Comparative Study of Adversarial Training Methods for Long-tailed Classification

Xiangxian Li
Shandong University
xiangxian_lee@mail.sdu.edu.cn

Haokai Ma
Shandong University
mahaokai@mail.sdu.edu.cn

Lei Meng*
Shandong University
lmeng@sdu.edu.cn

Xiangxu Meng
Shandong University
mxx@sdu.edu.cn

ABSTRACT

Adversarial training is originated in image classification to address the problem of adversarial attacks, where an invisible perturbation in an image leads to a significant change in model decision. It recently has been observed to be effective in alleviating the long-tailed classification problem, where an imbalanced size of classes makes the model has much lower performance on small classes. However, existing methods typically focus on the methods to generate perturbations for data, while the contributions of different perturbations to long-tailed classification have not been well analyzed. To this end, this paper presents an investigation on the perturbation generation and incorporation components of existing adversarial training methods and proposes a taxonomy that defines these methods using three levels of components, in terms of information, methodology, and optimization. This taxonomy may serve as a design paradigm where an adversarial training algorithm can be created by combining different components in the taxonomy. A comparative study is conducted to verify the influence of each component in long-tailed classification. Experimental results on two benchmarking datasets show that a combination of statistical perturbations and hybrid optimization achieves a promising performance, and the gradient-based method typically improves the performance of both the head and tail classes. More importantly, it is verified that a reasonable combination of the components in our taxonomy may create an algorithm that outperforms the state-of-the-art.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning approaches**.

KEYWORDS

Long-tailed classification, Adversarial training, Taxonomy

*Corresponding author

ADVM '21, October 20, 2021, Virtual Event, China

ACM ISBN 978-1-4503-8672-2/21/10…$15.00
https://doi.org/10.1145/3475724.3483601

ACM Reference Format:
Xiangxian Li, Haokai Ma, Lei Meng, and Xiangxu Meng. 2021. Comparative Study of Adversarial Training Methods for Long-tailed Classification. In *Proceedings of the 1st International Workshop on Adversarial Learning for Multimedia (ADVM '21), October 20, 2021, Virtual Event, China.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3475724.3483601

1 INTRODUCTION

The phenomenon of long-tailed distribution[11], in which a small number of classes dominate the samples, is common in existing large-scale datasets due to the information imbalance of Internet data. Long-tailed data may incur biased learning for conventional machine learning algorithms where the algorithms will favor optimizing the prediction for head classes. This usually downgrade the classification performance for the tail classes. Existing long-tailed data training methods [1, 3, 7, 10, 12, 22] usually improve the model's prediction accuracy for tail classes at the cost of weakening the optimization of head classes, so the improvement of the overall classification accuracy is limited. Adversarial training is a new direction to alleviate the side-effect resulted by the long-tailed distribution. It is achieved by perturbing the input data or features [8, 14] for model [19], leading to an effect of data augmentation.

The main difference between existing adversarial training methods lies in their ways to generate and add perturbations. Commonly-used perturbation generation methods include stochastic normal distribution[17], statistical information[2], gradient[5, 8, 13, 19, 21], Generative Adversarial Networks (GAN) based methods[14]; while methods to introduce perturbations include data perturbation[5, 8, 13, 14, 19, 21] and feature perturbation[2, 17]. It is worth mentioning that existing methods usually make a trade-off between the model performance and robustness. For example, algorithms to go against adversarial attacks usually achieve a better robustness but a lower performance than the backbone model[19]. In addition, most of the methods are applied to balanced data, their performance on long-tailed data has not been fully verified. This leads to the need of a comparative study for existing adversarial training methods to verify their effectiveness in long-tailed classification.

To address the aforementioned problems, this paper decouples and reorganizes basic components of existing adversarial training methods, and proposes a taxonomy for long-tailed classification. The proposed taxonomy defines existing methods using three levels of key components, including the information, methodology, and optimization levels. Among them, information level categorizes the existing adversarial training methods into data perturbations

