Multi-Task Learning Framework for Emotion Recognition in-the-wild

Tenggan Zhang[1,*], Chuanhe Liu[2,*], Xiaolong Liu[2,*], Yuchen Liu[1], Liyu Meng[2], Lei Sun[1], Wenqiang Jiang[2], Fengyuan Zhang[1], Jinming Zhao[3], and Qin Jin[1,†]

[1] School of Information, Renmin University of China
{zhangtenggan,zbxytx,qjin}@ruc.edu.cn
{sunlei.ruc,zhangfy.ruc}@gmail.com
[2] Beijing Seek Truth Data Technology Co.,Ltd.
{liuchuanhe,liuxiaolong,mengliyu,jiangwenqiang}@situdata.com
[3] Qiyuan Lab, Beijing, China
zhaojinming@qiyuanlab.com

Abstract. This paper presents our system for the Multi-Task Learning (MTL) Challenge in the 4th Affective Behavior Analysis in-the-wild (ABAW) competition. We explore the research problems of this challenge from three aspects: 1) For obtaining efficient and robust visual feature representations, we propose MAE-based unsupervised representation learning and IResNet/DenseNet-based supervised representation learning methods; 2) Considering the importance of temporal information in videos, we explore three types of sequential encoders to capture the temporal information, including the encoder based on transformer, the encoder based on LSTM, and the encoder based on GRU; 3) For modeling the correlation between these different tasks (i.e., valence, arousal, expression, and AU) for multi-task affective analysis, we first explore the dependency between these different tasks and propose three multi-task learning frameworks to model the correlations effectively. Our system achieves the performance of 1.7607 on the validation dataset and 1.4361 on the test dataset, ranking first in the MTL Challenge. The code is available at https://github.com/AIM3-RUC/ABAW4.

1 Introduction

Affective computing aims to develop technologies to empower machines with the capability of observing, interpreting, and generating emotions just like humans do [30]. There has emerged a wide range of application scenarios of affective computing, including health research, society analysis, and other interaction scenarios. More and more people are interested in affective computing due to the significant improvement of machine learning technology performance and the growing attention to the mental health field. There are lots of datasets to support the

---

[*] Equal Contribution.

[†] Corresponding Author.

research of affective computing, including Aff-wild [16], Aff-wild2 [19], and s-Aff-Wild2[12,13,17,22,15,21,20,18,14,33,16]. The advancement of multi-task learning algorithms [28] has also boosted performance via exploring supervision from different tasks.

Our system for the Multi-Task Learning (MTL) Challenge contains four key components. 1) We explore several unsupervised (MAE-based) and supervised (IResNet/DenseNet-based) visual feature representation learning methods for learning effective and robust visual representations; 2) We utilize three types of temporal encoders, including GRU [4], LSTM [29] and Transformer [31], to capture the sequential information in videos; 3) We employ multi-task frameworks to predict the valence, arousal, expression and AU values. Specifically, we investigate three different strategies for multi-task learning, namely $\underline{S}$hare $\underline{E}$ncoder (SE), $\underline{S}$hare $\underline{B}$ottom of $\underline{E}$ncoder (SBE) and $\underline{S}$hare $\underline{B}$ottom of $\underline{E}$ncoder with $\underline{H}$idden $\underline{S}$tates $\underline{F}$eedback (SBE-HSF); 4) Finally, we adopt ensemble strategies and cross-validation to further enhance the predictions, and we get the performance of 1.7607 on the validation dataset and 1.4361 on the test dataset, ranking first in the MTL Challenge.

2 Related Works

There are lots of solutions proposed for former ABAW competitions. We investigate some studies for valence and arousal prediction, facial expression classification and facial action unit detection, which are based on deep learning methods.

For valence and arousal prediction, [25] proposes a novel architecture to fuse temporal-aware multimodal features and an ensemble method to further enhance performance of regression models. [34] proposes a model for continuous emotion prediction using a cross-modal co-attention mechanism with three modalities (i.e., visual, audio and linguistic information). [27] combines local attention with GRU and uses multimodal features to enhance the performance. For expression classification, facing the problem that the changes of features for expression are difficult to be processed by one attention module, [32] proposes a novel attention mechanism to capture local and semantic features. [35] utilizes multimodal features, including visual, audio and text to build a transformer-based framework for expression classification and AU detection. For facial action unit detection, [8] utilizes a multi-task approach with a center contrastive loss and ROI attention module to learn the correlations of facial action units. [9] proposes a model-level ensemble method to achieve comparable results. [5] introduces a semantic correspondence convolution module to capture the relations of AU in a heat map regression framework dynamically.

