Imagine by Reasoning: A Reasoning-Based Implicit Semantic Data Augmentation for Long-Tailed Classification

Xiaohua Chen[1,2], **Yucan Zhou**[1*], **Dayan Wu**[1],
Wanqian Zhang[1], **Yu Zhou**[1], **Bo Li**[1,2], **Weiping Wang**[1,2]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{chenxiaohua, zhouyucan, wudayan, zhangwanqian, zhouyu, libo, wangweiping}@iie.ac.cn

Abstract

Real-world data often follows a long-tailed distribution, which makes the performance of existing classification algorithms degrade heavily. A key issue is that samples in tail categories fail to depict their intra-class diversity. Humans can imagine a sample in new poses, scenes, and view angles with their prior knowledge even if it is the first time to see this category. Inspired by this, we propose a novel reasoning-based implicit semantic data augmentation method to borrow transformation directions from other classes. Since the covariance matrix of each category represents the feature transformation directions, we can sample new directions from similar categories to generate definitely different instances. Specifically, the long-tailed distributed data is first adopted to train a backbone and a classifier. Then, a covariance matrix for each category is estimated, and a knowledge graph is constructed to store the relations of any two categories. Finally, tail samples are adaptively enhanced via propagating information from all the similar categories in the knowledge graph. Experimental results on CIFAR-100-LT, ImageNet-LT, and iNaturalist 2018 have demonstrated the effectiveness of our proposed method compared with the state-of-the-art methods.

Introduction

Deep neural networks have achieved dramatic performance when high-quality annotated and balance-distributed training data is provided (Krizhevsky, Sutskever, and Hinton 2012; He et al. 2016; Huang et al. 2017). However, in real-world scenarios, data is usually long-tail distributed, where many tail categories occupy a small number of samples and most samples belong to a few head categories. It is a great challenge to train deep models on long-tail distributed data. On the one hand, the separating hyperplane will be heavily skewed to the tail classes because of their weak statistical ability. More importantly, tail classes are easily overfitted as their samples fail to describe their intra-class diversity.

To strengthen the effect of tail classes in training, a typical and intuitive strategy is re-balancing, including data re-sampling and loss re-weighting. Data re-sampling achieves training fairness by balancing the data distribution through sampling, while loss re-weighting pays more attention to tail
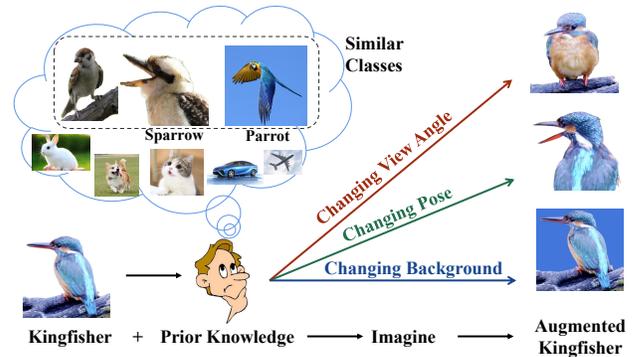

Figure 1: An illustration of imagining new samples by reasoning. people can imagine a kingfisher in various view angles, poses, and backgrounds with their prior knowledge.

categories by assigning higher penalties to the loss of tail classes. Although these methods can alleviate hyperplane skewness, over-fitting on tail categories is still a big issue because of the limited intra-class diversity.

Data augmentation is an efficient way to enrich tail categories, where cropping, rotation, mixup, and generative adversarial network are adopted to generate new samples (Simonyan and Zisserman 2015; He et al. 2016; Ratner et al. 2017; Bowles et al. 2018). Nevertheless, with the increasing number of augmented examples, the training speed is sharply slowed down. Fortunately, ISDA (Wang et al. 2019) propose an instance-based implicit data augmentation method to reduce the computation cost, where new instances can be generated by changing the original instance to semantic transformation directions sampled from the feature covariance matrix. However, for tail classes, it is incapable to estimate a diversified covariance matrix (Li et al. 2021).

We argue that humans can imagine a rare animal in different poses, colors, and backgrounds with prior knowledge, though we have only seen one picture of this animal. For instance (shown in Figure **??**), as "Kingfisher" is similar to "Sparrow" in humans' prior knowledge, we can easily imagine a "Kingfisher" with an open mouth when we have seen a "Sparrow" with an open mouth. Inspired by this, tail classes can borrow semantic transformations from other classes.

In this paper, in order to mimic the human reasoning pro-

---

*Yucan Zhou is the corresponding author

cess, we propose an effective Reasoning-Based Implicit Semantic Data Augmentation (**RISDA**) method for long-tailed classification. It enriches a tail instance by transferring semantic transformation directions from similar categories to expand the intra-class diversity. Specifically, we first train a network with the long-tail distributed data. Then, a covariance matrix for each category is estimated, which represents all the possible transformation directions of a category. After that, for each tail category whose covariance matrix is limited, we enhance it with similar categories. To recognize similar categories, a knowledge graph is constructed, where each non-diagonal element indicates the similarity of two categories. Finally, sufficient instances can be generated by augmenting a tail instance via propagating transformation directions from similar categories. Nevertheless, to ensure the tail instance contains all the features to transfer, we also complement it with features from similar categories. Therefore, our method can generate instances that are definitely different from the training data. The contributions of this paper are summarized as follows.

- We propose a reasoning-based implicit data augmentation method, which transfers semantic transformation directions from other classes to enhance tail classes. It can largely enrich the intra-class diversity for tail categories. To the best of our knowledge, this is the first time to transfer feature transformations from similar categories for augmentation.

- We construct a learnable knowledge graph so that our method can adaptively select similar categories for different samples to make the reasonable transformation.

- The proposed RISDA[1] outperforms current state-of-the-art methods on long-tailed datasets, such as CIFAR-100-LT, ImageNet-LT, and iNaturalist 2018.

Related Work

Re-Balance

A common re-balance strategy is re-sampling, which aims to achieve a balanced data distribution, including over-sampling (Buda, Maki, and Mazurowski 2018; Byrd and Lipton 2019) and under-sampling (He and Garcia 2009; Buda, Maki, and Mazurowski 2018). However, over-sampling may cause over-fitting by duplicating tail data, while under-sampling will damage feature representation when abandoning head data. Recently, some decoupling methods are developed, which adopt different sampling strategies in representation learning and classifier training(Kang et al. 2020; Zhou et al. 2020; Wang et al. 2021).

