bidirectional LSTM (bi-LSTM) that the output at the current time step can be both learned from the former and latter state. Also the LSTM output can be regarded as an additional position info. After obtaining the output sequences $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_K]$, we conduct the attention layer as an aggregation function for LSTM. The second flow is to transformer, which has been depicted in Section 2.1.

$$\mathbf{C} = CNN([\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]) \quad (5)$$

$$\overrightarrow{\mathbf{h}_t} = \overrightarrow{LSTM}(\mathbf{h}_{t-1}, \mathbf{c}_t), t \in 1, \cdots, K \quad (6)$$

$$\overleftarrow{\mathbf{h}_t} = \overleftarrow{LSTM}(\mathbf{h}_{t+1}, \mathbf{c}_t), t \in 1, \cdots, K \quad (7)$$

$$\mathbf{h}_t = [\overrightarrow{\mathbf{h}_t}; \overleftarrow{\mathbf{h}_t}], t \in 1, \cdots, K \quad (8)$$

$$\beta_t = \frac{\exp(\mathbf{W}^T \mathbf{h}_t)}{\sum_{\tau=1}^{K} \exp(\mathbf{W}^T \mathbf{h}_\tau)} \quad (9)$$

$$\mathbf{O}_{LSTM} = \sum_{t=1}^{K} \beta_t \mathbf{h}_t \quad (10)$$

$$\mathbf{O}_{Trans} = Transformer(\mathbf{C}), \quad (11)$$

where $\mathbf{W}$ is a trainable weight matrix.

Then two outputs are intergrated together. Let the fusion output as $\mathbf{O}_{final}$, we try to utilize the following three fusion strategies to put the output from LSTM $\mathbf{O}_{LSTM}$ and the output from transformer $\mathbf{O}_{Trans}$ together:

concatnation

$$\mathbf{O}_{final} = [\mathbf{O}_{LSTM}; \mathbf{O}_{Trans}] \quad (12)$$

plus

$$\mathbf{O}_{final} = \alpha \mathbf{O}_{LSTM} + (1-\alpha)\mathbf{O}_{Trans} \quad (13)$$

trainable plus (attention)

$$\mathbf{O}_{final} = \alpha_A \mathbf{O}_{LSTM} + \alpha_B \mathbf{O}_{Trans} \quad (14)$$

$$\alpha_A = \frac{\exp(\mathbf{W}_A)}{\exp(\mathbf{W}_A) + \exp(\mathbf{W}_B)} \quad (15)$$

$$\alpha_B = \frac{\exp(\mathbf{W}_B)}{\exp(\mathbf{W}_A) + \exp(\mathbf{W}_B)}, \quad (16)$$

where $\mathbf{W}_A$ and $\mathbf{W}_B$ are two different trainable weight matrices.

2.3 Cross Attention Transformer (CAT)

Recently in multimodal deep learning, many researches tend to utilize the cross-modal transformer to learn from different modalities (such audios, words, videos, etc) [13], [14], [15]. Inspired by this, we propose to employ this state-of-art model to merge the different information learned from CNN and LSTM. Since CNN makes a good fist at reducing frequency variations (i.e. frequency modeling), LSTM