3 Method

Given an image sequence consisting of $\{F_1, F_2, ..., F_n\}$ from video X, the goal of the MTL challenge is to produce four types of emotion predictions for each frame, including the label y^v for valence, the label y^a for arousal, the label y^e
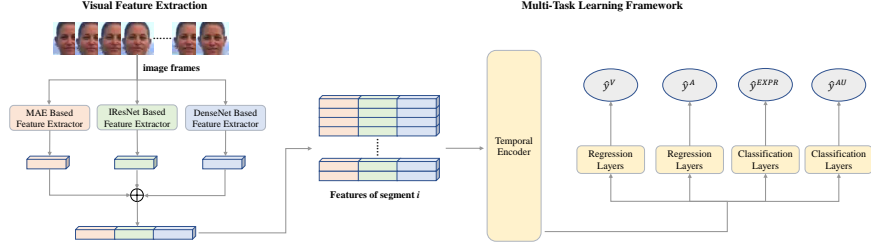
Fig. 1. The pipeline of our method for the challenge.

for expression, and the labels $\{y^{AU1}, y^{AU2}, ..., y^{AU26}\}$ for 12 AUs. Please note that only some sampled frames in a video are annotated in the training data, and the four types of annotations may be partially missing for an image frame. Our pipeline for the challenge is shown in figure 1.

3.1 Features

MAE-based Features The features of the first type are extracted by MAE [6] models[†] which use C-MS-Celeb [10] and EmotionNet [3] datasets at the pre-training stage. The first model is pre-trained on the C-MS-Celeb dataset and fine-tuned on different downstream tasks, including expression classification on the s-Aff-Wild2 dataset, AU classification task on the s-Aff-Wild2 dataset, expression classification on the AffectNet [26] dataset and expression classification on the dataset combining FER+ [2] and AffectNet [26] datasets. As for the second model, we first use the EmotionNet dataset to pre-train the MAE model with the reconstruction task, and then use the AffectNet [26] dataset to fine-tune the model further.

IResNet-based Features The features of the second type are extracted by IResNet100 models. The models are pre-trained in two different settings. As for the first setting, we use FER+ [2], RAF-DB [24,23], and AffectNet [26] datasets to pre-train the model. Specifically, the faces are aligned by keypoints and the input size is resized into 112x112 before pre-training. As for the second setting, we use the Glint360K [1] dataset to pre-train the model, and then use an FAU dataset with commercial authorization to train this model further.

DenseNet-based Features The features of the third type are extracted by a DenseNet [7] model. The pre-training stage uses FER+ and AffectNet datasets, and we also try to fine-tune the pre-trained model on the s-Aff-Wild2 dataset, including the expression classification task and AU classification task.

[†] https://github.com/pengzhiliang/MAE-pytorch

3.2   Temporal Encoder

Because the GPU memory is limited, the annotated frames are firstly split into segments. If the length of the split segment is l and n available annotated frames are contained in the video, we can split the frames into $[n/l] + 1$ segments, which means annotated frames $\{F_{(i-1)*l+1}, ..., F_{(i-1)*l+l}\}$ are contained in the i-th segment. After getting the visual features from the i-th segment f_i^m, three different temporal encoders including GRU, LSTM and transformer encoder are used to capture the temporal information in the video.

GRU-based Temporal Encoder We use a Gate Recurrent Unit Network (GRU) to encode the temporal information of the image sequence. Segment s_i means the i-th segment, and f_i^m means the input of GRU is the visual features for s_i. Furthermore, the hidden states of the last layer are fed from the previous segment s_{i-1} into the GRU to utilize the information from the last segment.

$$g_i, h_i = \mathrm{GRU}(f_i^m, h_{i-1}) \tag{1}$$

where h_i denotes the hidden states at the end of s_i. h_0 is initialized to be zeros. To ensure that the last frame of s_{i-1} and the first frame of segment s_i are consecutive frames, there is no overlap between the two adjacent segments.