Another strategy is re-weighting the loss to give more attention to tail classes. An intuitive way is to re-weight the loss of each class inversely proportional to the number of samples (Huang et al. 2016). Then, class-balance loss (Cui et al. 2019) is proposed to emphasize the effective number of samples. Thereafter, meta-class-weight (Jamal et al. 2020) estimates class weights by meta-learning, while LDAM (Cao et al. 2019) uses class-level re-weighting optimization schedule to train a label-distribution-aware loss.

---

[1]Code is available at https://github.com/xiaohua-chen/RISDA

In addition, some fine-grained instance-level re-weighting methods are studied. Such as Focal Loss (Lin et al. 2017) reduces the weights of easy samples to make the model focus on hard samples in training. L2RW (Ren et al. 2018) and meta-weight-net (Shu et al. 2019) assign instance-wise weights on the gradient direction.

Although re-balance methods have achieved great improvements, the intra-class diversity for tail categories is still limited, making them easily over-fitted.

Data Augmentation

To enhance the intra-class diversity for tail categories, data augmentation methods are explored. A direct way is to generate samples by combining other samples in a tail class linearly (He et al. 2008; Zhang et al. 2018). However, the combination may be meaningless. To address this issue, GAN (Bowles et al. 2018) is introduced. The above methods are explicit data augmentation methods. With the increasing number of samples, the convergence speed is sharply slowed down. ISDA (Wang et al. 2019) is a great way to handle this problem, which uses the class-wise covariance matrix and instance-wise feature to formulate a Gaussian distribution, thus infinite samples can be generated. But ISDA fails for tail classes where the insufficient samples are incapable of estimating a diversified covariance matrix. MetaAug (Li et al. 2021) tries to find an optimal covariance matrix that generalizes well on unseen samples. However, it still suffers from limited diversity for tail classes as the covariance matrix is calculated with samples at hand.

Fortunately, tail classes can borrow information from other categories (Chu et al. 2020; Liu et al. 2020; Xiao et al. 2021). As head categories are sufficiently diversified, some methods propose to transfer knowledge from head to tail. For example, Liu et al. model each category into a "feature cloud" and expand tail categories by transferring the intra-class angular distribution from head categories (Liu et al. 2020). Chu et al. generate new high-level features for tail categories by merging class-specific features of the head categories and class-generic features of the tail categories (Chu et al. 2020). However, the "head to tail" is defined by the number of samples, transferring variations based on this relation may be meaningless. For example, if "Apple" is a head class, and "Kingfisher" is a tail class, then transferring color variations from "Apple" to "Kingfisher" is unreasonable.

Our method follows the idea of implicit data augmentation and tries to borrow variations from other categories to enrich tail categories. However, different from other methods, we transfer variations from all the related classes in a knowledge graph other than the head categories.

The Proposed Method

A classifier usually performs worse for tail classes. To enhance their diversity, we propose to borrow variations from other categories as shown in Figure 2.

In stage-I, we use all the samples to train a feature subnetwork and a classifier. Subsequently, we use the extracted features to calculate the covariance matrix and the prototype for each category, where the covariance matrix represents all the
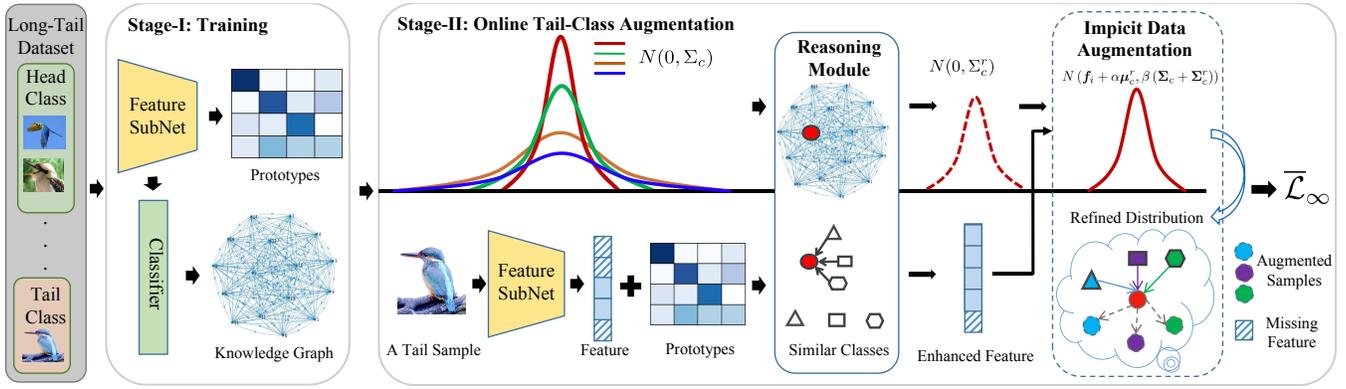
Figure 2: The framework of RISDA. In Stage-I, all training data is used to train a feature subnet and a classifier. Subsequently, we calculate the covariance matrix and the prototype for each category with the feature extracted with the subnet. Then, the classifier is used to construct a knowledge graph. In Stage-II, we augment each tail instance with a semantic transformation direction sampled from the covariance matrix. For tail classes, the covariance matrix is limited, so we refine it with similar classes defined in the knowledge graph. What's more, to ensure the tail instance contains all the features to transfer, we complement it with the prototypes from similar categories. Finally, unlimited instances can be sampled to calculate the $\overline{\mathcal{L}}_\infty$.

feature semantic transformation directions of each category, and the prototype contains its specific features. Then, the classifier is adopted to construct a knowledge graph, which stores the similarity of any two categories. In stage-II, a tail sample is first propagated to the feature subnetwork to get semantic features. Then, we aim to augment it with the semantic transformations sampled from the covariance matrix of its category. However, the covariance matrix of tail categories is limited, so we refine it with similar classes defined in the knowledge graph. What's more, the tail instance may miss some features to transfer. Thus, to ensure it contains all the features, we complement it with the features from similar categories. Consequently, unlimited instances can be generated by changing an instance with infinite semantic transformation directions from the refined distribution. Finally, an upper bound of the CE loss on unlimited data is derived to fine-tune the network.

