Modeling Noisy labels for Facial Expression Recognition

Mingjian Zhu, Hongxun Ding, Lanjing Yi

Computer Science and Engineering, SUSTECH, China

ABSTRACT. Facial expression recognition (FER) has a wide research prospect in human-computer interaction and emotion computing, including human-computer interaction, emotion analysis, intelligent security, entertainment, online education, intelligent medical care, etc. FER is challenging due to the class imbalance and noisy labels caused by data collection. We need to address both of these challenges in this project.

METHOD

The process consists of lots of epochs, some beginning epochs are warm up epoch, and the remaining epochs are train epochs.

To combat confirmation bias, which the model confirm its bias through the process, we apply two parallel models A and B, B is fixed while training A, and A is fixed while training B.

At the beginning of each epoch, we train the model. At the end of every epoch, there is a test module.

Warm-up

In each warm-up epoch, there is a warm-up module and a test module. The warm-up module utilize the total training set $S = s_i, i = 0, 1, 2, ..., n-1$, in which $s_i = (x_i, y_i)$.

Assume there are C classes, x_i is an image and y_i is a one-hot label.

We crew the samples batch by batch, a batch contains b samples. In each batch, the trained model makes a prediction p_i for every $x_i, i = 0, 1, 2, ..., b-1$, and we calculate the cross entrop loss $l_i = -\sum_1^C y_i[j]log(p_i[j])$.

And then, we use stochastic gradient descent with momentum to optimizer the trained model.

Train

In each train epoch, there is a train module and a test module. A train module consists of three stages. At the first stage, we calculate the cross entrop loss for every samples, and fit a gaussian distribution to the loss distribution $l_i, i = 0, 1, 2, ..., n-1$, get a probability distribution $w_i, i = 0, 1, 2, ..., n-1$. At the second stage, we compare the probability distribution with threshold $t_j, j = 0, 1, 2, ..., C-1$, and get a boolean distribution $r_i, i = 0, 1, 2, ..., n-1$. $r_i = w_i > t_j, for y_i = j$, and if $r_i = True$, we remain y_i as labeled sample, otherwise, we remove labels as unlabeled sample. At the last stage, we calculate cross entrop loss for every labeled sample, and use these losses to optimizer the trained model. What's more, we apply the penalty term in DivideMix to optimizer model, $penalty = \sum^C \pi_C log(\pi_C / \sum p_i for p_i = C)$.

Metrics

We calculate the recall for every class in the test set, and draw the confusion matrix.

EXPERIMENT

Model: Resnet50

Dataset: AffectNet, RAF-DB

PLAN

Current progress

Implement DivideMix, Ada-CM on AffectNet

Before the final report

Introduce some more methods into our implementation

Accomplish a relatively good accuracy rate

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Fig. 1. The parameters of resnet50.

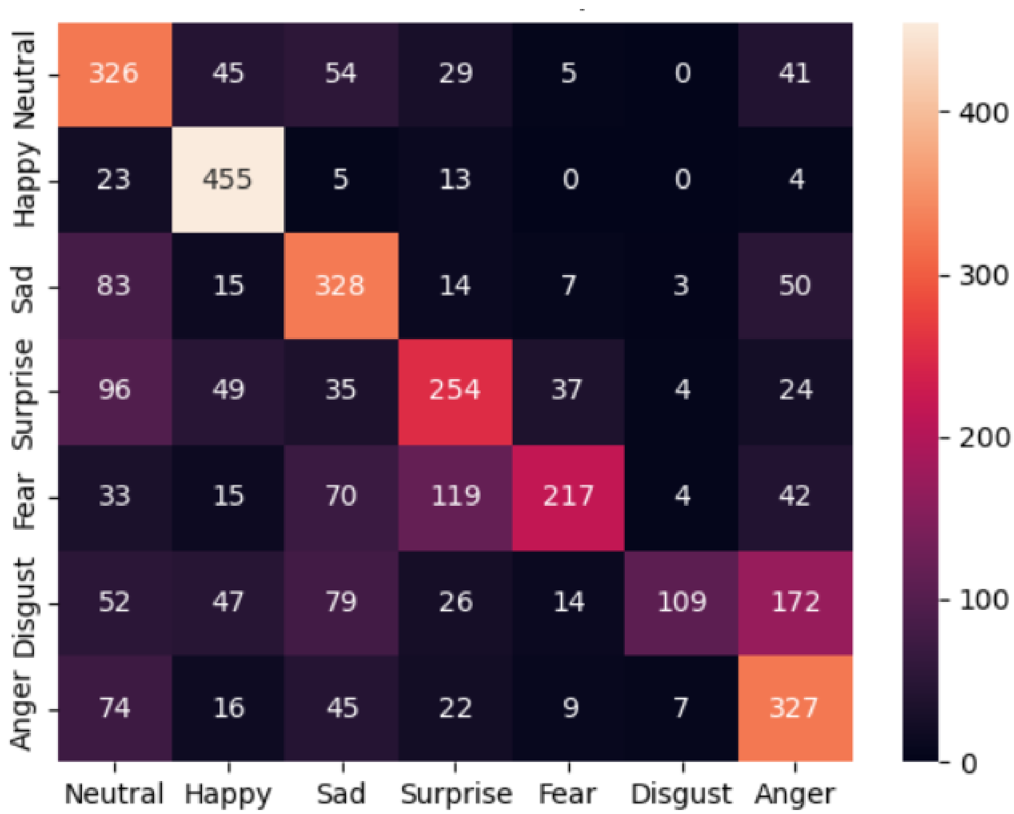

Fig. 2. Confusion matrix.

REFERENCES

1. Li, J., Socher, R., Hoi, S. C. (2020). Dividemix: Learning with noisy labels as semi-supervised learning. arXiv preprint arXiv:2002.07394.

2. Li, H., Wang, N., Yang, X., Wang, X., Gao, X. (2022). Towards Semi-Supervised Deep Facial Expression Recognition with An Adaptive Confidence Margin. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 4166-4175).

3. Wei, C., Sohn, K., Mellina, C., Yuille, A., Yang, F. (2021). Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 10857-10866).

4. Algan, G., Ulusoy, I. (2021). Image classification with deep learning in the presence of noisy labels: A survey. Knowledge-Based Systems, 215, 106771.

5. Song, H., Kim, M., Park, D., Shin, Y., Lee, J. G. (2022). Learning from noisy labels with deep neural networks: A survey. IEEE Transactions on Neural Networks and Learning Systems.