LSTM-based Temporal Encoder We employ a Long Short-Term Memory Network (LSTM) to model the sequential dependencies in the video. It can be formulated as follows:

$$g_i, h_i = \mathrm{LSTM}(f_i^m, h_{i-1}) \tag{2}$$

The symbols have the same meaning as in the GRU part.

Transformer-based Temporal Encoder We utilize a transformer encoder to model the temporal information in the video segment as well, which can be formulated as follows:

$$g_i = \mathrm{TRMEncoder}(f_i^m) \tag{3}$$

Unlike GRU and LSTM, the transformer encoder just models the context in a single segment and ignores the dependencies of frames between segments.

3.3   Single Task Loss Function

We first introduce the loss function for each task in this subsection.

Valence and Arousal estimation task:

We utilize the Mean Squared Error (MSE) loss which can be formulated as

$$L^V = L^A = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i) \tag{4}$$

where N denotes the number of frames in each batch, $\hat{y}_i$ and y_i denote the prediction and label of valence or arousal in each batch respectively.

Expression Classification task:

We utilize the Cross Entropy (CE) loss which can be formulated as

$$L^{EXPR} = -\sum_{i=1}^{N}\sum_{j=1}^{C}y_{ij}log(\hat{y}_{ij}) \tag{5}$$

where C is equal to 8 which denotes the total classification number of all expression, $\hat{y}_{ij}$ and y_{ij} denote the prediction and label of expression in each batch.

AU Classification task:

We utilize Binary Cross Entropy (BCE) loss which can be formulated as

$$L^{AU} = \sum_{i=1}^{N}\sum_{j=1}^{M}(-(y_{ij}log(\hat{y}_{ij}) + (1 - y_{ij})log(1 - \hat{y}_{ij})))) \tag{6}$$

where M is equal to 12 which denotes the total number of facial action units, $\hat{y}_{ij}$ and y_{ij} denote the logits and label of facial action units in each batch.

3.4   Multi-Task Learning Framework

As we mentioned above, the overall estimation objectives can be divided into four tasks, including the estimation of valence, arousal, expression and action units on expressive facial images. These four objectives focus on different information on the facial images, where the essential information about one task may be helpful to the modeling of some other tasks.

The dependencies between tasks are manifested mainly in two aspects: First, the low-level representations are common for some tasks and they can be shared to benefit each task. Second, some high-level task-specific information of one task could be important features for other tasks. For example, since the definition of expressions depends on facial action units to some extent, the high-level features in the AU detection task can help the estimation of expression.

In order to make use of such dependencies between different tasks, we make some efforts on the multi-task learning frameworks instead of the single-task models. Specifically, we propose three multi-task learning frameworks, as illustrated in Figure 2.

Share Encoder We propose the Share Encoder (SE) framework as the baseline, which is commonly used in the field of multi-task learning. In the SE framework, the temporal encoder is directly shared between different tasks, while each task
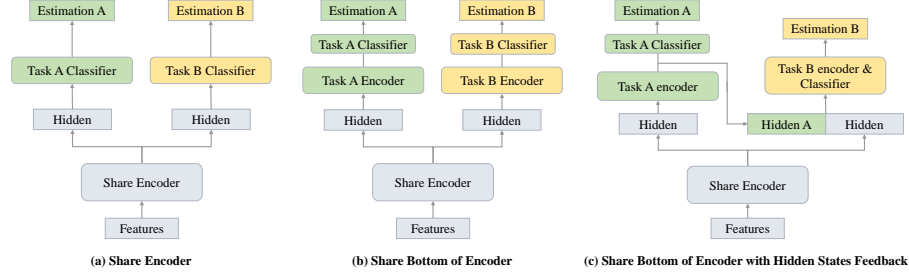
Fig. 2. Our proposed multi-task learning frameworks.

retains task-specific regression or classification layers. The structure of the SE framework is shown in Figure 2(a), which can be formulated as follows:

$$g_i = \text{TE}(f_i^m) \tag{7}$$

$$\hat{y}_i^t = W_p^t g_i + b_p^t, \ t \in T \tag{8}$$

where TE denotes the temporal encoder, T denotes the collection of chosen tasks in the multi-task learning framework, t denotes a specific task in {v, a, e, au}, y_i^t denotes the predictions of task t of segment s_i, W_p^t and b_p^t denote the parameters to be optimized.