In the following, we will describe how a tail sample is implicitly augmented by reasoning, and the loss function to fine-tune the network with unlimited generated data.

Implicit Data Augmentation

Given a long-tail distributed dataset $\{x_i, y_i\}_{i=1}^N$, where $y_i \in L = \{l_1, l_2, ..., l_C\}$, C is the number of categories. Inspired by the observation that instance-balanced sampling learns better generalizable representations (Kang et al. 2020), we use the original distribution to train a feature extractor F and a classifier H in Stage-I.

For a sample x_i in a tail class l_c, we extract its feature with $\boldsymbol{f}_i = F(x_i)$. Inspired by ISDA, we can randomly sample along $N(0, \beta\boldsymbol{\Sigma}_c)$ to generate features with different semantic transformations, i.e., $\tilde{\boldsymbol{f}}_i \sim N(\boldsymbol{f}_i, \beta\boldsymbol{\Sigma}_c)$.

As the semantic transformation directions of different categories are different, we need to compute a covariance matrix for each category. To get the covariance matrix, we first need to calculate each class prototype by averaging features

of all the samples in a class, as shown in Equation (1):

$$\boldsymbol{\mu}_c = \frac{1}{N_c} \sum_{i=1}^{N_c} \boldsymbol{f}_i, \tag{1}$$

where $c \in [1, C]$, N_c is the number of samples in class l_c, and $\boldsymbol{f}_i$ represents feature of the i-th sample in l_c. Secondly, we calculate each element in a covariance matrix with:

$$\boldsymbol{\Sigma}_c(m, n) = \frac{\boldsymbol{\Sigma}_{i=1}^{N_c} (\boldsymbol{f}_i^m - \boldsymbol{\mu}_c^m)(\boldsymbol{f}_i^n - \boldsymbol{\mu}_c^n)}{N_c - 1}. \tag{2}$$

After that, the covariance matrixes of all classes $\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \cdots, \boldsymbol{\Sigma}_C\}$ can be obtained. However, covariance matrixes for tail classes contain limited variations. To alleviate this issue, we propose a novel reasoning module to transfer plentiful variations from other classes.

Reasoning-Based Implicit Data Augmentation

In this section, we elaborate on the process of transferring knowledge to augment a tail sample. With the constructed knowledge graph, we can enrich the covariance matrixes of tail classes by reasoning.

Knowledge Graph Construction   Current methods usually transfer variations from head classes to the tail ones. Instead, our approach tries to transfer knowledge from similar categories. To achieve this, we use the classifier H to construct a knowledge graph, which is built with the confusion matrix on the training data. Actually, it is a category-to-category directed graph $\langle V, \boldsymbol{\varepsilon} \rangle$, where V are category nodes and $\boldsymbol{\varepsilon} \in R^{C \times C}$. Each element ε_{ij} represents the similarity of l_i and l_j, which can be calculated as follows:

$$\varepsilon_{ij} = \frac{\sum_{i=1}^{N_i} \mathbb{1}(H(\boldsymbol{f}_i) = l_j \wedge y_i = l_i)}{N_i}, \tag{3}$$

where $\mathbb{1}$ is the indicator function. ε_{ij} represents the ratio of the number of samples in l_i predicted into l_j.

Reasoning-Based Semantic Transformation We propose to refine the covariance matrix of each tail class with those of its similar classes. Intuitively, different categories are similar because they have similar appearances or living environments. For example, because "Kingfisher" and "Parrot" are very similar, when we see a flying "Parrot", we can imagine a flying "Kingfisher" even if we haven't seen a flying "Kingfisher". Therefore, we can infer a better intra-class covariance for tail classes by transferring semantic transformations from similar classes:

$$\Sigma_c^r = \sum_{i=1, i \neq c}^{C} \varepsilon_{c,i} \Sigma_i. \tag{4}$$

Ideally, we need to compute the reasoning-based class-conditional covariance matrix with entire training samples in each epoch. It is cost-expensive. So we use an online process to update and transfer covariance matrix batch by batch.

With the reasoning-based covariance matrix, we can change an instance with a more diversified semantic direction. However, an instance may miss some features that are necessary for a reasoning-based transformation direction. For example, only when the mouth is detected in a "Kingfisher", we can change the pose of its mouth. Therefore, to ensure a tail instance contains all the features to transfer, we complement it with features from similar categories:

$$\mu_c^r = \sum_{i=1, i \neq c}^{C} \varepsilon_{c,i} \mu_i. \tag{5}$$

After the feature distribution of each tail sample is refined by a reasoning prototype and a reasoning covariance matrix. Then, f_i can perform various semantic transformations along the random directions sampled from $N\left(\alpha\mu_c^r, \beta\left(\Sigma_c + \Sigma_c^r\right)\right)$ and generate the augmented feature $\tilde{f}_i$ during training with Equation (6):

$$\tilde{f}_i \sim N\left(f_i + \alpha\mu_c^r, \beta\left(\Sigma_c + \Sigma_c^r\right)\right), \tag{6}$$

where α and β are positive coefficients to control the strength of reasoning-based semantic data augmentation. In the experiment, both α and β are decayed by t/T, where t and T represent the current number of epochs and the number of total epochs, respectively.

Optimization

With our data augmentation method, a straightforward way to train a classifier is to generate samples for tail classes with Equation (6) untill they have comparable samples with head classes. Assume each sample in tail class l_c is augmented M times, then we can obtain a new data set $\left\{\left(f_i^1, y_i\right), \left(f_i^2, y_i\right), ..., \left(f_i^M, y_i\right)\right\}_{i=1}^{N_c}$, where f_i^k sampled from $\tilde{f}_i$ is the k-th augmented feature of f_i.

Thereafter, we can use the traditional cross-entropy loss to train a classifier:

$$\mathcal{L}_{\mathcal{M}}(\theta_F, W, b) =$$

$$\sum_{c \in TC} \frac{1}{N_c} \sum_{i=1}^{N_c} \frac{1}{M} \sum_{k=1}^{M} -\log\left(\frac{e^{w_{y_i}^T f_i^k + b_{y_i}}}{\sum_{j=1}^{C} e^{w_j^T f_i^k + b_j}}\right) \tag{7}$$

$$+ \sum_{c \in HC} \frac{1}{N_c} \sum_{i=1}^{N_c} -\log\left(\frac{e^{w_{y_i}^T f_i + b_{y_i}}}{\sum_{j=1}^{C} e^{w_j^T f_i + b_j}}\right),$$

where TC represents the set of tail classes, HC is the set of head classes, $W = [w_1, w_2, ..., w_C]^T$ and $b = [b_1, b_2, ..., b_C]^T$ are the weight matrixes and biases corresponding to the last fully connected layer, respectively.

