LION: Label Disambiguation for Semi-supervised Facial Expression Recognition with Progressive Negative Learning

Zhongjing Du[1*], **Xu Jiang**[1*], **Peng Wang**[1], **Qizheng Zhou**[2], **Xi Wu**[3], **Jiliu Zhou**[1], **Yan Wang**[1†]
[1]School of Computer Science, Sichuan University
[2]School of Applied Mathematics, State University of New York
[3]School of Computer Science, Chengdu University of Information Technology
{duzhongjing, jiangxu}@stu.scu.edu.cn, wpen@scu.edu.cn, qizheng_zhou@163.com,
wuxi@cuit.edu.cn, zhoujl@scu.edu.cn, wangyanscu@hotmail.com

Abstract

Semi-supervised deep facial expression recognition (SS-DFER) has recently attracted rising research interest due to its more practical setting of abundant unlabeled data. However, there are two main problems unconsidered in current SS-DFER methods: 1) label ambiguity, i.e., given labels mismatch with facial expressions; 2) inefficient utilization of unlabeled data with low-confidence. In this paper, we propose a novel SS-DFER method, including a **L**abel D**I**sambiguation module and a Pr**O**gressive **N**egative Learning module, namely LION, to simultaneously address both problems. Specifically, the label disambiguation module operates on labeled data, including data with accurate labels (clear data) and ambiguous labels (ambiguous data). It first uses clear data to calculate prototypes for all the expression classes, and then re-assign a candidate label set to all the ambiguous data. Based on the prototypes and the candidate label set, the ambiguous data can be relabeled more accurately. As for unlabeled data with low-confidence, the progressive negative learning module is developed to iteratively mine more complete complementary labels, which can guide the model to reduce the association between data and corresponding complementary labels. Experiments on three challenging datasets show that our method significantly outperforms the current state-of-the-art approaches in SS-DFER and surpasses fully-supervised baselines. Code will be available at *https://github.com/NUM-7/LION*.

1 Introduction

Facial expression is one of the most common displays of human emotion, which plays an important role in interpersonal communications. In the past few decades, with the emergence of large-scale well-labeled facial datasets, e.g., AffectNet [Mollahosseini *et al*., 2017] and RAF-DB [Li *et al*., 2017], many automatic facial expression recognition (FER) approaches based on fully supervised deep learning have been proposed and made significant progress in distinguishing facial expressions, including surprise, fear, disgust, happiness, sadness, anger and neutral. However, in the real world, due to the expensive and tedious annotation process, it is extremely difficult to collect a large volume of high-quality labels for a dataset.

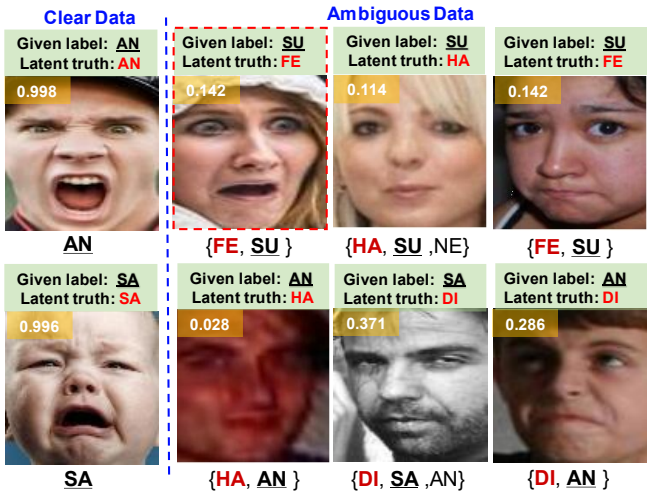

Figure 1: Clear and Ambiguous examples from RAF-DB dataset, including surprise (SU), fear (FE), disgust (DI), happiness (HA), sadness (SA), anger (AN) and neutral (NE). All the examples are reassessed by 35 volunteers and the confidence scores are presented in yellow bar. The given labels and latent truths are provided above the images. The sets below images are the candidate label sets constructed by our method.

Recently, semi-supervised deep facial expression recognition (SS-DFER) methods have been developed to make use of the unlabeled data. For example, Ada-CM [Li *et al*., 2022] applied the common semi-supervised learning (SSL) techniques, e.g., pseudo labeling and consistency regularization, to the FER task and learned an adaptive confidence margin to fully leverage unlabeled data. Nonetheless, we argue that there are still two main problems lacking consideration in the current SS-DFER methods, which impedes further performance improvement. The first one is the label ambiguity

problem in the labeled set. It means that the given labels may mismatch with the facial expressions due to (i) data variations in illumination, resolution, occlusion, and pose, (ii) ambiguity in expressions, (iii) crowd sourcing, and (iv) low-quality annotations obtained on a search engine. We present some clear and ambiguous examples from RAF-DB dataset in Fig.1. All the images are re-assessed by 35 volunteers. Taking the image annotated with "Surprise" (with red border) for example, the confidence score is only 0.142, and most volunteers think it should be "Fear" instead, which might be the potentially real label (latent truth). Training with these ambiguous data may result in difficulty of convergence and inevitable performance degradation.

The second problem of the current SS-DFER methods concerns the inefficient utilization of unlabeled data. As mentioned above, existing SS-DFER methods typically adopt strategies such as pseudo-labeling and consistency regularization to leverage the unlabeled data. To ensure that reliable information is mined from unlabeled data, these methods usually keep the unlabeled data with high confidence scores predicted by the models, but disregard those data with uncertain predictions. Such naïve operations only explore the value of a fraction of easy unlabeled data, wasting the rich information contained in low-confidence samples.