Share Bottom of Encoder Under the assumption that the bottom layers of the encoder capture more basic information in facial images while the top layers encode more task-specific features, we propose to only share the bottom layers of the temporal encoder between different tasks. The structure of the $\underline{S}$hare $\underline{B}$ottom of $\underline{E}$ncoder (SBE) framework is shown in Figure 2(b), which can be formulated as follows:

$$g_i = \text{TE}(f_i^m) \tag{9}$$

$$g_i^t = \text{TE}^t(g_i), \ t \in T \tag{10}$$

$$\hat{y}_i^t = W_p^t g_i^t + b_p^t, \ t \in T \tag{11}$$

where TE denotes the temporal encoder, t denotes a specific task and T denotes the collection of chosen tasks, TE^t denotes the task-specif temporal encoder of task t, y_i^t denotes the predictions of task t of segment s_i, W_p^t and b_p^t denote the parameters to be optimized.

Share Bottom of Encoder with Hidden States Feedback Although the proposed SBE framework has captured the low-level shared information between different tasks, it might ignore the high-level task-specific dependencies of tasks. In order to model such high-level dependencies, we propose the $\underline{S}$hare $\underline{B}$ottom of $\underline{E}$ncoder with $\underline{H}$idden $\underline{S}$tates $\underline{F}$eedback (SBE-HSF) framework, as illustrated

in Figure 2(c). In the SBE-HSF framework, all the tasks share the bottom layers of the temporal encoder and retain task-specific top layers, as in the SBE framework.

Afterward, considering that the information of one task could benefit the estimation of another task, we feed the last hidden states of the temporal encoder of the source task into the temporal encoder of the target task as features. It can be formulated as follows:

$$g_i = \text{TE}(f_i^m) \tag{12}$$

$$g_i^t = \text{TE}^t(g_i), \ t \in T \setminus \{t^{tgt}\} \tag{13}$$

$$g_i^{tgt} = \text{TE}^{tgt}(\text{Concat}(g_i, g_i^{src})) \tag{14}$$

$$\hat{y}_i^t = W_p^t g_i^t + b_p^t, \ t \in T \tag{15}$$

where TE denotes the temporal encoder, t denotes a specific task and T denotes the collection of chosen tasks, src and tgt denote the source and target task of the feedback structure, respectively, TE^t denotes the task-specif temporal encoder of task t, y_i^t denotes the predictions of task t of segment s_i, W_p^t and b_p^t denote the parameters to be optimized. In addition, in the backward propagation stage, the gradient of g_i^{src} is detached.

Multi-Task Loss Function In the multi-task learning framework, we utilize the multi-task loss function to optimize the model, which combines the loss functions of all tasks chosen for multi-task learning:

$$L = \sum_{t \in T} \alpha^t L^t \tag{16}$$

where t denotes a specific task and T denotes the collection of chosen tasks, L^t denotes the loss function of task t, which is mentioned above, α^t denotes the weight of L^t which is a hyper-parameter.

4 Experiments

4.1 Dataset

The Multi-Task Learning (MTL) Challenge in the fourth ABAW competition[12] uses the s-Aff-Wild2 dataset as the competition corpora, which is the static version of the Aff-Wild2[19] database and contains some specific frames of the Aff-Wild2 database.

As for feature extractors, the FER+[2], RAF-DB[24,23], AffectNet[26], C-MS-Celeb[10] and EmotionNet[3] datasets are used for pre-training. In addition, an authorized commercial FAU dataset is also used to pre-train the visual feature extractor. It contains 7K images in 15 face action unit categories(AU1, AU2, AU4, AU5, AU6, AU7, AU9, AU10, AU11, AU12, AU15, AU17, AU20, AU24, and AU26).