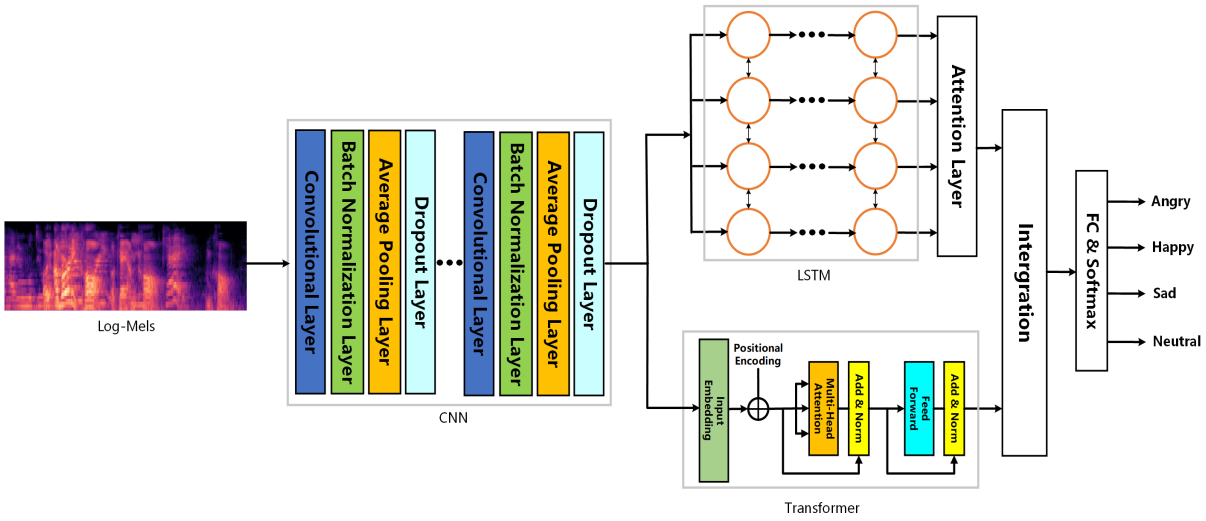


Fig. 1: Overview Structure of the proposed framework intergrate LSTM with transformer (ILT).

is skilled at learning characteristic of data over long periods of time (i.e. temporal modeling) [16], it is reasonable to joint-encoding outputs of them. Depicted in **Fig. 2**, the input features are fed into CNN and LSTM simutanouly. Considering that a tremendous length of input features will affect the function of LSTM and bring more parameters for training, we add an 1D convolutional layer before LSTM simply for reducing the length of input features. Let the output after CNN and LSTM be $\mathbf{H}^{(C)} = \mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_K$ and $\mathbf{H}^{(L)} = \mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_T$ respectively, when we treat $\mathbf{H}^{(C)}$ as queries, $\mathbf{H}^{(L)}$ as keys and values of transformer, we are mapping the information from CNN to LSTM, which is corresponding to the transformer (CtoL) in Fig. 2. Similarily, by considering $\mathbf{H}^{(L)}$ as query matrix, $\mathbf{H}^{(C)}$ as key and value matrix, the transformer is responsible for obtaining interactive information from LSTM to CNN, which is the transformer (LtoC) in Fig. 2. Because in both CtoL and LtoC, the self-attention are a combination with two different networks, they are called cross attention transformer. After that, the output from two cross attention transformers are addeed together and pass through the fully connected layer and softmax layer to obtain the final posterior probabilities of each emotion.

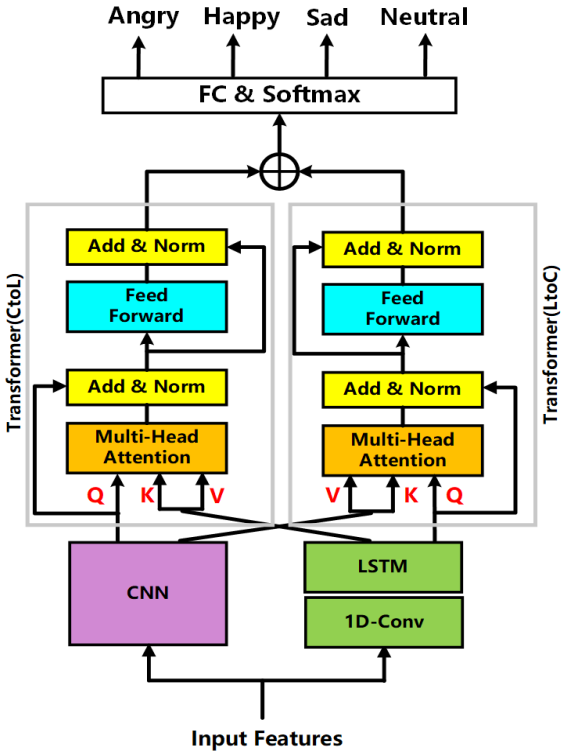


Fig. 2: The block diagram of the proposed cross attention transformer (CAT).