In this paper, we propose a novel SS-DFER approach with a **L**abel D**I**sambiguation module and a Pr**O**gressive **N**egative Learning module, namely LION, to simultaneously address the above two problems, i.e., label ambiguity and inefficient utilization of unlabeled data. For the former, the label disambiguation module is designed to screen out those ambiguously labeled data and re-assign them with more accurate labels. Specifically, if the given label disagrees with the class prediction by the classifier, we define it as an ambiguous label, otherwise a clear label. All the data with clear labels, or clear data, are utilized to derive class prototypes of the seven facial expressions, which guide the model to correct the ambiguous labels. For the ambiguous data, considering the subtle differences among different facial expressions, unlike previous methods that only assign one label to facial images, we maintain *a candidate label set* with a collection of candidate classes to store potentially real labels (as shown the sets below the facial images in Fig. 1) and update the original labels based on the candidate label set and class prototypes. For the latter problem, we first split the unlabeled data into a reliable set with high-confidence and an unreliable set with low-confidence. All the reliable data will also involve the calculation of class prototypes. As for the unreliable data, since it is difficult to give accurate class predictions directly, we consider negative learning [Kim *et al*., 2019; Duan *et al*., 2022] to excavate knowledge from complementary labels which indicate the classes the image does not belong to. Concretely, we take an iterative manner to progressively store all the class predictions with the lowest probability, i.e., complementary labels, in a memory bank. Meanwhile, the model is also trained with the complementary labels during iteration, so as to exclude those impossible classes for the unreliable data. In addition, we design a negative consistency loss (NC-Loss) to constrain the complementary labels of different augmented views of the same unlabeled data to be consistent, intending to reduce the variations of complementary labels. By the above manner, all the unlabeled data can be fully used. Overall, our contributions can be summarized as follows:

- We propose a novel SS-DFER method LION to reduce the impact of ambiguous labels by exploring the potentially real labels in a candidate label set. To the best of our knowledge, this is the first work which considers the label ambiguity problem in SS-DFER.
- A progressive negative learning module is also presented in LION to draw the knowledge from the complementary labels, thus fully leveraging the underestimated unlabeled data with low-confidence. In addition, we innovatively design a NC-Loss to further improve the performance of the model.
- Extensive experiments on three challenging datasets show the effectiveness of our proposed model. Particularly, it sets new records of recognition accuracy with 67.83% on RAF-DB and 45.61% on SFEW, which is much higher than the second-best SOTA method [Li *et al*., 2022] by 5.47% and 3.73%.

2 Related Work

2.1 Facial Expression Recognition

FER helps computers to understand the emotional state of humans, which is meaningful for intelligent human-computer interaction. Previous works mainly experimented on the laboratory-generated dataset, including CK+ [Patrick *et al*., 2010], Oulu-Casia [Zhao *et al*., 2011], and achieved inspiring recognition accuracy. However, in the wild, it is difficult to guarantee that all images are of high quality. Therefore, more and more works [Wang *et al*., 2020; She *et al*., 2021; Xue *et al*., 2021; Yang *et al*., 2023] have extended the research interest to the in-the-wild datasets.

A troublesome problem in the in-the-wild datasets is the ambiguous images, that is, the given label does not match with the facial expression due to various factors. To address this problem, [Zeng *et al*., 2018] introduced multiple training stages to solve inconsistency of annotations. [Chen *et al*., 2020] explored label distribution through constructing auxiliary label space graphs. [Wang *et al*., 2020] focused on mining confidence weight and the latent truth of each sample for smaller impact of ambiguous data. Leading performance has been achieved by [She *et al*., 2021] which tried to find latent distribution in the label space and estimate the pairwise uncertainty.

Despite the achievements in the in-the-wild FER task, most of the current methods are fully supervised and require a large scale of well-labeled dataset, which is time-cost and labor-intensive. [Florea *et al*., 2020] made the first attempt to investigate semi-supervised deep FER (SS-DFER) and proposed an extension of MixMatch [Berthelot *et al*., 2019]. [Li *et al*., 2022] proposed an adaptive thresholding approach to