4.2   Experiment Setup

As for the training setting, we use Nvidia GeForce GTX 1080 Ti GPUs to train the models, and the optimizer is the Adam[11]. The number of epochs is 50, the dropout rate of the temporal encoder and the FC layers is 0.3, the learning rate is 0.00005, the length of video segments is 250 for arousal and 64 for valence, expression and AU, and the batch size is 8.

As for the model architecture, the dimension of the feed-forward layers or the size of hidden states is 1024, the number of FC layers is 3 and the sizes of hidden states are {512, 256}. Specially, the encoder of transformer has 4 layers and 4 attention heads.

As for the smooth strategy, we search for the best window of valence and arousal for each result based on the performance on the validation set. Most window lengths are 5 and 10.

4.3   Overall Results on the validation set

In this section, we will demonstrate the overall experimental results of our proposed method for the valence, arousal, expression and action unit estimation tasks. Specifically, the experimental results are divided into three parts, including the single-task results, the exploration of multi-task dependencies and the results of multi-task learning frameworks. We report the average performance of 3 runs with different random seeds.

Single-Task Results In order to verify the performance of our proposed model without utilizing the multi-task dependencies, we conduct several single-task experiments. The results are demonstrated in Table 1.

Table 1. The performance of our proposed method on the validation set for each single task.

Model	Task	Features	Performance
Transformer	Valence	MAE,ires100,fau,DenseNet	0.6414
Transformer	Arousal	MAE,ires100,fau,DenseNet	0.6053
Transformer	EXPR	MAE,ires100,fau,DenseNet	0.4310
Transformer	AU	MAE,ires100,fau,DenseNet	0.4994

Results of Multi-Task Learning Frameworks We try different task combinations and apply the best task combination to the multi-task learning frameworks for each task. As a result, we find the best task combinations as follows: {V, EXPR} for valence, {V, A, AU} for arousal, {V, EXPR} for expression and {V, AU} for action unit. The experimental results of our proposed multi-task

learning frameworks and the comparison with single-task models are shown in Table 2. Specifically, the combination of features is the same as that in single-task settings, and the set of tasks chosen for the multi-task learning frameworks is based on the multi-task dependencies, which have been explored above.

As is shown in the table, first, all of our proposed multi-task frameworks outperform the single-task models on valence, expression and action unit estimation tasks. On the arousal estimation task, only the SE framework performs inferior to the single-task model and the other two frameworks outperform it. These results show that our proposed multi-task learning frameworks can improve performance and surpass the single-task models.

Moreover, the two proposed frameworks, SBE and SBE-HSF, show the advanced performance, where the former is an improvement on the SE framework and the latter is an improvement on the former. The SBE framework outperforms the SE frameworks, and the SBE-HSF framework outperforms the SBE framework on arousal, expression and action unit estimation tasks. It indicates our proposed multi-task learning framework can effectively improve performance.

Table 2. The performance of our proposed multi-task learning frameworks on the validation set.

	Valence		Arousal		EXPR		AU	
	Tasks	CCC	Tasks	CCC	Tasks	F1	Tasks	F1
Single Task	V	0.6414	A	0.6053	EXPR	0.4310	AU	0.4994
SE	V, EXPR	0.6529	V, A, AU	0.5989	V, EXPR	0.4406	V, AU	0.5084
SBE	V, EXPR	**0.6558**	V, A, AU	0.6091	V, EXPR	0.4460	V, AU	0.5107
SBE-HSF	Src: V Tgt: EXPR	0.6535	Src: V,AU Tgt: A	**0.6138**	Src: EXPR Tgt: V	**0.4543**	Src: V Tgt: AU	**0.5138**

4.4 Model Ensemble

Table 3. The single model results and ensemble result on the validation set for the valence prediction task.

Model	Features	Loss	Valence-CCC
Transformer	MAE,ires100,fau,DenseNet	V,EXPR	0.6778
LSTM	MAE,ires100,fau,DenseNet	V	0.6734
Ensemble			**0.7101**

We evaluate the proposed methods for the valence and arousal prediction task on the validation set. As is shown in the Table 3 and Table 4, the best performance for valence is achieved by transformer-based model, and the best performance for arousal is achieved by LSTM-based model and the GRU-based

Table 4. The single model results and ensemble result on the validation set for the arousal prediction task.