However, for the tail category, when sampling M times, the sampling variance is unstable and limited. An ideal way is to generate as much data as possible. But the increasing M can lead to additional calculations. To simplify computation while generating more data, we plan to implicitly generate unlimited features. When M is close to infinity, we can derive an easy upper-bound loss following the Law of Large Numbers. In detail, when $M \to \infty$, we take into account all possible enhanced samples, and then the loss for tail categories can be defined as:

$$\mathcal{L}_{\mathcal{M} \to \infty} = \sum_{c \in TC} \frac{1}{N_c} \sum_{i=1}^{N_c} E_{\tilde{f}_i}\left[-\log\left(\frac{e^{w_{y_i}^T \tilde{f}_i + b_{y_i}}}{\sum_{j=1}^{C} e^{w_j^T \tilde{f}_i + b_j}}\right)\right]$$

$$= \sum_{c \in TC} \frac{1}{N_c} \sum_{i=1}^{N_c} E_{\tilde{f}_i}\left[\log(\sum_{j=1}^{C} e^{(w_j^T - w_{y_i}^T)\tilde{f}_i + (b_j - b_{y_i})})\right]. \tag{8}$$

Nevertheless, the above equation is difficult to calculate accurately. With the help of Jensen's inequality $E[\log X] \leq \log E[X]$, we can derive its upper bound as follows:

$$\mathcal{L}_{\mathcal{M} \to \infty} \leq$$

$$\sum_{c \in TC} \frac{1}{N_c} \sum_{i=1}^{N_c} \log\left(E_{\tilde{f}_i}\left[\sum_{j=1}^{C} e^{(w_j^T - w_{y_i}^T)\tilde{f}_i + (b_j - b_{y_i})}\right]\right)$$

$$= \sum_{c \in TC} \frac{1}{N_c} \sum_{i=1}^{N_c} \log\left(\sum_{j=1}^{C} E_{\tilde{f}_i}\left[e^{(w_j^T - w_{y_i}^T)\tilde{f}_i + (b_j - b_{y_i})}\right]\right). \tag{9}$$

Because of $\tilde{f}_i \sim N\left(f_i + \alpha\mu_{y_i}^r, \beta\left(\Sigma_{y_i} + \Sigma_{y_i}^r\right)\right)$, we can obtain that $(w_j^T - w_{y_i}^T)\tilde{f}_i + (b_j - b_{y_i})$ is also a Gaussian random variable, i.e. $(w_j^T - w_{y_i}^T)\tilde{f}_i + (b_j - b_{y_i}) \sim N\left((w_j^T - w_{y_i}^T)(f_i + \alpha\mu_{y_i}^r) + (b_j - b_{y_i}), \sigma_i^j\right)$, where $\sigma_i^j = \beta(w_j^T - w_{y_i}^T)\left(\Sigma_{y_i} + \Sigma_{y_i}^r\right)(w_j - w_{y_i})$.

After that, we use the moment-generating function:

$$\mathbb{E}[e^{tX}] = e^{t\mu + \frac{1}{2}\sigma t^2}, X \sim N(\mu, \sigma) \tag{10}$$

Finally, we can get Equation (11):

$$\mathcal{L}_{\mathcal{M} \to \infty} \leq \overline{\mathcal{L}}_{\infty} = -\sum_{c \in TC} \frac{1}{N_c} \sum_{i=1}^{N_c} \log \frac{e^{Z_i^{y_i}}}{\sum_{j=1}^{C} e^{Z_i^j}}, \tag{11}$$

Imbalance factor (λ)	200	100	50	20	10
Cross-Entropy[†]	65.30	61.54	55.98	48.94	44.27
Mixup[†] (Zhang et al. 2018)	-	60.46	55.01	-	41.98
L2RW[†] (Ren et al. 2018)	67.00	61.10	56.83	49.25	47.88
Class-balanced CE[†] (Cui et al. 2019)	64.44	61.23	55.21	48.06	42.43
Class-balanced fine-tuning[†] (Cui et al. 2018)	61.34	58.50	53.78	47.70	42.43
Meta-weight net[†] (Shu et al. 2019)	63.38	58.39	54.34	46.96	41.09
Meta-class-weight with cross-entropy loss[†] (Jamal et al. 2020)	60.69	56.65	51.47	44.38	40.42
BBN[*] (Zhou et al. 2020)	-	57.44	52.98	-	40.88
Hybrid-PSC[*] (Wang et al. 2021)	-	55.03	51.07	-	37.63
Bag of Tricks[*] (Zhang et al. 2021)	-	52.17	48.31	-	-
MetaSAug with cross-entropy loss[*] (Li et al. 2021)	60.06	53.13	48.10	42.15	38.27
RISDA	**55.24**	**49.84**	**46.16**	**41.33**	**37.62**

Table 1: Error rate (%) of ResNet-32 on CIFAR-100-LT under different imbalance factors.

where $Z_i^j = \hat{y}_i^j + \alpha(\boldsymbol{w}_j^T - \boldsymbol{w}_{y_i}^T)\boldsymbol{\mu}_j^r + \frac{\sigma_i^j}{2}$, and $\hat{y}_i^j$ is the j-th logits of the output for x_i. Essentially, Equation (11) provides a surrogate loss for our reasoning implicit data augmentation method, through optimizing upper bound $\overline{\mathcal{L}}$ instead of minimizing the exact loss function $\mathcal{L}_\infty$. This new loss can effectively adjust the classification decision boundary with the reasoning-based augmented features.