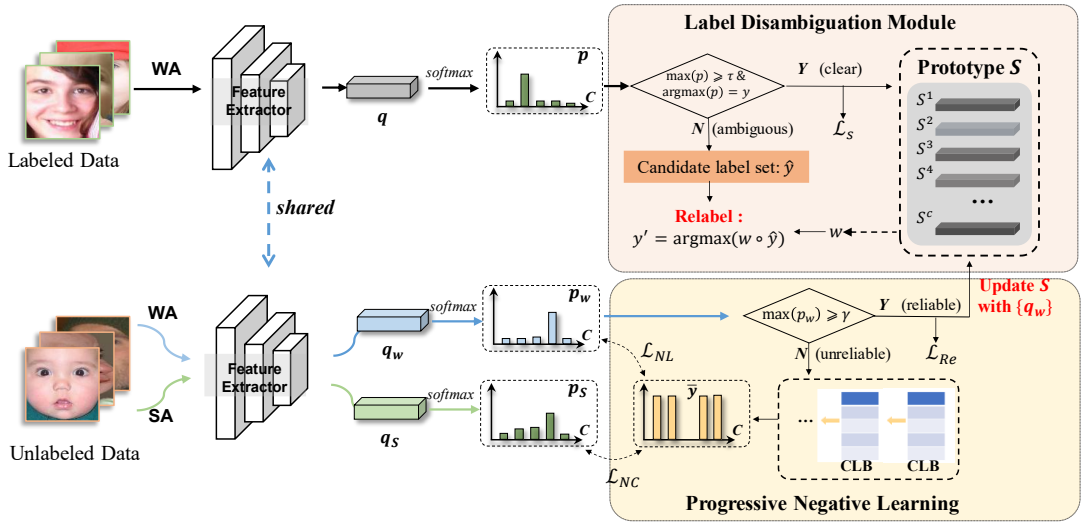

Figure 2: Illustration of our LION model. The labeled images are divided into ambiguous and clear sets. Clear images are used to compute prototypes of all expression classes, and also to train the model by the L_S loss. Ambiguous images are relabeled by the label disambiguation module. The unlabeled images are split into reliable and unreliable sets. For the reliable images, they also contribute to the computation of prototypes, and optimize the model by the L_{Re} loss. As for the unreliable images, we propose a progressive negative learning module to mine their complementary labels which are utilized to supervise the model by the L_{NL} loss. Additionally, a negative consistency loss L_{NC} is imposed between the weakly and strongly augmented images.

generate reliable pseudo-labels for high-confidence unlabeled samples. However, the SS-DFER is still mostly an unexplored research field. This work aims to tackle two main challenges in SS-DFER, including label ambiguity and inefficient utilization of unlabeled data with low-confidence.

2.2 Semi-Supervised Learning

Semi-supervised learning (SSL) appropriately frees up the stringent requirement of fully supervised learning for completely labeled dataset, while at the same time guaranteeing the performance of the model. Existing SSL methods can be divided into the following five main approaches: graph-based methods [Marino *et al*., 2016; Wang *et al*., 2020], generative model-based methods [Donahue *et al*., 2016; Denton *et al*., 2016], methods using consistent regularization [Sajjadi *et al*., 2016; Xie *et al*., 2020], methods using pseudo labeling [Rizve *et al*., 2021; Pham *et al*., 2021] and hybrid methods [Sohn *et al*., 2020; Zhang *et al*., 2021; Xu *et al*., 2021].

Among them, the hybrid methods generally achieve state-of-the-art performance on datasets such as CIFAR-10, CIFAR-100 [Krizhevsky *et al*., 2009], and ImageNet [Deng *et al*., 2009]. For example, [Xie *et al*., 2020] and [Sohn *et al*., 2020] set fixed thresholds to obtain pseudo labels for weakly augmented unlabeled images, and used them to supervise the prediction of the strongly augmented counterpart. [Zhang *et al*., 2021] and [Xu *et al*., 2021] further explored the application of dynamic thresholding in semi-supervised tasks. However, direct application of these SSL approaches to the SS-DFER task is unsatisfactory due to the interference of ambiguous data and the underutilization of unlabeled data. In our work, we solve both the problems by proposing a disambiguation module and a progressive negative learning module.

3 Method

Our proposed LION aims to tackle the FER task in a semi-supervised setting. Specifically, a label disambiguation module is devised to correct the ambiguous labels in the labeled set, while a progressive negative learning module is designed to make full use of unlabeled data, especially for those low-confidence data. In this section, we first illustrate our problem formulation and show the overview of our method in Sec. 3.1. Then, the label disambiguation module of our method is introduced in Sec. 3.2. Furthermore, we describe how the progressive negative learning module works in Sec. 3.3. Finally, we display the whole training objective in Sec. 3.4.

3.1 Overview

In the semi-supervised setting, we are provided with a labeled set $D_L = \{x_i, y_i\}_{i=1}^{M}$ and an unlabeled set $D_U = \{x_i\}_{i=M+1}^{M+N}$, where x_i is the input image, y_i is corresponding one-hot label with C classes, M and N are the number of data in D_L and D_U, respectively. Notably, not all the labels $\{y_i\}$ are correctly assigned for their corresponding $\{x_i\}$. Our purpose is to train a robust model to relabel the ambiguous labels and distinguish facial expressions accurately by fully leveraging both a large number of unlabeled images and few labeled images. Fig.2 gives an overview of our model. For a labeled image, we firstly apply weak augmentation (WA) to it and send it to a

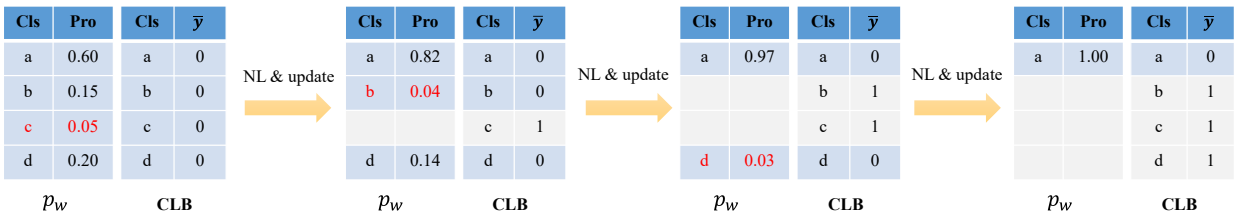

Figure 3: Diagram of progressive negative learning. "Cls" represents the class list (shown as a, b, c, d). "Pro" represents the predicted probability score corresponding to each class. "CLB" denotes the complementary label bank. "$\bar{y}$" stores the selected classes for the complementary label. In each iteration, we choose the class corresponding to the smallest probability value less than δ (marked with red), and set its $\bar{y}$ to 1. Then, the "$\bar{y}$" of CLB forms the complementary label to optimize the model by Eq. 9 and Eq. 10. In the next iteration, new probability scores of all classes (excluding the previous chosen ones) are predicted, and the similar operations are conducted until the remaining probability scores are all higher than δ.