Model	Features	Loss	Arousal-CCC
LSTM	MAE,ires100,fau,DenseNet	V,A,AU	0.6384
LSTM	MAE,ires100,fau,DenseNet	V,A,AU	0.6354
GRU	MAE,ires100,fau,DenseNet	V,A,AU	0.6292
GRU	MAE,ires100,DenseNet	V,A,AU	0.6244
Ensemble			**0.6604**

model also achieves competitive performance for arousal. Furthermore, the ensemble result can achieve 0.7101 on valence and 0.6604 on arousal, which shows that the results of different models benefit each other.

Table 5. The single model results and ensemble result on the validation set for the EXPR prediction task.

Model	Features	Loss	EXPR-F1
Transformer	MAE,ires100,fau,DenseNet	V,EXPR	0.4739
Transformer	MAE,ires100,fau,DenseNet	V,EXPR	0.4796
Ensemble			**0.5090**

Table 5 shows the results on the validation set for expression prediction. As is shown in the table, the transformer-based model can achieve the best performance for expression and the ensemble result can achieve 0.5090 on the validation set. We use the vote strategy for expression ensemble, and we choose the class with the least number in the training set when the number of classes with the most votes is more than one.

Table 6. The single model results and ensemble result on the validation set for the AU prediction task.

Model	Features	Loss	Threshold	AU-F1
Transformer	MAE,ires100,fau,DenseNet	V,AU	0.5	0.5217
Transformer	MAE,ires100,fau,DenseNet	V,AU	0.5	0.5213
Transformer	MAE,ires100,fau,DenseNet	V,A,AU	0.5	0.5262
LSTM	MAE,ires100,fau,DenseNet	V,AU	0.5	0.5246
LSTM	MAE,ires100,fau,DenseNet	V,AU	0.5	0.5228
LSTM	MAE,ires100,DenseNet	V,AU	0.5	0.5227
Ensemble			0.5	0.5486
			variable	**0.5664**

Table 6 shows the results on the validation set for AU prediction. As is shown in the table, the transformer-based model and LSTM-based model can achieve

excellent performance for AU and the ensemble result can achieve 0.5664 on the validation set. We try two ensemble types for AU. The first is the vote strategy, and we predict 1 when 0 and 1 have the same number of votes. The second is averaging the probabilities from different models for each AU and search the best threshold based on the performance on the validation set for the final prediction.

Table 7. The results of the 6-fold cross-validation experiments. The first five folds are from the training set. Fold 6 means the official validation set.

	Valence	Arousal	EXPR	AU	P_{MTL}
Fold 1	0.6742	0.6663	0.4013	0.5558	1.6274
Fold 2	0.5681	0.6597	0.3673	0.5496	1.5306
Fold 3	0.6784	0.6536	0.3327	0.5977	1.5963
Fold 4	0.6706	0.6169	0.3851	0.5886	1.6275
Fold 5	0.7015	0.6707	0.4389	0.5409	1.6658
Fold 6	0.6672	0.6290	0.4156	0.5149	1.5786
Average	0.6600	0.6494	0.3901	0.5579	1.6027

6-fold cross-validation is also conducted for avoiding overfitting on the validation set. After analyzing the dataset distribution, we find the training set is about five times the size of the validation set, so we divide the training set into five folds, and each fold has approximately the same video number and frame number as the validation set. The validation set can be seen as the 6th fold. The feature set {MAE, ires100, fau, DenseNet} and the transformer-based structure are chosen for valence, expression and AU prediction. The feature set {MAE, ires100, fau, DenseNet} and the LSTM-based structure are chosen for arousal prediction. Note that we have features fine-tuned on the s-Aff-Wild2 dataset, which may interfere with the results of the corresponding task, so we remove the features fine-tuned on the s-Aff-Wild2 dataset for corresponding 6-fold cross-validation experiments. The results are shown in Table 7.

4.5   Results on the test set

We will briefly explain our submission strategies and show the test results of them, which are demonstrated in table 8.

We only use a simple strategy for the 1st and 2nd submissions, which means we train models on the training set using the features we extract, and choose the models of best epochs for different tasks. Specifically, only several models are chosen to ensemble to prevent lowering the result and we use vote strategy for expression and AU ensemble for the 1st submission. Furthermore, more models are used to ensemble and we choose the best ensemble strategy to pursue the highest performance on the validation set for 2nd submission.