But when we generate unlimited samples for tail categories, the data distribution is still unbalanced where the head categories become "tail categories". Therefore, similar to the tail class, we also make infinite enhancements for head classes. In this way, we find that head categories are still dominant because the augmentation results rely on the training samples and the head class has most of the samples. To solve this problem, we introduce the re-weighting strategy by setting different weights to different classes, which are defined as $\rho_c \approx (1 - \gamma)/(1 - \gamma^{N_c})$, where γ is a hyperparameter. With the re-weighting strategy, we can modify the loss function as follows:

$$\mathcal{L}_{\mathcal{M}\to\infty} \le \overline{\mathcal{L}}_\infty = -\sum_{c=1}^{C} \frac{1}{N_c} \sum_{i=1}^{N_c} \rho_c log \frac{e^{Z_i^{y_i}}}{\sum_{j=1}^{C} e^{Z_i^j}}. \quad (12)$$

Our reasoning module can generate more diversified samples by reasoning features and transformations from other classes, which is shown in the experiment. In addition, the proposed novel robust loss can be easily adopted as a plug-in module for other methods.

Experiment

Datasets

We conduct experiments on three long-tailed datasets: CIFAR-100-LT, ImageNet-LT, and iNaturalist 2018[2].

CIFAR-100 is a balanced dataset containing 60,000 images from 100 categories. In our experiment, an imbalance factor λ, which is the ratio of sample numbers of the most frequent and least frequent classes, is used to generate different training sets for CIFAR-100-LT (Cui et al. 2019; Cao

et al. 2019). By varying $\lambda \in \{200, 100, 50, 20, 10\}$, we can obtain five training sets.

ImageNet-LT is constructed from ImageNet by discarding some training samples (Liu et al. 2019). It has 115,846 training images, 20,000 validation images, and 50,000 testing images. The most frequent class contains 1,280 images, while the least frequent one only has 5 samples.

The iNaturalist 2018 is a large-scale species dataset with an extremely imbalanced distribution, where the imbalance factor is 500. It contains 437,513 training images and 24,426 validation images from 8,142 classes.

Results on CIFAR-100-LT

For CIFAR-100-LT, we use ResNet-32 as our backbone, which is trained by standard stochastic gradient descent (SGD) with a momentum of 0.9 and a weight decay 5×10^{-4}. We train the model for 200 epochs with a batch size of 100. The initial learning rate is set to 0.10, and the linear warm-up learning rate schedule is adopted. Besides, we decay the learning rate by 0.01 at the 160th and 180th epochs. For the hyperparameters α and β, we select them from $\{0.25, 0.50, 0.75, 1.00, 1.25, 1.50\}$. Different λ have different optimal α and β, which are shown in Table 2. We compare our RISDA with some state-of-the-art methods. The results are shown in Table 1, where "*" indicates the results reported in the original paper, and "†" indicates the results reported in (Li et al. 2021).

λ	200	100	50	20	10
α	0.50	0.50	0.75	0.75	0.50
β	1.00	0.75	1.00	0.75	0.50

Table 2: Optimal α and β under different λ

We can draw the following conclusions: Firstly, our method can achieve the best performance compared with existing state-of-the-art methods. Secondly, the more unbalanced the data set is, the more our method improves. For example, when the λ is 200 and 100, the error rate is reduced by 4.82% and 3.29 % compared with MetaSAug.

Method	Error rate (%)
Cross-Entropy [†]	61.12
Class-balanced CE[†] (Cui et al. 2019)	59.15
OLTR[†] (Liu et al. 2019)	59.64
LDAM[†] (Cao et al. 2019)	58.14
LDAM-DRW[†] (Cao et al. 2019)	54.26
Meta-class-weight[†] (Jamal et al. 2020)	55.08
MetaSAug[*] (Li et al. 2021)	52.61
Bag of Tricks[*] (Zhang et al. 2021)	56.87
RISDA	**49.31**

Table 3: Results on ImageNet-LT of different methods.

While, when the λ is 20, the error rate is reduced by 0.82%. However, when the λ is 10, the error rate is only reduced by 0.65%. We believe that this situation occurs because the fewer samples, the worse the covariance matrix learned by the tail category. Then, the variations transferred from similar categories can better improve the feature diversity of a tail category. Therefore, we can conclude that our method is more suitable for dealing with extremely imbalanced data.

Results on ImageNet-LT

For fairness, we use ResNet-50 as the backbone. Concretely, the ResNet-50 is trained for 100 epochs by SGD with a momentum of 0.9 and a weight decay 2×10^{-4}. We set the initial learning rate to 0.1 and use the linear warm-up learning rate schedule for the first five epochs. Then, we decay the learning rate by 0.1 at the 60th and 80th epochs. Due to GPU memory limitation, we only transfer the covariance, so we augment a sample $\boldsymbol{f}_i$ during training with $N(\boldsymbol{f}_i, \beta(\boldsymbol{\Sigma}_c + \boldsymbol{\Sigma}_c^r))$. We set $\beta = 7.5$. To recognize tail classes for augmentation, we define classes with more than 100 images as head classes, otherwise are the tail classes.

We compare our RISDA with the following methods: 1)Cross-Entropy(CE), class-balanced loss, meta-class-weight, LDAM-DRW, OLTR, and Bag of Tricks. 2)Mixup and MetaSAug. The experimental results are shown in Table 3. We can see that the error rate of RISDA is reduced to 49.31%. More importantly, the error rate is reduced by 3.30% compared with MetaSAug, which shows that the diversity depicted by tail classes is limited, and can be greatly enriched by borrowing transformations from other classes.

Results on iNaturalist 2018

For iNaturalist 2018, we also implement ResNet-50 to achieve better classification results. We train the network with a batch size 128 for 120 epochs by SGD with a momentum of 0.9 and a weight decay 1×10^{-4}. The learning rate is initialized to 0.05. We adopt the linear warm-up learning rate schedule (Goyal et al. 2017) and decay the learning rate by 0.1 at 60th and 80th epochs. Referring to the setting in paper (Liu et al. 2019), we split the classes into head classes (with more than 100 images) and tail classes (with less than or equal to 100 images). Due to GPU memory limitation, we also only transfer the covariance, so a sample $\boldsymbol{f}_i$ is augmented during training with $N(\boldsymbol{f}_i, \beta(\boldsymbol{\Sigma}_c + \boldsymbol{\Sigma}_c^r))$.