feature extractor to obtain the feature vector q, followed by a softmax function to derive its prediction probability p. To remedy the ambiguous labels, the label disambiguation module first screens out the clear and ambiguous images separately according to the maximum probability in p, i.e., $\max(p)$. The clear images are used to compute the prototypes of all the expressions, guiding the relabeling process. All the ambiguous images will discard the original labels and be assigned with a candidate label set with multiple candidate classes, then approach the latent truth via the candidate labels and the prototypes. For an unlabeled image, both WA and strong augmentation (SA) are applied, and the weight-sharing feature extractor is employed to obtain the feature vectors q_w, q_s, and their prediction probabilities p_w and p_s, respectively. Then, based on the maximum probability in p_w, i.e., $\max(p_w)$, we divide the unlabeled images into two subsets. Specifically, if $\max(p_w)$ is greater than a certain threshold, we define it as a reliable sample and the pseudo label $\tilde{y}$ predicted from WA version will be used to supervise the SA version via the cross-entropy loss. Otherwise, we define it as an unreliable sample, and our proposed progressive negative learning strategy will be used to unearth the complementary labels which lead the learning of unreliable samples. Meanwhile, a negative consistency loss (NC-Loss) is proposed to enforce the consistency between the complementary labels of WA and SA versions, ensuring the accuracy and robustness of complementary labels. We will elaborate on key technologies in the following sub-sections.

3.2 Label Disambiguation Module

For a labeled image, we first generate a weakly-augmented version and obtain its prediction probability p via the feature extractor and softmax function. If the maximum predicted probability $\max(p)$ exceeds a certain threshold τ and the class of $\max(p)$ is the same with the given label y, we define it as a clear image. Otherwise, it will be defined as an ambiguous one:

$$x_i \in \begin{cases} \text{clear image}, & \text{if } \max(p) \geq \tau \text{ and } \operatorname{argmax}(p) = y \\ \text{ambiguous image}, & \text{otherwise} \end{cases}, \quad (1)$$

where τ is a positive constant.

For the clear images, we use the given label y to supervise the predicted probability p:

$$L_S = \frac{1}{N_{cl}} \sum_1^{N_{cl}} CE(p, y), \quad (2)$$

where N_{cl} represents the number of clear images and $CE(\cdot)$ denotes the standard cross-entropy. In addition, since clear images have high-confidence labels, we also employ their feature vectors q to calculate prototypes $\{S^c\}$ corresponding to each class $c \in (1, \dots, C)$. These prototypes can be viewed as anchors of each class, which can guide the subsequent relabeling process. Particularly, the prototypes are updated in an Exponential Moving Average (EMA) fashion:

$$S^c = \text{Normalize}\big(\phi S^{c\prime} + (1-\phi)q\big), \text{if } c = \operatorname{argmax}(p), \quad (3)$$

where $S^{c\prime}$ is the prototype of last iteration, ϕ is set to 0.99 following [Wang *et al.*, 2022].

As for each ambiguous image, we assign a candidate label set $\hat{y}$ (i.e., a C-dimension vector) to it. Different from the one-hot label y, $\hat{y}$ is a multi-hot vector with multiple classes setting to 1, any of which could potentially be the unique real label. We derive the candidate label set $\hat{y}$ as follows:

$$\hat{y}^c = \begin{cases} 1 & \text{if } c = \operatorname{argmax}(p) \\ 1 & \text{else if } p^c \geq \mu \\ 0 & \text{otherwise} \end{cases}, \quad (4)$$

where $\hat{y}^c$ is the c-th class in $\hat{y}$, p^c is the probability score corresponding to the class c, μ is a threshold. In this manner, all possible candidate classes that could become the real label are selected to construct the candidate label set. To further approximate the real label, we make the label of each candidate class soft by assigning a weight w^c to $\hat{y}^c$, and propose a moving-average style strategy to update the weight:

$$w^c = \phi w^{c\prime} + (1-\phi) d(q, S^c), \quad (5)$$

where w^c is initialized by the predicted probability of class c, $w^{c\prime}$ is the weight of last iteration, $d(\cdot)$ is the cosine similarity between the feature representation q and the prototype corresponding to the class S^c. When the weight of one class in the candidate label set exceeds the threshold τ, the image will be relabeled as the most convinced class.

$$y' = \operatorname{argmax}(w^c \circ \hat{y}^c), \text{ if } \max(w^c \circ \hat{y}^c) \geq \tau. \quad (6)$$

Then it can be viewed as a clear image and optimize the model like Eq. 2.

3.3 Progressive Negative Learning

To fully leverage all unlabeled samples, we design a progressive negative learning module to explore the knowledge contained in them, especially for those low-confidence unlabeled samples, so as to further improve the performance of the model.