3. Experimental setup

3.1 Dataset

The interactive emotional dyadic motion capture database (IEMOCAP) [17] is used to evaluate our approach. It contains approximately 12 hours of speech. There are 10 actors (5 males and 5 females) to perform 5 dyadic sessions, with 10 emotions (anger, happiness, sadness, neutral, frustration, excitement, fear, surprise, disgust, and other), which have been evaluated by at least three different annotators. Initially, along with previous researches [18], [19], considering the imbalanced label and lack of some emotions, only the following five emotions are extracted: anger, happiness, excitement, sadness, and neutral. Then happiness and excitement are merged into happiness because of the similarity between them. Finally, the amount of labels used in this experiment are 1103, 1636, 1084, 1708 respectively for anger, happiness, sadness and neutral, totally 5531 examples. These dataset is radomly split into training, validation and test sets in the ratio of 0.55:0.25:0.2. We employ the five-fold cross validation to train our model and report the average results.

3.2 Feature extraction and preprossing

Due to the small size of data, which may lead to severe overfitting, we first augment our training and validation data

Table 1: Accruacy comparasion of two baseline systems and our proposed ILT. PE reprents positional encoding in transformer, WA is the weighted accuracy and UA is the unweighted accuracy.

Net Structure	Fusion Mode	PE	WA(%)	UA(%)
CNN-transformer (baseline1)	-	False	65.76	67.17
		True	**66.91**	**67.60**
CNN-LSTM-transformer (baseline2)	-	False	65.60	67.36
		True	66.47	67.52
ILT	concatenation	False	66.05	66.82
		True	66.21	67.40
	plus	False	67.26	68.38
		True	66.50	67.36
	trainable plus (attention)	False	**67.56**	**68.63**
		True	67.12	67.89

Table 2: Performance of different variants of our proposed CAT. The representation CNN+LSTM means that the input feature go through CNN and LSTM simultaneously and then two outputs are added together.

System	WA(%)	UA(%)
CAT	**68.10**	**69.25**
CNN-Transformer (baseline)	66.91	67.60
LSTM-Transformer	55.19	56.40
CNN+LSTM	66.90	67.98
CAT (CNNtoLSTM only)	66.98	67.83
CAT (LSTMtoCNN only)	65.57	66.49

by adding high signal-noise ratio (SNR) white noise. Then the speech signal in IEMOCAP is sampled at 16kHz and zero-padding or cutting is applied for the utterances whose duration is less or more than 7.5 seconds. We use the modified utterances to calculate the log-mel spectrograms (log-Mels) with the window size 25 ms and the frame size 10 ms as input feature to our model. The log-Mels are extracted by *librosa* toolkit and the number of filter banks is set to 40, therefore we can obtain a matrix of size 40×750. We use the *pytorch* toolkit to implement all our models. We choose cross-entropy loss as loss function, and Adam with a learning rate of 0.0001 and 1^{st}&2^{nd} momentum of 0.9&0.99 as optimizer. The batch size and training epoch are set to 40 and 100. As evaluation criteria, we employ three metrics in this experiment: confusion matrix, weighted accuracy (WA), and unweighted accuracy (UA). Confusion matrix depicts the classification situation of each label. Weighted accuracy is the classification accuracy for the entire data set, and unweighted accuracy is an average of the classification accuracy for each emotion [7]. Here, all hyper parameters of our model were finetuned to maximize the WA.

3.3 Baseline systems

For the baseline model, we test the CNN connected with transformer sequentially (CNN-Transformer), which is regarded as Fig. 1 with LSTM removed. Moreover, we insert LSTM into the middle of our baseline system as another baseline system (CNN-LSTM-Transformer).