Further, we use two carefully designed strategies for the 3rd and 5th submissions, including Train-Val-Mix and 6-Fold. Specifically, the Train-Val-Mix

Table 8. The results of different submission strategies on the test set.

Submit	Strategy	P_{MTL}
1	Ensemble 1	1.4105
2	Ensemble 2	1.3189
3	Train-Val-Mix	1.3717
4	Ensemble 3	1.3453
5	6-Fold	**1.4361**

strategy means the training and validation set are mixed up for training. In this case, we don't have meaningful validation performance to choose models, so we analyze the distribution of the best epochs for previous experiments under the same parameter setting, and empirically choose the models. The selected epoch interval is from 10 to 19 for valence, from 15 to 19 for arousal, from 15 to 24 for expression, and from 30 to 34 for AU. Further, all these models are used to ensemble for better results. As for the 6-Fold strategy, five folds are used for the training stage and the rest fold is used for validation each time. Since we get six models under six settings, all six models are used to ensemble to get the final results. Additionally, the 4th submission is a combination of 2nd and 3rd submissions.

As is shown in the Table8, the 6-Fold strategy achieves the best performance on the test set, and the 1st ensemble strategy also achieves competitive performance.

5   Conclusion

In this paper, we introduce our framework for the Multi-Task Learning (MTL) Challenge of the 4th Affective Behavior Analysis in-the-wild (ABAW) competition. Our method utilizes visual information and uses three different sequential models to capture the sequential information. And we also explore three multi-task framework strategies using the relations of different tasks. In addition, the smooth method and ensemble strategies are used to get better performance. Our method achieves the performance of 1.7607 on the validation dataset and 1.4361 on the test dataset, ranking first in the MTL Challenge.

6   Acknowledgement

References

1. An, X., Zhu, X., Gao, Y., Xiao, Y., Zhao, Y., Feng, Z., Wu, L., Qin, B., Zhang, M., Zhang, D., et al.: Partial fc: Training 10 million identities on a single machine.

In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1445–1449 (2021)

2. Barsoum, E., Zhang, C., Canton Ferrer, C., Zhang, Z.: Training deep networks for facial expression recognition with crowd-sourced label distribution. In: ACM International Conference on Multimodal Interaction (ICMI) (2016)

3. Benitez-Quiroz, C.F., Srinivasan, R., Martínez, A.M.: Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 5562–5570. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.600, https://doi.org/10.1109/CVPR.2016.600

4. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)

5. Fan, Y., Lam, J., Li, V.: Facial action unit intensity estimation via semantic correspondence learning with dynamic graph convolution. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 12701–12708 (2020)

6. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.B.: Masked autoencoders are scalable vision learners. CoRR **abs/2111.06377** (2021), https://arxiv.org/abs/2111.06377

7. Iandola, F., Moskewicz, M., Karayev, S., Girshick, R., Darrell, T., Keutzer, K.: Densenet: Implementing efficient convnet descriptor pyramids. arXiv preprint arXiv:1404.1869 (2014)

8. Jacob, G.M., Stenger, B.: Facial action unit detection with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7680–7689 (2021)

9. Jiang, W., Wu, Y., Qiao, F., Meng, L., Deng, Y., Liu, C.: Model level ensemble for facial action unit recognition at the 3rd abaw challenge. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2337–2344 (2022)

10. Jin, C., Jin, R., Chen, K., Dou, Y.: A community detection approach to cleaning extremely large face database. Computational intelligence and neuroscience **2018** (2018)

11. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

12. Kollias, D.: Abaw: Learning from synthetic data & multi-task learning challenges. arXiv preprint arXiv:2207.01138 (2022)

13. Kollias, D.: Abaw: Valence-arousal estimation, expression recognition, action unit detection & multi-task learning challenges. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2328–2336 (2022)

14. Kollias, D., Cheng, S., Pantic, M., Zafeiriou, S.: Photorealistic facial synthesis in the dimensional affect space. In: Proceedings of the European Conference on Computer Vision (ECCV) Workshops. pp. 0–0 (2018)