Method	Error rate (%)
Cross-Entropy[†]	34.24
Class-balanced CE[†] (Cui et al. 2019)	33.57
Class-balanced focal[*] (Cui et al. 2019)	38.88
cRT[*] (Kang et al. 2020)	32.40
LDAM[*] (Cao et al. 2019)	35.42
LDAM-DRW[*] (Cao et al. 2019)	32.00
BBN[*] (Zhou et al. 2020)	33.71
Meta-class-weight[*] (Jamal et al. 2020)	32.45
MetaSAug[*] (Li et al. 2021)	31.25
Hybrid-PSC[*] (Wang et al. 2021)	31.90
Bag of Tricks[*] (Zhang et al. 2021)	**29.13**
RISDA	30.85

Table 4: Results on iNaturalist 2018 of different methods.

We compare our RISDA with the following methods: 1) CE, class-balanced with CE and focal loss, meta-class-weight, and LDAM-DRW. 2) BBN, decoupling, and Hybrid-PSC. 3)Bag of Tricks and MetaSAug. The experimental results are shown in Table 4. The average error rate of our method is reduced to 30.85%.

Analysis

In this section, we conduct experiments to study the effect of different components of the proposed RISDA and discuss the parameter sensitivity on CIFAR-100-LT.

Ablation Study To verify the effect of each part of our method, we do experiments by removing the re-weighting (w/o w) or the reasoning (w/o r) component from our RISDA. Results shown in Table 5 demonstrate that: (1) Re-weighting is important in our RISDA. Although we aim to generate unlimited samples through the Gaussian distribution, augmentation is implemented in each sample. Therefore, re-weighting plays a significant role to achieve fairness in training. (2) The reasoning module can effectively refine the feature and the covariance matrix to improve the accuracy of classification. (3) In addition, when λ is 200, the reasoning module decreased by 1.73%, and when the λ is 10, the reasoning module decreased by 0.81%, which proves again that our reasoning module is more suitable for dealing with extremely unbalanced data.

λ	200	100	50	20	10
w/o w	36.82	42.76	46.98	54.40	58.60
w/o r	56.97	51.09	47.60	42.60	38.43
RISDA	**55.24**	**49.84**	**46.16**	**41.33**	**37.62**

Table 5: Results of RISDA on CIFAR-100-LT under different imbalance factors.

Sensitivity of α and β For the Gaussian distribution $N(\alpha\boldsymbol{\mu}_c, \beta(\boldsymbol{\Sigma}_c + \boldsymbol{\Sigma}_c^r))$ constructed by the reasoning module, the multivariate of sampling includes two significant componets: reasoning-based complementary feature $\alpha\boldsymbol{\mu}_c$
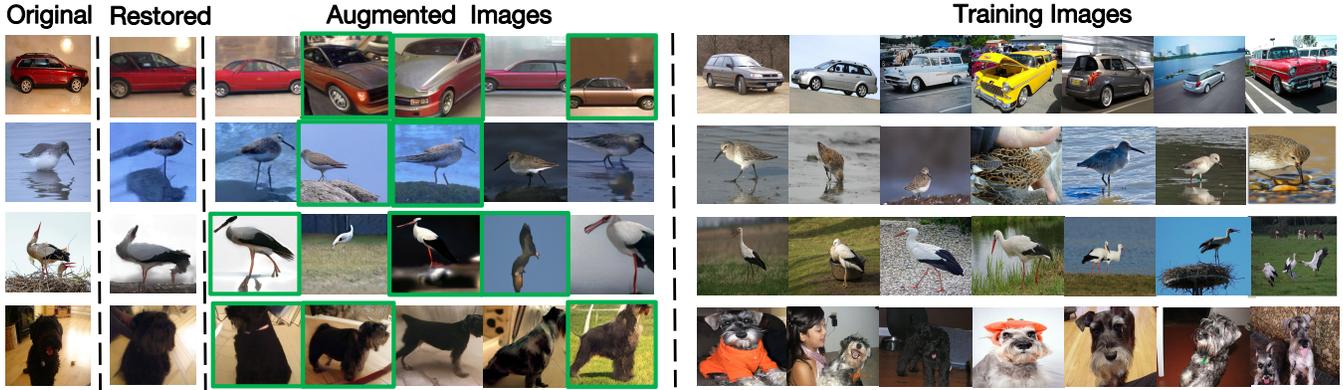
Figure 3: Visualization of the semantically augmented images for the four tail classes on ImageNet-LT: Wagon, Dunlin, White Stork, and Miniature Schnauzer. "Original" represents the original training sample. "Restored" is the image reconstructed by a generator with features of the corresponding training sample. And "Augmented Images" are generated with features sampled from our refined feature distribution. Our method can generate samples with semantic transformations that are not found in the original training set in terms of color, angle, pose, background, etc. These samples are highlighted in green boxes.
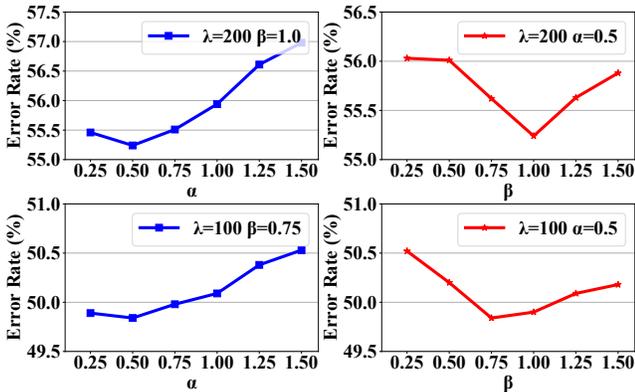

Figure 4: The influence of different α and β on CIFAR-100-LT with $\lambda = 100$ and $\lambda = 200$.

Figure 5: Results with different number of head categories.

and the reasoning-based covariance matrix $\beta\left(\Sigma_c + \Sigma_c^r\right)$. We analyze the influence of the transformation strength α, $\beta \in \{0.25, 0.50, 0.75, 1.00, 1.25, 1.50\}$.

As shown in Figure 4, we can see that: (1) $\alpha = 0.50$ can achieve the best performance with $\lambda = 100$, and 200. Then, when α increases, the performance drops. It indicates that features of other classes can help to complement the feature of a tail instance. (2) $\beta = 1.00$ achieves best results on $\lambda = 200$, and $\beta = 0.75$ performs best on $\lambda = 100$. We can conclude that the more unbalanced, the larger β is better.

Effect of Head Categories　Head categories are expected to have rich diversity. Transferring transformations from other classes may damage their representation, reducing the overall performance. Therefore, we explore the influence of head categories by varying the number in $\{10, 20, 30, 40\}$.