4. Experiments and Analysis

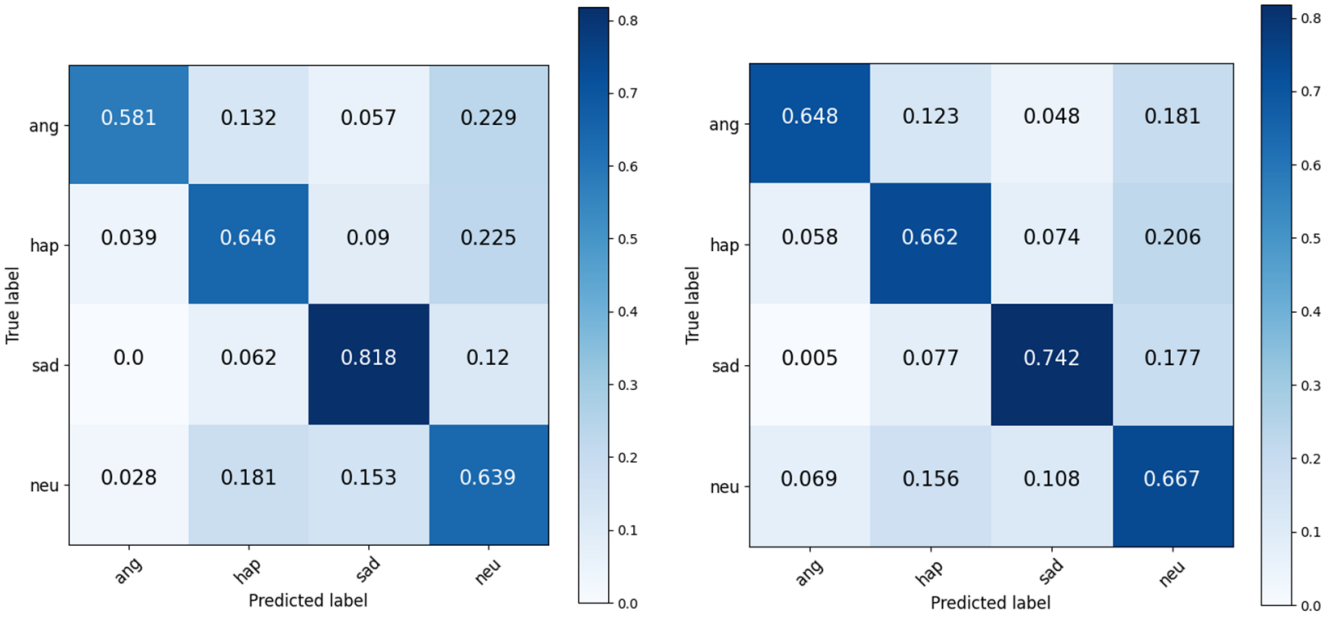
4.1 Experiment on ILT

Table 1 summarizes the experiment results of ILT and our baselines in different conditions. Far from obtaining better performance, simply connecting LSTM between CNN and transformer even brings a little bit worse accuracy. We infer that because too many pooling layers in CNN lead to less temporal information, LSTM cannot learn anything from a relatively short interval. Among the three fusion ways tested (concatenation, plus, and trainable plus(attention)) in ILT, the last one shows the best result among them, which exceeds the baseline models with +0.65% and +1.03% absolute improvements in WA and UA, respectively. This reveals that, by putting LSTM in the same position of transformer and intergrating them with the attention mechanism, we can focus on more emotional part of input utterances. The attention factor learned by training data is 0.65 for LSTM and 0.35 for transformer, indicating that LSTM contributes more in this integration mode.

In addition, there also exists an interesting phenomenon. In baselines/ILT with concatenation fusion, it appears that positional encoding can make performance increase to some extent. However, for ILT with plus and trainable plus integration modes, a better performance is obtained without PE. We suppose that since the two models are placed in parallel, the function of positional encoding is duplicated with the function of LSTM.