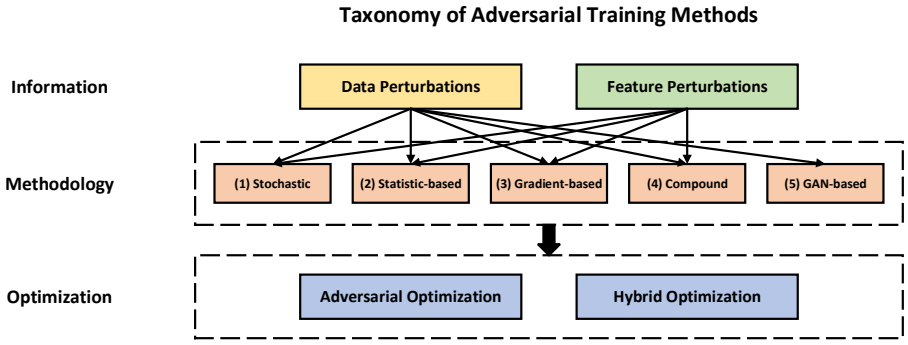

Figure 1: Illustration of the proposed taxonomy for adversarial training methods in long-tailed classification.

and feature perturbations based on their ways to introduce the perturbations; at the methodology level, the perturbation generation methods are classified into stochastic, statistic-based, gradient-based, compound, and GAN-based perturbation; and optimization level includes two methods termed adversarial optimization and hybrid optimization. This taxonomy may serve as a design paradigm for adversarial training methods, which covers all of the existing methods in the literature and offers the way to create new algorithms by combining the components listed in the three levels of the taxonomy. We verified the effectiveness of existing methods and their variants on the commonly used long-tailed datasets and conducted extensive experiments to evaluate the overall accuracy of a model and compare its performance over the head and tail classes. Through comparative experiments, the effectiveness of the combination of various perturbation adding methods and optimization methods is analyzed. In the case study, the law of the effects of various perturbations on the model performance is summarized. We also discussed the prospects of adversarial training in improving tail class imbalance in long-tailed classification problems and maintaining the overall prediction performance of the model.

To summarize, the main contributions of this paper include:

- A taxonomy is proposed to categorize the key components of adversarial training methods for long-tailed classification. This facilitates the development of novel algorithms by combining the three levels of key components in the taxonomy.
- Experimental verification is conducted on the effectiveness of existing adversarial training methods for long-tailed classification. Besides, the effects of the key components presented in the taxonomy are analyzed. These provide guidelines for the future development of adversarial training methods.

2 RELATED WORKS

2.1 Adversarial Training in Image Classification

Deep neural networks may produce errors in image classification due to "aberrations" caused by perturbations[4]. Adversarial training generates and optimizes various perturbations, thereby enhancing the robustness of models and achieving data augmentation to improve prediction performance.

Works of applying adversarial training in image classification mainly combine original data or features with perturbation generating methods. One common method is generating Gaussian-distribution-based noise directly or iteratively[5], and combine noise with data or features through linear combination[6, 17]. Statistic-based methods statistics distribution of dataset to generate more controllable perturbations[2]. Gradient-based perturbation generation is another approach which based on the gradient of model's prediction loss, usually combined with gradient ascent method based on confused classes[8], adjusting method[13], and attacking methods like the Fast Gradient Sign Method (FGSM) and Project Gradient Descent (PGD)[19][21]. GAN-based methods improve generator and discriminator by adversarial training, and the quality of sample generation is thus enhanced[14].

2.2 Long-tailed Image Classification

Long-tailed training methods are generally divided into re-sampling, re-weighting, and loss adjustment. Re-sampling methods reconstruct the data distribution by sampling algorithms, such as class-balanced re-sampling [3] , binary-branch training combining common and reverse sampling[22], retraining the classifier by balanced sampling[7]. Re-weighting assigns weights for classes to adjust predictions of the model. Curriculum learning[18], weights normalization[7] of classifier, logits adjustment[12], and applying expert models[16, 20] are effective re-weighting methods. Loss adjustment aims to improve traditional classification loss functions by altering items[1, 3, 10] to adapt models to long-tailed distribution.

Adversarial training is a new attempt in long-tailed classification. From the perspective of robustness, [19] introduce compound perturbations for adversarial training. At the perspective of accuracy, works like[2, 8, 17] take random or gradient-based perturbation to enhance the tail classes training. GAN-based method[14] generates new samples to alleviate the learning problem of tail classes.

3 TAXONOMY OF ADVERSARIAL TRAINING METHODS FOR LONG-TAILED CLASSIFICATION

The proposed taxonomy describes existing adversarial training methods in long-tailed classification using three levels of components , including information, methodology and optimization, as illustrated in Figure 1. The following sections present their details.