15. Kollias, D., Cheng, S., Ververas, E., Kotsia, I., Zafeiriou, S.: Deep neural network augmentation: Generating faces for affect analysis. International Journal of Computer Vision **128**(5), 1455–1484 (2020)

16. Kollias, D., Nicolaou, M.A., Kotsia, I., Zhao, G., Zafeiriou, S.: Recognition of affect in the wild using deep neural networks. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on. pp. 1972–1979. IEEE (2017)

17. Kollias, D., Sharmanska, V., Zafeiriou, S.: Distribution matching for heterogeneous multi-task learning: a large-scale face study. arXiv preprint arXiv:2105.03790 (2021)

18. Kollias, D., Tzirakis, P., Nicolaou, M.A., Papaioannou, A., Zhao, G., Schuller, B., Kotsia, I., Zafeiriou, S.: Deep affect prediction in-the-wild: Aff-wild database and challenge, deep architectures, and beyond. International Journal of Computer Vision pp. 1–23 (2019)
19. Kollias, D., Zafeiriou, S.: Aff-wild2: Extending the aff-wild database for affect recognition. CoRR abs/1811.07770 (2018), http://arxiv.org/abs/1811.07770
20. Kollias, D., Zafeiriou, S.: Expression, affect, action unit recognition: Aff-wild2, multi-task learning and arcface. arXiv preprint arXiv:1910.04855 (2019)
21. Kollias, D., Zafeiriou, S.: Va-stargan: Continuous affect generation. In: International Conference on Advanced Concepts for Intelligent Vision Systems. pp. 227–238. Springer (2020)
22. Kollias, D., Zafeiriou, S.: Affect analysis in-the-wild: Valence-arousal, expressions, action units and a unified framework. arXiv preprint arXiv:2103.15792 (2021)
23. Li, S., Deng, W.: Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition. IEEE Transactions on Image Processing 28(1), 356–370 (2019)
24. Li, S., Deng, W., Du, J.: Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2584–2593. IEEE (2017)
25. Meng, L., Liu, Y., Liu, X., Huang, Z., Jiang, W., Zhang, T., Liu, C., Jin, Q.: Valence and arousal estimation based on multimodal temporal-aware features for videos in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2345–2352 (2022)
26. Mollahosseini, A., Hasani, B., Mahoor, M.H.: Affectnet: A database for facial expression, valence, and arousal computing in the wild. IEEE Transactions on Affective Computing 10(1), 18–31 (2017)
27. Nguyen, H.H., Huynh, V.T., Kim, S.H.: An ensemble approach for facial expression analysis in video. arXiv preprint arXiv:2203.12891 (2022)
28. Ruder, S.: An overview of multi-task learning in deep neural networks. CoRR abs/1706.05098 (2017), http://arxiv.org/abs/1706.05098
29. Sak, H., Senior, A., Beaufays, F.: Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. arXiv preprint arXiv:1402.1128 (2014)
30. Tao, J., Tan, T.: Affective computing: A review. In: Tao, J., Tan, T., Picard, R.W. (eds.) Affective Computing and Intelligent Interaction, First International Conference, ACII 2005, Beijing, China, October 22-24, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3784, pp. 981–995. Springer (2005). https://doi.org/10.1007/11573548_125, https://doi.org/10.1007/11573548_125
31. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems 30 (2017)
32. Wen, Z., Lin, W., Wang, T., Xu, G.: Distract your attention: multi-head cross attention network for facial expression recognition. arXiv preprint arXiv:2109.07270 (2021)
33. Zafeiriou, S., Kollias, D., Nicolaou, M.A., Papaioannou, A., Zhao, G., Kotsia, I.: Aff-wild: Valence and arousal 'in-the-wild'challenge. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on. pp. 1980–1987. IEEE (2017)
34. Zhang, S., An, R., Ding, Y., Guan, C.: Continuous emotion recognition using visual-audio-linguistic information: A technical report for abaw3. arXiv preprint arXiv:2203.13031 (2022)

35. Zhang, W., Qiu, F., Wang, S., Zeng, H., Zhang, Z., An, R., Ma, B., Ding, Y.: Transformer-based multimodal information fusion for facial expression analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2428–2437 (2022)