Data Augmentation-Based Long-tailed Learning

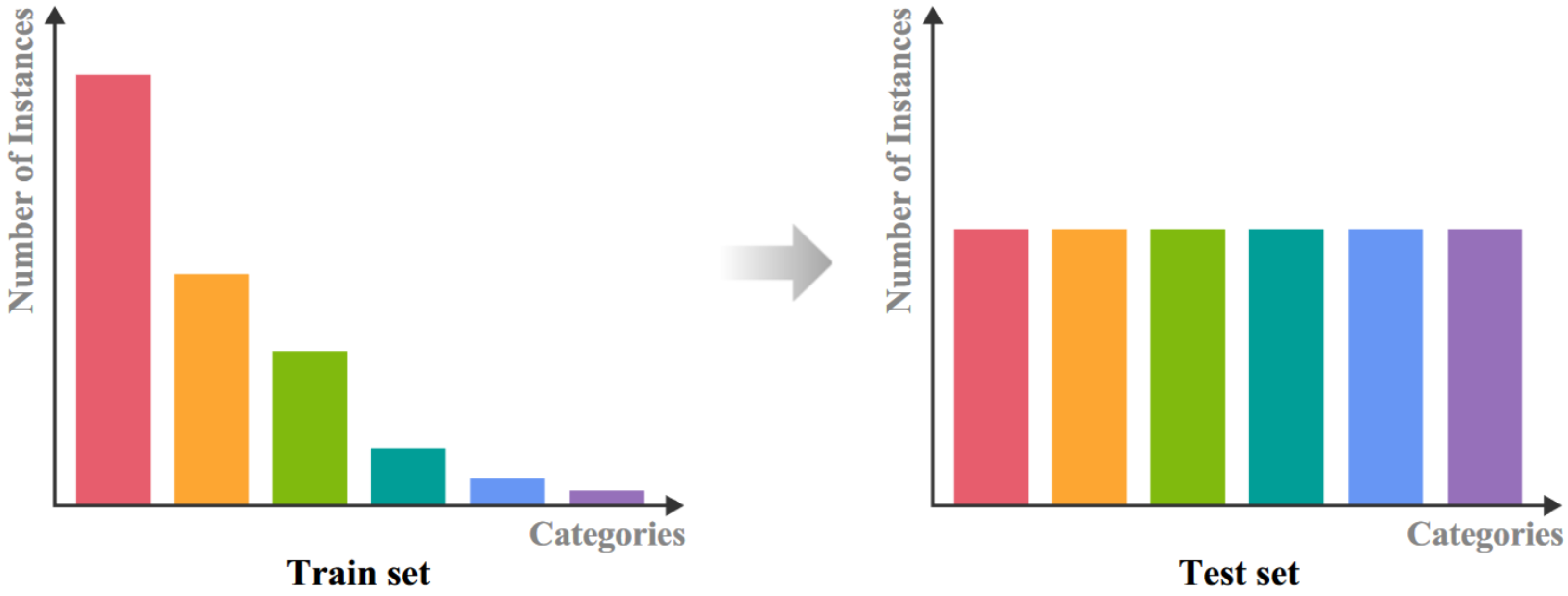
CUDA: Curriculum of Data Augmentation for Long-tailed Recognition

Sumyeong Ahn*, Jongwoo Ko*, Se-Young Yun
KAIST AI
Seoul, Korea
{sumyeongahn, jongwoo.ko, yunseyoung}@kaist.ac.kr

ICLR 2023

Background

Long-tailed Recognition



Background

Taxonomy of existing deep long-tailed learning methods[1]

- **Long-tailed Learning**
  - **Class Re-balancing** (Sec. 3.1)
    - **Re-sampling** (Sec. 3.1.1)
    - **Class-sensitive Learning** (Sec. 3.1.2)
    - **Logit Adjustment** (Sec. 3.1.3)
  - **Information Augmentation** (Sec. 3.2)
    - **Transfer Learning** (Sec. 3.2.1)
    - **Data Augmentation** (Sec. 3.2.2)
  - **Module Improvement** (Sec. 3.3)
    - **Representation Learning** (Sec. 3.3.1)
    - **Classifier Design** (Sec. 3.3.2)
    - **Decoupled Training** (Sec. 3.3.3)
    - **Ensemble Learning** (Sec. 3.3.4)