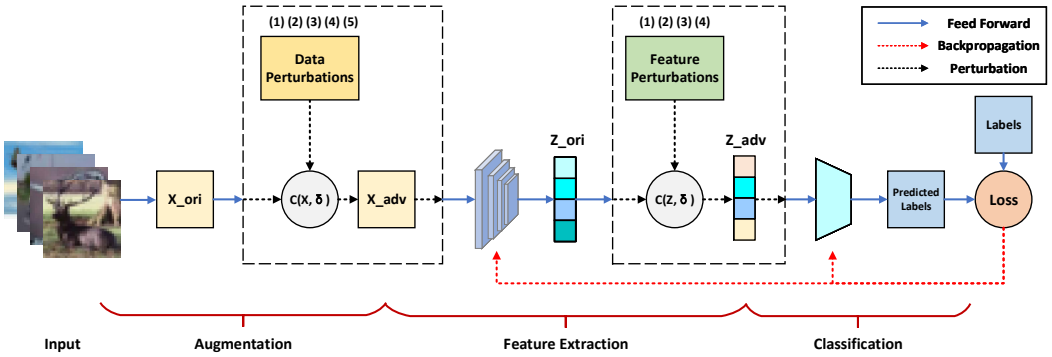

Figure 2: Illustration of the adversarial training paradigm with data and feature perturbations for long-tailed classification.

3.1 Inofrmation Level

Information level categories existing adversarial training methods into two classes, i.e., data perturbation and feature perturbation, based on their ways to introduce the perturbations, as illustrated in the following sections.

3.1.1 Data Perturbations. We illustrate the process of classification model training and indicate data perturbations using a black dotted frame in Figure 1. Data perturbation methods combine original data $\mathbf{X}_{ori}$ and perturbations δ into adversarial data $\mathbf{X}_{adv}$. The process of data perturbations is defined as:

$$\mathbf{X}_{adv} = C(\mathbf{X}_{ori}, \delta) \tag{1}$$

where $C(.,.)$ is the combination method such as linear combination. Existing works[8, 14, 19] perturb data on various generation methods, which enhances the randomness of the data and the model learning of the classification boundary.

3.1.2 Feature Perturbations. The perturbation on features usually refers to combine perturbations δ and the original features $\mathbf{Z}_{ori}$ that extracted by the backbone model $f(.)$ from original data $\mathbf{X}_{ori}$, i.e., $\mathbf{Z}_{ori} = f(\mathbf{X}_{ori})$. In Figure 2, we mark the feature perturbation in classification training using black dotted frame. The formula of generating adversarial features $\mathbf{Z}_{adv}$ is:

$$\mathbf{Z}_{adv} = C(\mathbf{Z}_{ori}, \delta) \tag{2}$$

where $C(.,.)$ is the combination method. The perturbation on features is implemented by introducing stochastic noise[17] or combining multi-classes features to augment the feature space[2].

3.2 Methodology Level

Methodology level summarizes the existing methods of generating perturbations. We have classified and numbered the existing methods in Figure 1 and mark them on Figure 2, which are (1) Stochastic, (2) Statistic-based, (3) Gradient-based, (4) Compound, and (5) GAN-based. Among them, (1), (2), (3), and (4) are commonly used for data and features, (5) is a unique method of data perturbations. Next, we will introduce them and the specific methods usually chosen.

3.2.1 Stochastic Perturbations. Stochastic perturbations are widely used to adversarial training. It introduces noises in a random way. The most commonly used stochastic perturbation δ_n is noise under Gaussian distribution, the formula is:

$$\delta_n \sim N(\mu, \sigma^2) \tag{3}$$

where $N(\mu, \sigma^2)$ is the Gaussian distribution with the mean μ and the variance σ^2. Usually, the standard normal distribution with $\mu = 0, \sigma = 1$ is used [17, 19], in consider of avoiding to introduce deviations to the mean while enhancing the variance of data/features.

3.2.2 Statistic-based Perturbations. Statistics-based perturbations also apply stochastic methods, but in a controllable form. Through statistical analysis of the overall or partial data/features, the mean, variance, etc. of the generated perturbations will be set within a specific range, the formula of Statistic-based perturbations δ_s is:

$$\delta_s = R(S(\mathbf{V})) \tag{4}$$

where $\mathbf{V}$ is data/features, $S(.)$ is the function of analysing data/features statistics, and $R(.)$ is a random generation method, which can be Gaussian distribution or random combination[2].