For each unlabeled sample, we transform it into both WA and SA versions. Then, we classify the unlabeled sample into reliable set or unreliable set according to its maximum probability of WA version, i.e., $\max(p_w)$. If $\max(p_w)$ is larger than a threshold γ, the unlabeled sample is of high confidence for the model, thus belonging to the reliable set. Otherwise, the unlabeled sample with low confidence belongs to the unreliable set [Li *et al*., 2022]. We formulate this as follows:

$$x_i \in \begin{cases} \text{reliable set}, & \text{if } \max(p_w) \geq \gamma \\ \text{unreliable set}, & \text{otherwise} \end{cases}, \quad (7)$$

The reliable samples have two applications. First, since model trusts the predictions of them with high confidence, we also utilize their feature vectors $\{q_w\}$ to update the class prototypes by replacing q in Eq. 3 with q_w, thereby generating more robust prototypes. Second, all the reliable samples will be used to train the model like FixMatch [Sohn *et al*., 2020]. Concretely, we pick up the class with the maximum probability as the pseudo label $\tilde{y} = \text{argmax}(p_w)$, then harness the pseudo label to supervise the prediction of the SA version, as follows:

$$L_{Re} = \frac{1}{N_{re}} \sum_{1}^{N_{re}} CE(p_s, \tilde{y}), \quad (8)$$

where N_{re} denotes the number of reliable unlabeled samples.

As for the unreliable samples, although it is difficult to predict which expression category they belong to, there should be some categories with a sufficiently low probability score. Hence, it is easy for the model to know the classes these unreliable samples do not belong to, which are called complementary labels. To get as comprehensive a complementary label as possible, we maintain a complementary label bank (CLB) and update it progressively, as shown in Fig. 3. Specifically, given the prediction p_w of an unreliable sample, we find its minimum probability score $\min(p_w)$. If $\min(p_w)$ is smaller than a threshold δ, we add its corresponding class into the CLB, and accordingly get the complementary label $\bar{y}$ as follows:

$$\bar{y}^c = \begin{cases} 1, & \text{if } \min(p_w) \leq \delta \text{ and } c = \text{argmin}(p_w) \\ 0, & \text{otherwise} \end{cases}, \quad (9)$$

where δ is a positive constant to ensure that there is sufficiently strong confidence to assign 1s in the complementary label. Afterwards, the complementary label $\bar{y}$ is used to train the model via negative learning as follows:

$$L_{NL} = -\sum_{c \in C} \bar{y}^c (1 - p_w^c), \quad (10)$$

where p_w^c represents the predicted probability of class c in p_w.

The above process will be performed iteratively until the remaining probabilities are all higher than the certain threshold δ. It is worth noting that the classes in previous complementary labels will not involve the probability prediction in the next iterations.

Algorithm 1 The training process of LION

Input: Labeled set $D_L = \{x_i, y_i\}_{i=1}^{M}$, unlabeled set $D_U = \{x_i\}_{i=M+1}^{M+N}$.
Parameters: The maximum number of epochs E, the thresholds τ, μ, γ, and δ.
Output: Updated LION model.
/ Training*/*
for $e = 1: E$ **do**
  */*For labeled set*/*
  for $i = 1: M$ **do**
    Weakly augment x_i and obtain its feature vector q and prediction probability p of x_i.
    if $\max(p) \geq \tau$ and $\text{argmax}(p) = y$ **then**
      Compute L_S by Eq. 2.
      Compute prototypes S by Eq. 3.
    else
      Assign a candidate label set $\hat{y}$ by Eq. 4.
      Soften the candidate labels by weights w and update them by Eq. 5.
      Relabel x_i with y' by Eq. 6.
    end if
  end for
  */*For unlabeled set*/*
  for $i = M + 1: M + N$ **do**
    Weakly and strongly augment x_i and obtain its feature vectors q_w and q_s as well as its prediction probabilities p_w and p_s.
    if $\max(p_w) \geq \gamma$ **then**
      Update prototypes S by Eq. 3.
      Compute L_{Re} by Eq. 8.
    else
      while $\min(p_w) \leq \delta$ **do**
        Obtain the complementary label $\bar{y}$ by Eq.9.
        Compute L_{NL} by Eq. 10.
        Re-compute q_w, q_s, p_w, p_s.
      end while
      Compute L_{NC} by Eq. 11.
    end if
  end for
end for

In addition, to further reduce the variations of complementary labels and enhance the robustness of the model, we propose a negative consistency loss (NC-Loss) between the WA and SA versions:

$$L_{NC} = -\sum_{c \in C} \bar{y}^c \log(1 - p_s^c), \quad (11)$$

where $\bar{y}^c$ refers to the complementary label obtained from the WA version and p_s is the prediction of the SA version. For better understanding, we summarize the whole training process in Algrithm 1.