As shown in Figure 5, the green bar on the left and the orange bar on the right shows the results with the different number of head categories on $\lambda = 200$ and $\lambda = 100$, respectively. We can see that when the top 20 categories are re-
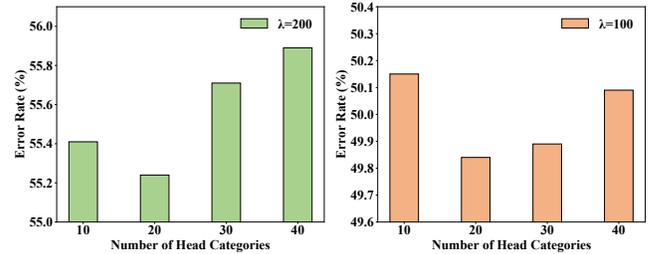
garded as head categories, the best performance is achieved. When it increases, categories with limited diversity are underrepresented. However, when it decreases, the representation of some rich categories is damaged.

Qualitative Analysis

To intuitively show that our RISDA can generate more various samples, we implement the visualization method proposed in ISDA to show some implicitly generated samples. As shown in Figure 3, we can see that our RISDA can generate samples with different backgrounds, view angles, and poses. More importantly, images in green boxes are quite different from those in training sets. For example, the "Wagon" changes the background, view angle, and color while the "Dunlin" changed the posture, background, view angle. These changes are not available in the training data, and we believe they are delivered from similar categories. Therefore, it can be concluded that our RISDA can transfer more meaningful transformations than ISDA.

Conclusion and Future Work

In this paper, we have proposed a novel reasoning-based implicit semantic data augmentation method to effectively enrich the intra-class diversity for tail categories. By transferring transformation directions from other classes, an in-

Imbalance factor (λ)	200	100	50	20	10
Cross-Entropy	34.13	29.86	25.06	17.56	13.82
Mixup[†] (Zhang et al. 2018)	-	26.94	22.18	-	12.90
L2RW[†] (Ren et al. 2018)	33.75	27.77	23.55	18.65	17.88
Class-balanced CE[†] (Cui et al. 2019)	31.23	27.32	21.87	15.44	13.10
Class-balanced fine-tuning[†] (Cui et al. 2018)	33.76	28.66	22.56	16.78	16.83
Meta-weight net[†] (Shu et al. 2019)	32.80	26.43	20.90	15.55	12.45
Meta-class-weight with cross-entropy loss[†] (Jamal et al. 2020)	29.34	23.59	19.49	13.54	11.15
BBN[*] (Zhou et al. 2020)	-	20.18	17.82	-	11.68
Hybrid-PSC[*] (Wang et al. 2021)	-	21.18	**14.64**	-	**9.94**
Bag of Tricks[*] (Zhang et al. 2021)	-	19.97	16.41	-	-
MetaSAug with cross-entropy loss[*] (Li et al. 2021)	**23.11**	**19.46**	15.97	**12.36**	10.56
RISDA	<u>26.00</u>	<u>20.11</u>	<u>15.76</u>	<u>13.02</u>	<u>10.64</u>

Table 6: Error rate (%) of ResNet-32 on CIFAR-10-LT under different imbalance factors.

stance from tail categories can be augmented with definitely different semantics. Experimental results have shown that our proposed method can improve the performance of long-tail classification. Besides, the visualization experiment has demonstrated that our augmentation method can generate samples that are quite different from those in the training data. In the future, we will work on some fine-grained transformations to achieve more reasonable augmentation.

Acknowledgments

Appendix

Details about Baselines on CIFAR

In the CIFAR-100-LT and CIFAR-10-LT experiment, we compare with the following methods: (1) Cross-Entropy (CE): we train the model with vanilla cross-entropy loss. (2)Re-weighting (RW): we train the model with two re-weighting strategies: class-level and instance-level re-weighting. Class-level re-weighting assign different weights to different classes according to the number of samples in loss function, including class-balanced loss (Cui et al. 2019) , meta-class-weight (Jamal et al. 2020), and LDAM-DRW (Cao et al. 2019). Instance-level re-weighting allocate weights to samples, including Focal loss (Lin et al. 2017), L2RW (Ren et al. 2018) and meta-weight-net (Shu et al. 2019). (3)Data augmentation method improves classification performance by enhancing data diversity, including mixup (Zhang et al. 2018) and MetaSAug (Li et al. 2021). (4)Two-stage learning splits the learning procedure into feature representation learning and classifier training, such as BBN (Zhou et al. 2020), decoupling (Kang et al. 2020) and Hybrid-PSC (Wang et al. 2021). In addition, we also compare with Bag of Tricks(Zhang et al. 2021), which integrates the commonly used training tricks in long-tail classification.

Results on CIFAR-10-LT

CIFAR-10 is a balanced dataset containing 60,000 images from 10 categories, and the training set has 5000 images per class. By varying $\lambda \in \{200, 100, 50, 20, 10\}$, we can obtain five training sets. A long-tail distribution usually contains many classes. As CIFAR-10-LT only contains 10 classes, we leave out its experiments in the main body.

Experimental results on CIFAR-10-LT are shown in Table 6. We can see that our RISDA is better than most baseline methods, but achieves comparable performance with MetaSAug. We believe that this situation occurs because our method is good at dealing with large-scale long-tailed classification problem. With more categories, there are rich similarity relations to make reasonable transformatnion. In addition, the CIFAR-10-LT dataset contains a large number of samples in each category. Even in the most unbalanced case, there are 25 samples for the least tail categories. Consequently, the diversity of samples learned in each category is rich enough. Therefore, although RISDA performs better, we argue that it is more suitable to deal with many classes.

λ	200	100	50	20	10
CE[†]	65.30	61.54	55.98	48.94	44.27
ISDA	64.48	58.65	54.35	47.25	42.84
ISDA+w	56.97	51.09	47.60	42.60	38.43
RISDA	**55.24**	**49.84**	**46.16**	**41.33**	**37.62**

Table 7: Error rate (%) on CIFAR-100-LT.

Comparison with ISDA

Our RISDA is developed from ISDA to achieve implicitly data augmentation. However, ISDA fails on long-tail distributed data. Firstly, the diversity of augmented data for tail classes is quite limited because ISDA enhances a sample with the covariance of its class. As tail classes only own a few samples, their covariance is limited. Secondly, ISDA lacks strategies to achieve balance training. For the sake of fairness, we show the results of ISDA and ISDA with

reweighting in Table 7. We can see that even if equipped with reweighting, ISDA is still worse than RISDA because of the limited diversity.