3.2.3 Gradient-based Perturbations. In addition to randomly generating perturbations, gradient-based method use the gradient of loss through model prediction and combination it with gradient algorithm to generate perturbations, interfere with the training progress of the models, as shown in formula:

$$\delta_g = A(\nabla\mathcal{L}(\hat{\mathbf{Y}}_{ori}, \mathbf{Y})) \tag{5}$$

where $\hat{\mathbf{Y}}$ is predictions, $\mathbf{Y}$ is labels, $\nabla\mathcal{L}(.,.)$ is the gradient of predicting loss, and $A(.)$ is the gradient algorithm, usually $A(.)$ can be easily as gradient ascent[8], or attack algorithm such as PGD[19].

3.2.4 Compound Perturbations. Compound perturbation combines different perturbations, such as stochastic perturbations and gradient-based perturbations[19]. First, random perturbation is used to generate perturbations for data/features, as shown in formula:

$$\mathbf{V}_{adv} = C(\mathbf{V}_{ori}, \delta_s) \tag{6}$$

where $\mathbf{V}_{adv}$ and $\mathbf{V}_{ori}$ represent adversarial and original data/features, δ_s is the stochastic perturbation, and $C(.,.)$ is the combination method. After generating the adversarial data/features, gradient-based methods are applied as:

$$\delta_{s+g} = A(\nabla\mathcal{L}(\hat{\mathbf{Y}}_{adv}, \mathbf{Y})) \tag{7}$$

which is similar to Equation (5), but where $\hat{\mathbf{Y}}_{adv}$ is predictions of adversarial data/features.

3.2.5 GAN-based Perturbations. GAN enhances the performance of discriminator and the confidence of samples generated by the generator through adversarial training. GAN introduces perturbations through generating samples. We succinctly mark it as:

$$\delta_{GAN} = GAN(\mathbf{X}, \mathbf{Y}) \tag{8}$$

where $GAN(.,.)$ is GAN, $\mathbf{X}$ is data and $\mathbf{Y}$ is labels. The work based on GAN generation[14] realizes the over-sampling of tail class in unbalanced classification by generating data.

3.3 Optimization Level

After the perturbations are generated by the methods and combined on the data or features, in the optimization level, the adversarial optimization optimizes adversarial data/features directly, and the hybrid optimization improve optimization by some decoupling and reconstructed optimization methods.

3.3.1 Adversarial Optimization. Adversarial optimization optimizes model with the adversarial data/features, as shown in Equation (9).

$$\mathcal{L}_{adv} = \mathcal{L}(\hat{\mathbf{Y}}_{adv}, \mathbf{Y}) \tag{9}$$

where $\mathcal{L}$ is the loss function, which is usually the Cross Entropy Loss, Cosine Loss[15] or their improved versions, $\hat{\mathbf{Y}}_{adv}$ is predictions of adversarial data/features, and $\mathbf{Y}$ is labels. Existing works [8], [9] directly optimize the perturbed or newly generated data/features.

3.3.2 Hybrid Optimization. Hybrid optimization method introduces original data/features to optimize on the basis of adversarial optimization, and improves the loss function accordingly, which will be introduced in turn.

Hybrid optimization by adversarial and original data/features. This hybrid method introduces original data/feature, and the original loss is shown as follow:

$$\mathcal{L}_{ori} = \mathcal{L}(\hat{\mathbf{Y}}_{ori}, \mathbf{Y}) \tag{10}$$

where $\hat{\mathbf{Y}}_{ori}$ is the predictions on original data/feature and $\mathbf{Y}$ represents labels. The original loss is linear combined to the loss of adversarial optimization:

$$\mathcal{L}_{adv+ori} = \mathcal{L}_{adv} + \mathcal{L}_{ori} \tag{11}$$

In addition to data expansion, it is also optimizes the model by two loss functions simultaneously.

Hybrid optimization by adversarial data/features with regularization. Based on adversarial optimization, regularization term is introducing to constrain the adversarial training process[19]:

$$\mathcal{L}_{reg} = \mathcal{R}(\hat{\mathbf{Y}}_{adv}) \tag{12}$$

where $\mathcal{R}$ is the regularization function and $\hat{\mathbf{Y}}_{adv}$ is the predictions of adversarial data/features. The regularization term constrains models to better find the optimization direction on the basis of adversarial optimization. This term is added as:

$$\mathcal{L}_{adv+reg} = \mathcal{L}_{adv} + \mathcal{L}_{reg} \tag{13}$$

Hybrid optimization by adversarial and original data/features with regularization. The regularization term can be added to original and adversarial optimization, i.e., merge Equation (11) and (13):

$$\mathcal{L}_{adv+reg} = \mathcal{L}_{adv} + \mathcal{L}_{ori} + \mathcal{L}_{reg} \tag{14}$$

Table 1: Statistics of the datasets used in the experiments.