4.2 Experiment on CAT

Depicted in **Table 2**, the Cross Attention Transformer achieves the best performance, which has an increase of +1.19% on WA and +1.65% on UA, indicating the effectiveness of joint learning of the information from the two basic deep learning models. Besides, we explore the ablation study to see different contributions to accuracy with different part of the CAT. We first use the output of CNN simply added to the result from LSTM, which shows the transformer part can bring 1.20% WA improvement and 1.27% UA improvement. Then by setting the weight of transformer (LtoC) or the weight of transformer (CtoL) in Fig. 2 as 0, we get the CAT (CNNtoLSTM only) and CAT (LSTMtoCNN only) respectively. Comparing with passing through the transformer directely, mapping into a different model indeed can improve the performance. By adding these two variant CAT together, a better accuracy can be observed. Moreover, mapping from CNN to LSTM leads to a better WA and UA than from LSTM to CNN, inspiring



(a) CNN-Transformer (b) Cross Attention Transformer

Fig. 3: Confusion matrix of baseline system and proposal CAT. (*ang* anger state; *hap* happiness; *sad* sadness; *neu* neutral)

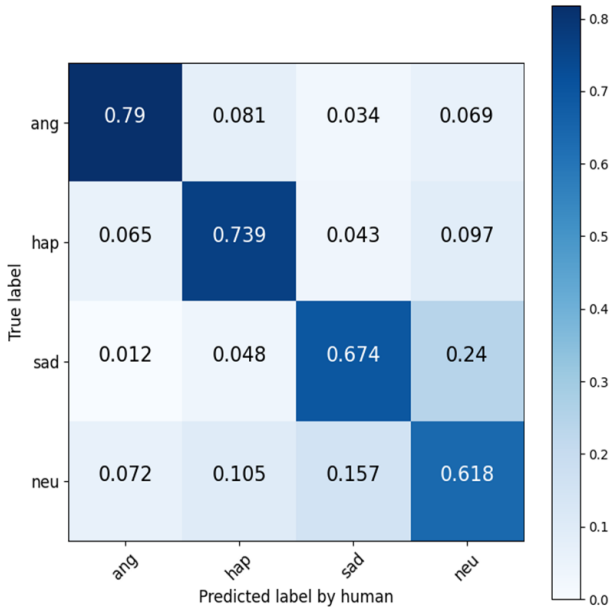


Fig. 4: Human's confusion matrix [20] of the IEMOCAP dataset. It is obtained by non-expert participants assessing utterences of the dataset in a "humanized machine learning" style, which means that for each utterence it is possible to see the correct answer after one gives his own answer.

us to finetune the weight of the CAT to further improve the performance as a future work.

Meanwhile, we report the confusion matrix from our proposal CAT and our baseline model, shown in **Fig. 3**. We observe that sadness obtains the highest recognition rate, and happiness obtains the lowest recognition confused with other classes. This seems plausible because the neutral state is located in the center of the arousal-valence space of emotion, which makes the discrimination from other classes difficult. Moreover, the error rate from anger to happiness is quite higher than others, which can be reasonable considering the dimensional emotional model: both anger and happiness have a high arousal level. Hence, the system's frequent confusion is due to the fact that valence is harder to predict than activation.

Different from our baseline systems, we can see the probabilities of misclassifying anger and happiness as neutral and misclassifying neutral as anger and happiness are reduced so that anger, happiness and neutral can achieve a higher probability of being correctly predicted by using the CAT. Although there is less sadness being truly classified as themselves, the error pattern is more balanced than the baseline system. These results indicate that the cross attention transformer network provides an improvement in the recognition accuracy of most emotions. Comparing with human error structure depicted in **Fig. 4**, it can be observed that the error pattern of cross attention transformer is closer to human ones than baseline CNN-Transformer. However, while it is hard to confuse angry with happiness and neutral and confuse happiness with neutral for humans, the error rate is quite high in our model. This indicates that we need to further focus on the reason for such bias, i.e. the explainable of network structure.

5. Conclusion

This paper proposes two SER frameworks ILT and CAT to aggregate transformer with other deep learning models. In ILT, we combine LSTM with transformer in a parallel style; in CAT, we utilize the cross attention transformer to learn the interaction between the information that CNN and LSTM carries. Experiments on the IEMOCAP dataset demonstrate the effectiveness of our proposed algorithms. For future work, we will evaluate the performance using other input features, such as prosodic features, voice quality

features, etc. Furthermore, we plan to focus on the recognized text data by real-time automatic speech recognition (ASR) for multimodal learning for SER.

References

[1] El Ayadi, M., Kamel, M. S. and Karray, F.: Survey on speech emotion recognition: Features, classification schemes, and databases, *Pattern recognition*, Vol. 44, No. 3, pp. 572–587 (2011).