3.4 Overall Objective Function

Method	RAF-DB				SFEW		AffectNet	
	100labels	400labels	2000labels	4000labels	100labels	400labels	2000labels	10000labels
Baseline	52.43±2.24	67.75±0.95	78.91±0.43	81.90±0.48	33.76±1.84	43.85±2.83	47.52±0.75	53.18±0.68
Pseudo-Labeling [Lee *et al*., 2013]	54.96±4.24	69.99±1.81	79.18±0.27	82.88±0.49	34.27±1.67	45.27±1.32	48.78±0.67	53.82±1.29
MixMatch [Berthelot *et al*., 2019]	54.57±4.16	73.14±1.40	79.63±0.91	83.57±0.49	34.13±2.58	44.91±1.87	49.63±0.49	53.49±0.47
UDA [Xie *et al*., 2020]	58.15±1.54	72.39±1.64	81.16±0.54	83.56±0.82	39.22±2.30	48.90±1.56	50.42±0.45	56.49±0.27
ReMixMatch [Berthelot *et al*.,2020]	58.83±2.34	73.34±1.82	79.66±0.66	83.51±0.18	35.69±2.73	48.39±0.71	50.38±0.63	55.81±0.34
MarginMix [Florea *et al*., 2020]	58.91±1.78	73.31±1.64	80.22±0.76	83.47±0.28	38.69±1.93	49.21±0.92	50.58±0.42	56.41±0.28
FixMatch [Sohn *et al*., 2020]	60.67±2.25	73.36±1.59	81.27±0.27	83.31±0.33	38.90±1.90	50.73±0.45	50.79±0.37	56.50±0.43
Ada-CM [Li *et al*., 2022]	62.36±1.10	74.44±1.53	82.05±0.22	84.42±0.49	41.88±2.12	52.43±0.67	51.22±0.29	57.42±0.43
LION	**67.83±0.64**	**76.43±1.12**	**82.39±0.13**	**84.81±0.16**	**45.61±0.32**	**54.18±0.52**	**52.71±0.21**	**59.11±0.38**
Fully Supervised	84.13				51.05		52.97	

Table 1: Performance comparison (%) with the state-of-the-art methods on RAF-DB, SFEW and AffectNet. The best results are in **bold** font and the second-best results are underlined.

As mentioned above, there are four losses to optimize the parameters of our LION model: 1) L_S loss on labeled clear data; 2) L_{Re} loss on unlabeled reliable data; 3) L_{NL} loss on the WA version of unlabeled unreliable data; 4) L_{NC} loss between the WA and SA versions of unlabeled unreliable data. Our LION model is optimized in an end-to-end process. To sum up, the total loss is formulated as follows:

$$L_{total} = L_S + \lambda_1 L_{Re} + \lambda_2 L_{NL} + \lambda_3 L_{NC}, \quad (12)$$

where λ_1, λ_2, and λ_3 are hyper-parameters to balance these terms.

4 Experiments

4.1 Datasets and Metrics

Datasets. We verify the effectiveness of our LION model on three datasets: RAF-DB [Li *et al*., 2017], AffectNet [Mollahosseini *et al*., 2017], and SFEW [Dhall et al., 2011]. **RAF-DB** is constructed by 30,000 facial images which are annotated by 40 experts. In the experiment, we select fear, surprise, sadness, happiness, disgust, anger and neutral as classification categories. The sizes of the training and test sets are 12,271 and 3,068, respectively. **AffectNet** has 420,000 face images containing 8 manually annotated expression labels. We choose the same seven categories as RAF-DB in experiment and use 240,000 images for training and 3,500 images for validation. **SFEW** contains static frames extracted from movies including 958 training images and 436 test images. To simulate the semi-supervised setting, we randomly discard a portion of labels at different ratios.

Performance Metrics. We conduct experiments using different random seeds and calculate the mean accuracy and standard deviation on the test set to evaluate the performance of the method.

4.2 Implementation Details

By default, ResNet-18 is used as backbone network, which is pre-trained on MS-Celeb-1M face recognition dataset [Guo *et al*., 2016]. Facial images are aligned and resized to 224×224 by MTCNN [Zhang *et al*., 2016]. RandomCrop and RandomHorizontalFlip are employed as weak augmentation. RandAugment [Cubuk *et al*., 2020] is used as strong augmentation following [Li *et al*., 2022]. The whole network is trained for 20 epochs with the Adam optimizer. The initial learning rate is set to 5×10^{-4}. The batch size is 16. The above setting keeps consistent with that of all the compared methods for fairness. As for the hyperparameters, we set τ=0.80, γ=0.83, δ=0.05, μ=0.3 and the trade-off parameters λ_1, λ_2, λ_3 are set to 0.85, 0.45, 0.06, respectively.

4.3 Comparison With the State-of-the-Art

To test the performance of LION, we compare it with several state-of-the-art methods, including Pseudo-Labeling [Lee *et al*., 2013], MixMatch [Berthelot *et al*., 2019], UDA [Xie *et al*., 2020], Margin-Mix [Florea *et al*., 2020], ReMixMatch [Berthelot *et al*.,2020], FixMatch [Sohn *et al*., 2020], and Ada-CM [Li *et al*., 2022], on all the three datasets with different ratios of labeled data. All these methods are highly representative and influential in the field of semi-supervised image classification, and we have tailored them to adapt our semi-supervised deep FER (SS-DFER) task. Particularly, we regard our model trained using only limited labeled data as the baseline. And we apply DLP-CNN [Li *et al*., 2019] on RAF-DB and SFEW, RAN [Wang *et al*., 2020] on AffectNet as fully-supervised baseline.

Table 1 shows the comparison results. From the table, it is clear that all the semi-supervised methods achieve better performance than baseline due to the utilization of unlabeled data. Compared with the second-best method Ada-CM, our method outperforms it overwhelmingly under all ratios of labeled data on all datasets. Even when only 100 images are labeled in RAF-DB and SFEW, our method still leads the performance by 5.47% and 3.73% accuracy, showing the powerful capability of our method to make use of unlabeled data. Furthermore, compared with the fully supervised result, our

LD	L_{NL}	L_{NC}	RAF-DB	SFEW
			100labels	400labels
			58.21±2.21	47.89±1.56
✓			61.19±1.23	51.82±1.12
✓	✓		65.06±0.21	53.66±0.43
✓		✓	61.86±0.32	52.61±0.35
✓	✓	✓	67.83±0.64	54.18±0.52

Table 2: Evaluation (%) of LD, L_{NL}, and L_{NC} on RAF-DB and SFEW.

method can still beat the baseline with a large margin, i.e., 0.68% on RAF-DB, 3.13% on SFEW, 6.14% on AffectNet when 4000, 400, 10000 labeled samples are used. The above results show that our model makes fuller use of the unlabeled data and reduces the negative impact caused by the ambiguous data.