Datasets	Imbalance Ratio	#Classes	#Training	#Testing
CIFAR 10-LT	0.1	10	20,431	10,000
CIFAR 100-LT	0.1	100	19,573	10,000

4 EXPERIMENTS

4.1 Experimental Setup

4.1.1 Datasets. We use two benchmarking datasets CIFAR 10-LT and CIFAR 100-LT that are commonly used in long-tailed classification for experiments. Their statistics are showing in Table 1.

CIFAR 10-LT is a subset sampled from the CIFAR-10 dataset[9] which contains 60,000 images in 10 classes evenly. We adopt the method of Cao et al.[1] to construct the training set of CIFAR 10-LT. Imbalance Ratio(IR) ρ denotes the ratio of sample sizes n of the most sampled class and the least sampled class, i.e., $\rho = max_i n_i / min_i n_i$. Then using exponential decay to compute the sample sizes for other classes in order. In our experiments, we set $\rho = 0.1$, i.e., the number of training samples is from 500 to 5,000, and the testing samples of CIFAR 10-LT is the same as the CIFAR-10.

CIFAR 100-LT is constructed from the CIFAR-100[9] which contains 60,000 images in 100 classes evenly. We set $\rho = 0.1$ so the number of training samples in CIFAR 100-LT is from 50 to 500 and the testing set is the same as the CIFAR-100.

4.1.2 Evaluation Measures. In the experiment, we calculated the Top-1 Accuracy and measured the performance of various methods in the classification task, the equation of Accuracy is:

$$Accuracy = (TP + TN)/(P + N) \tag{15}$$

Where TP, TN, P, and N are True Positives, True Negatives, Positives, and Negatives. In order to study the effect of methods in the long-tailed classification, we equally divided classes into head classes and tail classes[20], i.e., for the CIFAR 10-LT dataset, classes 1-5 is the head and classes 6-10 is the tail; for the CIFAR 100-LT dataset, the head is classes 1-50 and the tail is classes 51-100. After that, we calculate the average of the Accuracy in the head and tail.

4.1.3 Implementation Details. Following the work of Cao et al.[1], we use the ResNet-32 as the backbone of classification models. The model of backbone without adversarial method is chosen as our baseline model. The SGD optimizer with momentum of 0.9 is adopted, and the learning rate ranges from 0.01 to 0.3. The weight decay of SGD is selected from $\{0.001, 0.0005, 0.0001\}$. We set the batch size to be 128, and models are trained by 100 epochs, on epoch 30 and 60, the learning rate is decayed by 0.1 and 0.01.

4.2 Performance Comparison

In this section, we analyse results of the comparative experiment. We first experimented methods of perturbing data with noises that based on standard normal distribution (White Noise) and statistic (Statistical Noise). We set the coefficient of controlling the noise size between 0.05-1. We implement RoBal[19] using the settings of RoBal-N, BLT[8] which perturb on data and a lite version that constructing of DGC[17] which generate perturbations on features.

We combine methods with different optimization strategies, the basic components of which are optimization by adversarial data/features (Adv), optimization by original data/features(Ori),

Table 2: Comparison results of existing adversarial methods and their extension methods on the CIFAR 10-LT and CIFAR 100-LT datasets. Methods are divided into three levels according to the proposed taxonomy. (All: Top-1 Accuracy of all classes, Head: average Accuracy of head classes, Tail: average Accuracy of tail classes).