[2] Shen, G., Lai, R., Chen, R., Zhang, Y., Zhang, K., Han, Q. and Song, H.: WISE: Word-Level Interaction-Based Multimodal Fusion for Speech Emotion Recognition., *INTERSPEECH*, pp. 369–373 (2020).

[3] Han, K., Yu, D. and Tashev, I.: Speech emotion recognition using deep neural network and extreme learning machine, *Fifteenth annual conference of the international speech communication association* (2014).

[4] Lee, C. M., Yildirim, S., Bulut, M., Kazemzadeh, A., Busso, C., Deng, Z., Lee, S. and Narayanan, S.: Emotion recognition based on phoneme classes, *Eighth International Conference on Spoken Language Processing* (2004).

[5] Kwon, O.-W., Chan, K., Hao, J. and Lee, T.-W.: Emotion recognition by speech signals, *Eighth European Conference on Speech Communication and Technology* (2003).

[6] Mao, Q., Dong, M., Huang, Z. and Zhan, Y.: Learning salient features for speech emotion recognition using convolutional neural networks, *IEEE transactions on multimedia*, Vol. 16, No. 8, pp. 2203–2213 (2014).

[7] Lee, J. and Tashev, I.: High-level feature representation using recurrent neural network for speech emotion recognition, *Sixteenth annual conference of the international speech communication association* (2015).

[8] Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B. and Zafeiriou, S.: Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network, *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, pp. 5200–5204 (2016).

[9] Chen, M., He, X., Yang, J. and Zhang, H.: 3-D convolutional recurrent neural networks with attention model for speech emotion recognition, *IEEE Signal Processing Letters*, Vol. 25, No. 10, pp. 1440–1444 (2018).

[10] Tarantino, L., Garner, P. N. and Lazaridis, A.: Self-attention for speech emotion recognition., *Interspeech*, pp. 2578–2582 (2019).

[11] Nediyanchath, A., Paramasivam, P. and Yenigalla, P.: Multi-head attention for speech emotion recognition with auxiliary learning of gender recognition, *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, pp. 7179–7183 (2020).

[12] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. and Polosukhin, I.: Attention is all you need, *Advances in neural information processing systems*, pp. 5998–6008 (2017).

[13] Pan, Z., Luo, Z., Yang, J. and Li, H.: Multi-Modal Attention for Speech Emotion Recognition, *Interspeech 2020*, ISCA, ISCA, pp. 364–368 (online), DOI: 10.21437/Interspeech.2020-1653 (25-29 October 2020).

[14] Delbrouck, J.-B., Tits, N., Brousmiche, M. and Dupont, S.: A Transformer-based joint-encoding for Emotion Recognition and Sentiment Analysis, *arXiv preprint arXiv:2006.15955* (2020).

[15] Khare, A., Parthasarathy, S. and Sundaram, S.: Multi-modal embeddings using multi-task learning for emotion recognition, *arXiv preprint arXiv:2009.05019* (2020).

[16] Sainath, T. N., Vinyals, O., Senior, A. and Sak, H.: Convolutional, long short-term memory, fully connected deep neural networks, *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, pp. 4580–4584 (2015).

[17] Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S. and Narayanan, S. S.: IEMOCAP: Interactive emotional dyadic motion capture database, *Language resources and evaluation*, Vol. 42, No. 4, pp. 335–359 (2008).

[18] Li, J.-L. and Lee, C.-C.: Attentive to Individual: A Multimodal Emotion Recognition Network with Personalized Attention Profile., *Interspeech*, pp. 211–215 (2019).

[19] Zhou, H. and Liu, K.: Speech Emotion Recognition with Discriminative Feature Learning, *Interspeech 2020*, ISCA, ISCA, pp. 4094–4097 (online), DOI: 10.21437/Interspeech.2020-2237 (2020).

[20] Chernykh, V., Sterling, G. and Prihodko, P.: Emotion Recognition From Speech With Recurrent Neural Networks, *ArXiv*, Vol. abs/1701.08071 (2017).