4.4 Ablation Study

In this section, we carry out ablation study to verify the contribution of components and investigate the optimal values of hyper-parameters in LION.

Effectiveness of Components in LION. There are two important components in our LION model: 1) the label disambiguation module (LD for short) and 2) the progressive negative learning module. We intend to ablate them from the whole module to evaluate their contribution. Particularly, we attribute the contribution of the progressive negative learning module to the L_{NL} loss and the L_{NC} loss. We construct a baseline by removing the LD module, the L_{NL} loss and the L_{NC} loss. The experiment is performed on RAF-DB and SFEW with 100 labels and 400 labels, respectively. From Table 2, we can observe that the LD module significantly improves the performance (e.g., +2.98% on RAF-DB and +3.93% on SFEW), which signifies its capability in reducing the impact of ambiguous labels and optimizing the decision boundary. Moreover, when L_{NL} or L_{NC} is incorporated, the performance further rises by 3.87% on RAF-DB or 0.79% on SFEW. Both losses are employed simultaneously together with the LD module will bring greater performance gains. This is because both the modules are actually conducive to each other. Specifically, the progressive negative learning module enables the model to learn more knowledge from unlabeled data, which can encourage the model to give more reliable predictions. These reliable predictions provide more accurate information to the prototypes, helping the LD module correct more ambiguous labels. Correspondingly, more clear data will also optimize the decision boundary of the model, thus promoting the progressive negative learning module to work more effectively. In summary, all these results fully demonstrate the effectiveness of the proposed components.

Evaluation of the Parameter τ. The parameter τ is the threshold to determine if the data is ambiguous or not. We investigate its effect under values in [0.6,0.9]. Figure 4(a) shows that the performance is positively correlated with the

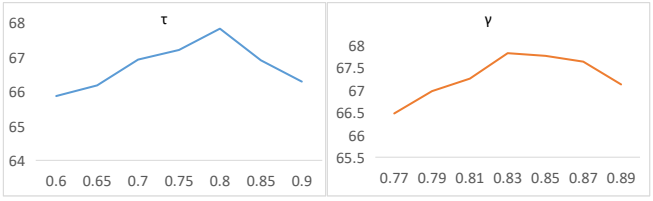

Figure 4: Evaluation of parameters (a) τ and (b) γ on RAF-DB.

Figure 5: Visualization of the relabeling results by our LION. **Blue**: original ambiguous label. **Red**: corrected label by LION.

increasing τ from 0.6 to 0.8. When τ exceeds 0.8, the performance degrades. This is because a large or small τ will guide the model with either wrong ambiguous data or clear data. Accordingly, we set τ as 0.8.

Evaluation of the Parameter γ. The parameter γ is used to classify the unlabeled data as reliable or unreliable ones. We investigate its effect under values in [0.77,0.89]. From Figure 4(b), we can see that the best result is achieved when γ=0.83. When γ is too small, more data with wrong pseudo labels are considered reliable and the updates of prototypes will be affected.

4.5 Visualization

In this section, we show some ambiguous examples relabeled by our method. In Figure 5, the blue represents original ambiguous label and the red represents corresponding corrected label by our LION model. We can observe that the corrected labels by our model are more in tune with human intuition. For example, the girl in the third column obviously has signals like a tight face and a closed mouth, which are barely possible to be "surprise" from the perspective of human. On the contrary, the corrected label "fear" is more suitable for these signals.

5 Conclusion

In this paper, we propose a novel SS-DFER method LION which includes a label disambiguation module and a progressive negative learning module. The label disambiguation module corrects the given ambiguous labels based on the candidate label set and prototypes. Progressive negative learning module mines complementary labels more completely during iteration. In addition, a negative consistency loss (NC-Loss) is proposed for a more robust model by reducing the variations of complementary labels. Extensive experiments conducted on three challenging datasets show that LION achieves state-of-the-art results and surpasses fully-supervised baselines.

Acknowledgments

This work is supported by National Natural Science Foundation of China (NSFC 62071314), Sichuan Science and Technology Program 2023YFG0263, 2023NSFSC0497, and Opening Foundation of Agile and Intelligent Computing Key Laboratory of Sichuan Province.

Contribution Statement

Zhongjing Du and Xu Jiang contributed equally to this work. Yan Wang is the corresponding author to the work.

References

[Berthelot *et al*., 2019] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, Colin A. Raffel. Mixmatch: A holistic approach to semi-supervised learning. In NeurIPS, 2019.

[Chen *et al*., 2020] Shikai Chen, Jianfeng Wang, Y uedong Chen, ZhongchaoShi, Xin Geng, and Y ong Rui. Label distribution learningon auxiliary label space graphs for facial expression recognition. In CVPR, 2020.

[Deng *et al*., 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In CVPR, 2009.

[Denton *et al*., 2016] Emily Denton, Sam Gross, Rob Fergus. Semi-Supervised Learning with Context-Conditional Generative Adversarial Networks. arXiv:1611.06430, 2016.

[Dhall *et al*., 2011] Abhinav Dhall, Roland Goecke, Simon Lucey, Tom Gedeon. Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. In ICCV, 2011.