Information	Methodology	Optimization	CIFAR 10-LT			CIFAR 100-LT		
			All	Head	Tail	All	Head	Tail
Data Perturbation	Stochastic	Adv (White Noise)	85.18	87.84	82.52	54.28	64.96	43.60
		Adv+Ori (White Noise)	85.27	88.22	82.32	54.37	65.14	43.60
		Adv+KL (White Noise)	84.80	87.54	82.06	54.81	65.26	44.36
		Adv+Ori+KL (White Noise)	84.94	88.04	81.84	55.08	65.22	44.94
	Statistic-based	Adv (Statistical Noise)	85.10	88.16	82.03	55.58	66.39	44.76
		Adv+Ori (Statistical Noise)	85.24	88.42	82.06	55.38	66.54	44.22
		Adv+KL (Statistical Noise)	**86.10**	88.04	**84.16**	54.86	66.26	43.46
		Adv+Ori+KL (Statistical Noise)	85.82	88.66	82.98	**56.00**	66.36	**45.64**
	Gradient-based	Adv (BLT[8])	85.61	**88.98**	82.24	54.70	**66.64**	42.76
		Adv+Ori (BLT)	85.03	88.54	81.52	54.35	66.20	42.50
	Compound	Adv (RoBal[19])	72.28	66.58	77.98	30.32	32.96	27.68
		Adv+Ori (RoBal)	80.57	80.00	81.14	42.71	52.42	33.00
		Adv+KL (RoBal)	77.89	78.30	77.48	43.64	49.82	37.46
		Adv+Ori+KL (RoBal)	80.00	79.76	80.24	46.02	54.10	37.94
Feature Perturbation	Stochastic	Adv (DGC[17])	83.58	86.64	80.52	50.54	60.74	40.34
		Adv+Ori (DGC)	84.93	87.92	81.94	53.78	64.12	43.44
ResNet-32 (Backbone)			84.92	88.14	81.70	54.01	63.96	44.06

and optimization by loss with KL Divergence(KL). The ratio of adversarial data/feature optimization loss and original data/feature optimization loss is 0.1-2, and the coefficient of KL term ranges from 1-20. Results are reported in Table 2. We can observe the followings:

- Compared with the baseline model, some adversarial training methods improve the prediction on the tai while maintaining or even elevating the accuracy of the head, such as Statistical Noise with Adv+Ori+KL and BLT with Adv. This shows that the recombining methods may alleviate the problem of reducing the head in traditional long-tailed classification methods.
- Experiments in the CIFAR 10-LT dataset using Adv+Ori or Adv+KL optimization generally achieve better results than that adding Adv only; and in the CIFAR 100-LT dataset where the classification is more complicated, applying Adv+Ori+KL hybrid optimization for further optimization can achieve better predictions.
- Statistical Noise generally gets better performance than White Noise. It uses the mean of the dataset to limit the range of noise generation, so that data perturbations are in a more controllable range. More controllable perturbations can also be better combined with various optimization methods as shown in results.
- RoBal with Adv optimization will make the head prediction drop significantly. It may be caused by the distorting data distribution and the loss function. In addition to adding KL items, the problem can also be alleviate by Ori optimization.
- BLT adds the augmented tail data to each original batch. In the Adv+Ori optimization, original data downgrade the overall improvement compared with Adv and reflect in the head and tail. This shows that the ratio of adversarial and original data is important to algorithm designing.
- The comparison of the DGC method with Adv and Adv+Ori shows that introducing original data/feature constraints may have a good effect on the feature perturbations method.

4.3 Comparative Study on Performance of Head and Tail Classes

In this section, we analyse models that have good performance in experiments. Taking the CIFAR 10-LT as an example, we report the specific accuracy of each class in detail, as show in Table 3.

The ResNet-32 (Backbone) model is set as a baseline. The predictions of tail classes in models are generally lower than that in head classes. From the perspective of class accuracy, the best prediction is usually in the Class 1 or the Class 2; but the prediction accuracy does not only depend on the number of samples in the class, Class 4 in the head and Class 6 in the tail are the two worst predicted classes, We will discuss that in Section 4.4.

The RoBal (Adv+Ori) model achieves an improvement of some classes in the tail at the cost of head predictions drop. The BLT (Adv) model has obvious optimization of the head. Since BLT introduce gradient-based perturbations, the process of creating hard samples lacks randomness compared with noise based methods. When only using Adv optimization, the improvement of BLT in Class 4 and Class 6 is limited. After using Adv+Ori optimization, the number of head samples has been greatly expanded, which leads to Class 4 improved, but the prediction of Class 6 is still poor. The White Noise (Adv+Ori) model and the Statistical Noise (Adv+KL) model introduce noise and perturb the data distribution globally, so that the tail and the poorly classified classes have a chance to be improved. Statistical Noise model combines Adv+KL or Adv+Ori+KL enhance the constrain, so reducing the downgrading to the head.

The results shows that the introduction of noise can help improve the performance of tail and poorly classified classes in long-tailed classification. At the same time, combined with compound optimization methods can maintain the prediction performance of head classes. On this basis, the use of gradient perturbation for batch enhancement may be able to further improve the head.