Our RISDA diversify tail classes from two aspects: (1) We transfer covariance from similar classes to tail classes to expand the variations. Therefore, RSIDA can imagine a flying bird when there are only birds on the roost in the training set(the flying is transferred from similar classes). (2) A tail sample may miss some features since tail classes are under-fitted. If a bird's wing features can't be extracted, the corresponding semantic changes will not work. In RSIDA, we can infer some missing features through the knowledge graph, so as to achieve more variations for a specific sample.

References

Bowles, C.; Chen, L.; Guerrero, R.; Bentley, P.; Gunn, R. N.; Hammers, A.; Dickie, D. A.; del C. Valdés Hernández, M.; Wardlaw, J. M.; and Rueckert, D. 2018. GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks. *CoRR*, abs/1810.10863.

Buda, M.; Maki, A.; and Mazurowski, M. A. 2018. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106: 249–259.

Byrd, J.; and Lipton, Z. C. 2019. What is the Effect of Importance Weighting in Deep Learning? In *ICML*, 872–881.

Cao, K.; Wei, C.; Gaidon, A.; Aréchiga, N.; and Ma, T. 2019. Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss. In *NeurIPS*, 1565–1576.

Chu, P.; Bian, X.; Liu, S.; and Ling, H. 2020. Feature space augmentation for long-tailed data. In *ECCV*, 694–710.

Cui, Y.; Jia, M.; Lin, T.; Song, Y.; and Belongie, S. J. 2019. Class-Balanced Loss Based on Effective Number of Samples. In *CVPR*, 9268–9277.

Cui, Y.; Song, Y.; Sun, C.; Howard, A.; and Belongie, S. J. 2018. Large Scale Fine-Grained Categorization and Domain-Specific Transfer Learning. In *CVPR*, 4109–4118.

Goyal, P.; Dollár, P.; Girshick, R. B.; Noordhuis, P.; Wesolowski, L.; Kyrola, A.; Tulloch, A.; Jia, Y.; and He, K. 2017. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. *CoRR*, abs/1706.02677.

He, H.; Bai, Y.; Garcia, E. A.; and Li, S. 2008. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *WCCI*, 1322–1328.

He, H.; and Garcia, E. A. 2009. Learning from Imbalanced Data. *IEEE Trans. Knowl. Data Eng.*, 21(9): 1263–1284.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *CVPR*, 770–778.

Huang, C.; Li, Y.; Loy, C. C.; and Tang, X. 2016. Learning deep representation for imbalanced classification. In *CVPR*, 5375–5384.

Huang, G.; Liu, Z.; van der Maaten, L.; and Weinberger, K. Q. 2017. Densely Connected Convolutional Networks. In *CVPR*, 2261–2269.

Jamal, M. A.; Brown, M.; Yang, M.; Wang, L.; and Gong, B. 2020. Rethinking Class-Balanced Methods for Long-Tailed Visual Recognition From a Domain Adaptation Perspective. In *CVPR*, 7607–7616.

Kang, B.; Xie, S.; Rohrbach, M.; Yan, Z.; Gordo, A.; Feng, J.; and Kalantidis, Y. 2020. Decoupling Representation and Classifier for Long-Tailed Recognition. In *ICLR*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In Bartlett, P. L.; Pereira, F. C. N.; Burges, C. J. C.; Bottou, L.; and Weinberger, K. Q., eds., *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, 1106–1114.

Li, S.; Gong, K.; Liu, C. H.; Wang, Y.; Qiao, F.; and Cheng, X. 2021. MetaSAug: Meta Semantic Augmentation for Long-Tailed Visual Recognition. In *CVPR*, 5212–5221.

Lin, T.; Goyal, P.; Girshick, R. B.; He, K.; and Dollár, P. 2017. Focal Loss for Dense Object Detection. In *ICCV*, 2999–3007.

Liu, J.; Sun, Y.; Han, C.; Dou, Z.; and Li, W. 2020. Deep representation learning on long-tailed data: A learnable embedding augmentation perspective. In *CVPR*, 2970–2979.

Liu, Z.; Miao, Z.; Zhan, X.; Wang, J.; Gong, B.; and Yu, S. X. 2019. Large-scale long-tailed recognition in an open world. In *CVPR*, 2537–2546.

Ratner, A. J.; Ehrenberg, H. R.; Hussain, Z.; Dunnmon, J.; and Ré, C. 2017. Learning to Compose Domain-Specific Transformations for Data Augmentation. In *NeurIPS*, 3236–3246.

Ren, M.; Zeng, W.; Yang, B.; and Urtasun, R. 2018. Learning to reweight examples for robust deep learning. In *ICML*, 4334–4343.

Shu, J.; Xie, Q.; Yi, L.; Zhao, Q.; Zhou, S.; Xu, Z.; and Meng, D. 2019. Meta-Weight-Net: Learning an Explicit Mapping For Sample Weighting. In *NeurIPS*, 1917–1928.

Simonyan, K.; and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*.

Wang, P.; Han, K.; Wei, X.-S.; Zhang, L.; and Wang, L. 2021. Contrastive Learning based Hybrid Networks for Long-Tailed Image Classification. In *CVPR*, 943–952.

Wang, Y.; Pan, X.; Song, S.; Zhang, H.; Huang, G.; and Wu, C. 2019. Implicit Semantic Data Augmentation for Deep Networks. In *NeurIPS*, 12614–12623.

Xiao, L.; Zhang, X.; Jing, L.; Huang, C.; and Song, M. 2021. Does Head Label Help for Long-Tailed Multi-Label Text Classification. In *AAAI*, 14103–14111.

Zhang, H.; Cissé, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2018. mixup: Beyond Empirical Risk Minimization. In *ICLR*.

Zhang, Y.; Wei, X.; Zhou, B.; and Wu, J. 2021. Bag of Tricks for Long-Tailed Visual Recognition with Deep Convolutional Neural Networks. In *AAAI*, 3447–3455.

Zhou, B.; Cui, Q.; Wei, X.-S.; and Chen, Z.-M. 2020. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *CVPR*, 9719–9728.