[Donahue *et al*., 2016] Jeff Donahue, Philipp Krähenbühl, Trevor Darrell. Adversarial feature learning. arXiv:1605.09782, 2016.

[Duan *et al*., 2022] Yue Duan, Zhen Zhao, Lei Qi, Lei Wang, Luping Zhou, Yinghuan Shi, Yang Gao. MutexMatch: Semi-supervised Learning with Mutex-based Consistency Regularization. arXiv:2203.14316, 2022.

[Florea *et al*., 2020] Corneliu Florea, Mihai Badea, Laura Florea, AndreiRacoviteanu, Constantin Vertan. Margin-mix: Semi-supervised learning for face expression recognition. In ECCV, 2020.

[Guo *et al*., 2016] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In ECCV, 2016.

[Kim *et al*., 2019] Youngdong Kim, Junho Yim, Juseung Yun, Junmo Kim. NLNL: Negative learning for noisy labels. In ICCV, 2019.

[Krizhevsky *et al*., 2009] Krizhevsky, Alex, and Geoffrey Hinton. Learning multiple layers of features from tiny images. In Technical Report TR-2009, University of Toronto, 2009.

[Li *et al*., 2017] Shan Li and Weihong Deng. Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition. In CVPR, 2017.

[Li *et al*., 2019] Shan Li, Weihong Deng. Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition. In IEEE Transactions on Image Processing, 2019

[Li *et al*., 2022] Hangyu Li, Nannan Wang, Xi Yang, Xiaoyu Wang, Xinbo Gao. Towards Semi-Supervised Deep Facial Expression Recognition with An Adaptive Confidence Margin. In CVPR, 2022.

[Lucey *et al*., 2010] Patrick Lucey, Jeffrey F. Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In CVPR, 2010.

[Marino *et al*., 2016] Kenneth Marino, Ruslan Salakhutdinov, Abhinav Gupta. The more you know: Using knowledge graphs for image classification. arXiv:1612.04844, 2016.

[Mollahosseini *et al*., 2017] Ali Mollahosseini, Behzad Hasani, Mohammad H Mahoor. Affectnet: A database for facial expression, valence, arousal computing in the wild. In IEEE Transactions on Affective Computing, 2019.

[Pham *et al*., 2021] Hieu Pham, Zihang Dai, Qizhe Xie, Quoc V. Le. Meta Pseudo Labels. In CVPR, 2021.

[Rizve *et al*., 2021] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. In ICLR, 2021.

[Sajjadi *et al*., 2016] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen.Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In NeurIPS, 2016.

[She *et al*., 2021] Jiahui She, Yibo Hu, Hailin Shi, Jun Wang, Qiu Shen, Tao Mei. Dive into ambiguity: Latent distribution mining and pairwise uncertainty estimation for facial expression recognition. In CVPR, 2021.

[Sohn *et al*., 2020] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In NeurIPS, 2020.

[Wang *et al*., 2020] Haibo Wang, Chuan Zhou, Xin Chen, Jia Wu, Shirui Pan, Jilong Wang. Graph stochastic neural networks for semi-supervised learning. In NeurIPS, 2020.

[Wang *et al*., 2020] Kai Wang, Xiaojiang Peng, Jianfei Yang, Debin Meng, Yu Qiao. Region attention networks for

pose and occlusion robust facial expression recognition. In IEEE Transactions on Image Processing, 2020.

[Wang *et al*., 2020] Kai Wang, Xiaojiang Peng, Jianfei Yang, Shijian Lu, and Yu Qiao. Suppressing uncertainties for large-scale facial expression recognition. In CVPR, 2020.

[Wang *et al*., 2022] Haobo Wang, Ruixuan Xiao, Yixuan Li, Lei Feng, Gang Niu, Gang Chen, Junbo Zhao. PICO: Contrastive Label Disambiguation for Partial Label Learning. In ICLR, 2022.

[Xie *et al*., 2020] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In NeurIPS, 2020.

[Xu *et al*., 2021] Yi Xu, Lei Shang, Jinxing Ye, Qi Qian, Y u-Feng Li, Baigui Sun, Hao Li, and Rong Jin. Dash: Semi-supervised learning with dynamic thresholding. In ICML, 2021.

[Xue *et al*., 2021] Xue, Fanglei, Qiangchang Wang, and Guodong Guo. Transfer: Learning relation-aware facial expression representations with transformers. In ICCV, 2021.

[Yang *et al*., 2023] Yujie Yang, Lin Hu, Chen Zu, Qizheng Zhou, Xi Wu, Jiliu Zhou, Yan Wang: Facial Expression Recognition with Contrastive Learning and Uncertainty-Guided Relabeling. In International Journal of Neural Systems, 2023.

[Zeng *et al*., 2018] Jiabei Zeng, Shiguang Shan, and Xilin Chen. Facial expression recognition with inconsistently annotated datasets. In ECCV, 2018.

[Zhang *et al*., 2016] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, Y u Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. In IEEE Signal Processing Letters, 2016.

[Zhang *et al*., 2021] Bowen Zhang, Yidong Wang, Wenxin Hou, HAO WU, Jindong Wang, Manabu Okumura, Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In NeurIPS, 2021.

[Zhao *et al*., 2011] Guoying Zhao, Xiaohua Huang, Matti Taini, Stan Z Li, and Matti PietikäInen. Facial expression recognition from near-infrared videos. Image and Vision Computing, 2011.