4.4 Effects of Adversarial Training on Latent Embeddings

As we discuss in Section 4.3, Class 4 (cat) and Class 6 (dog) are hard to classify. To study the optimization of methods for predicting indistinguishable classes, we use t-SNE to visualize the latent embeddings of ResNet-32 (Backbone) model, Statistical Noise (Adv+Ori+KL) model and Statistical Noise (Adv+KL) model, as shown in Figure 3.

Compared with the ResNet-32 (Backbone) model, the Statistical Noise (Adv+KL) model creates an offset and rotation in the feature space, but at the same time, the two classes are more dispersed

Table 3: The accuracy of the effective method on the CIFAR 10-LT dataset in each class (Class n: Top-1 Accuracy of class n, Head: average Accuracy of head classes, Tail: average Accuracy of tail classes) Comparison between the best-performing algorithms on the CIFAR 10-LT dataset in terms of classes (Class n: Top-1 Accuracy of the n-th class, Head: average Accuracy of head classes (class1 - class5), Tail: average Accuracy of tail classes (class6-class10))

Model	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10	Head	Tail
ResNet-32 (Backbone)	95.40	98.10	86.10	75.60	85.50	76.30	84.40	83.70	83.50	80.60	88.14	81.70
White Noise (Adv+Ori)	95.60	96.70	84.90	**78.90**	85.00	76.60	85.00	**83.90**	82.50	83.60	88.22	82.32
Statistical Noise (Adv+KL)	94.00	97.40	85.70	78.20	84.90	**80.80**	86.90	83.10	**86.80**	83.20	88.04	**84.16**
RoBal (Adv+Ori)	92.00	95.00	72.20	68.90	71.90	69.70	**89.30**	79.00	81.30	**86.40**	80.00	81.14
BLT (Adv)	**96.10**	**98.50**	**86.30**	75.50	**88.50**	76.90	87.20	81.40	84.60	81.10	**88.98**	82.24
DGC (Adv+Ori)	93.80	97.80	86.00	74.80	87.20	77.80	86.70	79.30	83.40	82.50	87.92	81.94

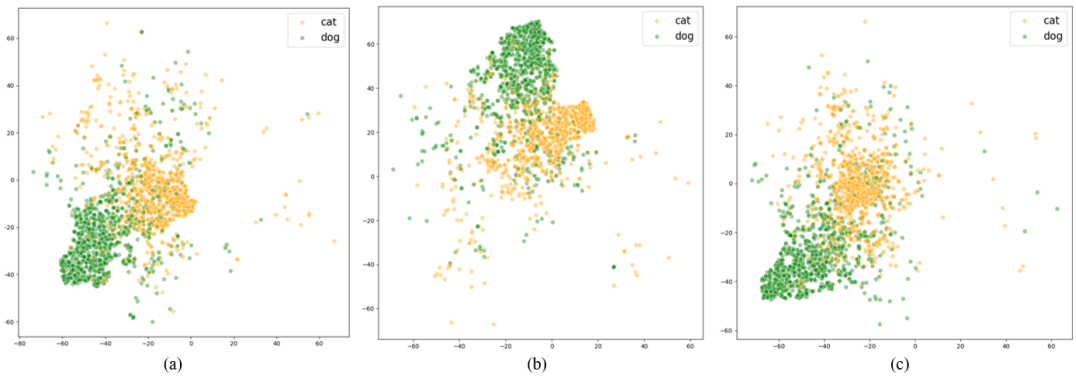

Figure 3: Visualization of latent embeddings in 2-D space. (a) ResNet-32 (Backbone);(b) Statistical Noise (Adv+KL);(c) Statistical Noise (Adv+Ori+KL).

and the Class dog is clustered more closely, which improves the predictions in two classes. The range of spatial distribution of the Statistical Noise (Adv+Ori+KL) model is basically the same, but it makes the Class cat more concentrated and far away from the Class dog, so it is more obvious to improve the accuracy of Class cat.

According to this study, it can be found that adversarial training makes the latent embedding space change significantly, and at the same time it may get some indistinguishable classes separated, the Ori optimization could be applied to constrain this change.

4.5 Discussion

As observed from the performance comparison in Section 4.2, the state-of-the-art algorithms may be further improved with a reasonable replacement of their components as listed in our taxonomy. Interestingly, as shown in Table 2, different generation methods for perturbation favor different optimization components. For example, Adv+Ori optimization leads to a significant increase in performance of RoBal while decreasing that of BLT. The incorporating of KL divergence harms the performance of using white noise, but leads to better performance for the algorithm using statistical perturbations.

In Table 3, using white noise or statistical noise may better improve the performance of tail classes while remaining that of head classes than some state-of-the-art algorithms. Existing algorithms, such as gradient-based BLT, outperform baseline over both the head and tail classes. This illustrates that types of perturbations may influence the representation learning for the long-tailed data. We also studied the effect of optimization by visualization. For models in Figure 3, Statistical Noise (Adv+KL) introduces transformation of latent embeddings, and after adding Ori optimization, the latent embeddings are restored while the characteristics of perturbations are retained. The constraint effects and combination methods of different optimization can be adopted in various scenes.

5 CONCLUSION

This paper presents a taxonomy of adversarial training for long-tailed classification, which defines existing methods in three levels of components, including information, methodology, and optimization. Serving as a design paradigm, different adversarial training algorithms can be created based on the developed taxonomy. Experimental findings verified that the state-of-the-art algorithms can be further improved by replacing them with the components in our taxonomy. Extensive case studies illustrate that such improvement is achieved by a reasonable combination of different perturbation generation and incorporation components in the information, methodology and optimization levels.

Despite the encouraging results, there is still a number of issues to be further explored. First, the gains of performance brought by different perturbations need to be analysed. Secondly, furthering being designed methods of reorganization of perturbations and optimizations are critical. The comparative experiments of different perturbation generation methods demonstrate the significance of enhancing the randomness of data/features; in optimization, we can consider performing hybrid or multi-branch algorithms based on different targets such as optimizing tail classes and difficult classes.

ACKNOWLEDGMENTS

This work is supported in part by the National Natural Science Foundation of China (Grant no. 62006141) and the Special Project of Science and Technology Innovation Base of Key Laboratory of Shandong Province for Software Engineering (Project ID:11480004042015).

REFERENCES

[1] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. 2019. Learning imbalanced datasets with label-distribution-aware margin loss. *arXiv preprint arXiv:1906.07413* (2019).

[2] Peng Chu, Xiao Bian, Shaopeng Liu, and Haibin Ling. 2020. Feature Space Augmentation for Long-Tailed Data. In *European Conf. on Computer Vision (ECCV)*.

[3] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9268–9277.

[4] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Mądry. 2019. Adversarial examples are not bugs, they are features. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. 125–136.

[5] Yunseok Jang, Tianchen Zhao, Seunghoon Hong, and Honglak Lee. 2019. Adversarial defense via learning to generate diverse attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2740–2749.

[6] Haoming Jiang, Zhehui Chen, Yuyang Shi, Bo Dai, and Tuo Zhao. 2018. Learning to defense by learning to attack. *arXiv preprint arXiv:1811.01213* (2018).

[7] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. 2019. Decoupling Representation and Classifier for Long-Tailed Recognition. In *International Conference on Learning Representations*.

[8] Jedrzej Kozerawski, Victor Fragoso, Nikolaos Karianakis, Gaurav Mittal, Matthew Turk, and Mei Chen. 2020. BLT: Balancing Long-Tailed Datasets with Adversarially-Perturbed Images. In *Proceedings of the Asian Conference on Computer Vision*.

[9] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).

[10] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*. 2980–2988.

[11] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. 2019. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2537–2546.

[12] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. 2020. Long-tail learning via logit adjustment. *arXiv preprint arXiv:2007.07314* (2020).

[13] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. 2017. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1765–1773.

[14] Sankha Subhra Mullick, Shounak Datta, and Swagatam Das. 2019. Generative adversarial minority oversampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1695–1704.

[15] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. 2018. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5265–5274.

[16] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X Yu. 2020. Long-tailed recognition by routing diverse distribution-aware experts. *arXiv preprint arXiv:2010.01809* (2020).

[17] Xinyue Wang, Yilin Lyu, and Liping Jing. 2020. Deep Generative Model for Robust Imbalance Classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14124–14133.

[18] Yiru Wang, Weihao Gan, Jie Yang, Wei Wu, and Junjie Yan. 2019. Dynamic curriculum learning for imbalanced data classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5017–5026.

[19] Tong Wu, Ziwei Liu, Qingqiu Huang, Yu Wang, and Dahua Lin. 2021. Adversarial Robustness under Long-Tailed Distribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8659–8668.

[20] Liuyu Xiang, Guiguang Ding, and Jungong Han. 2020. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *European Conference on Computer Vision*. Springer, 247–263.

[21] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. 2019. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*. PMLR, 7472–7482.

[22] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. 2020. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9719–9728.