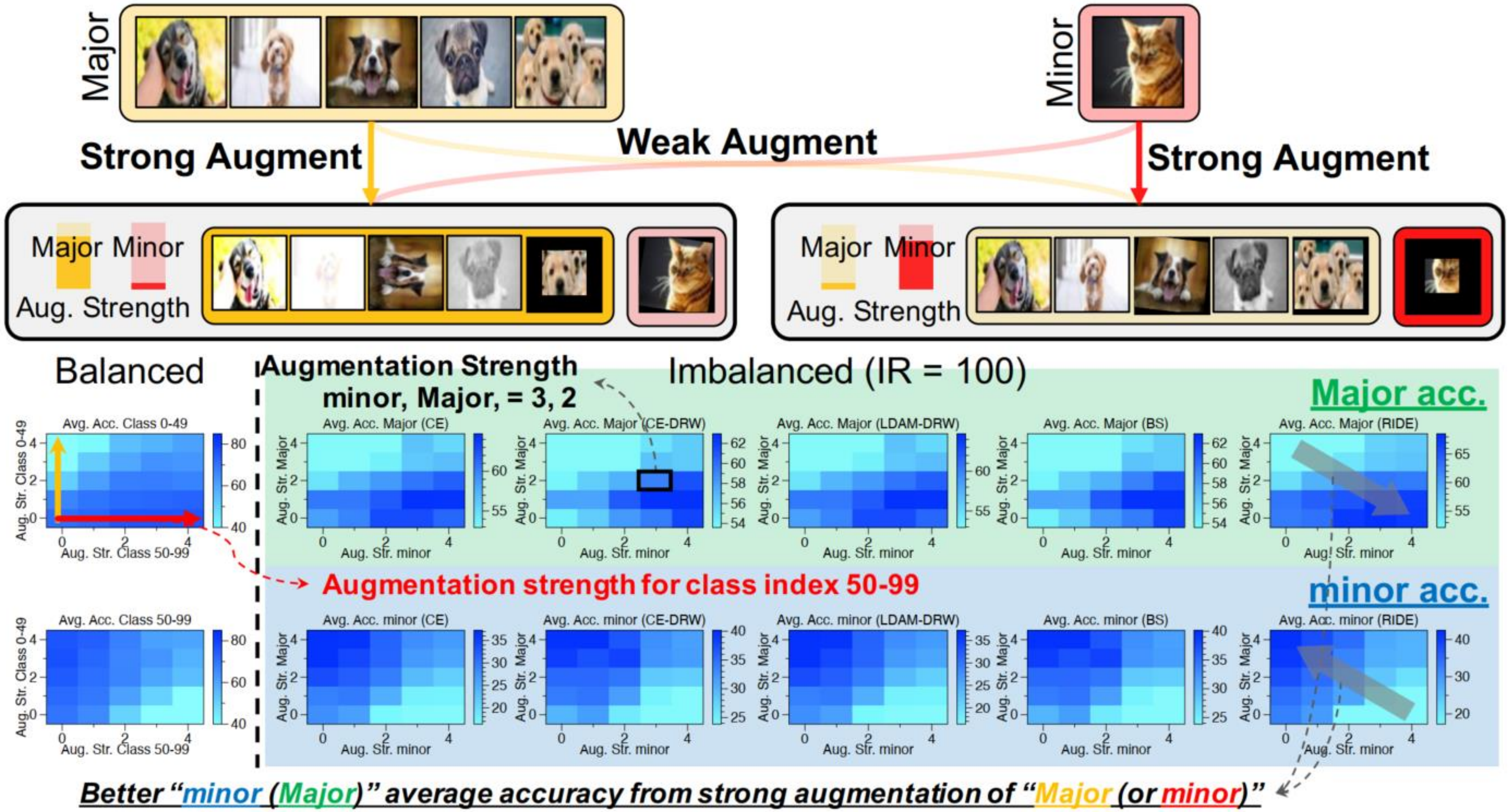
[1] Zhang Y, Kang B, Hooi B, et al. Deep long-tailed learning: A survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence 2023.

Motivation

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

Few works have considered the influence of the degree of augmentation of different classes on class imbalance problems.

Controlling the strength of class-wise augmentation:



Key Observation:

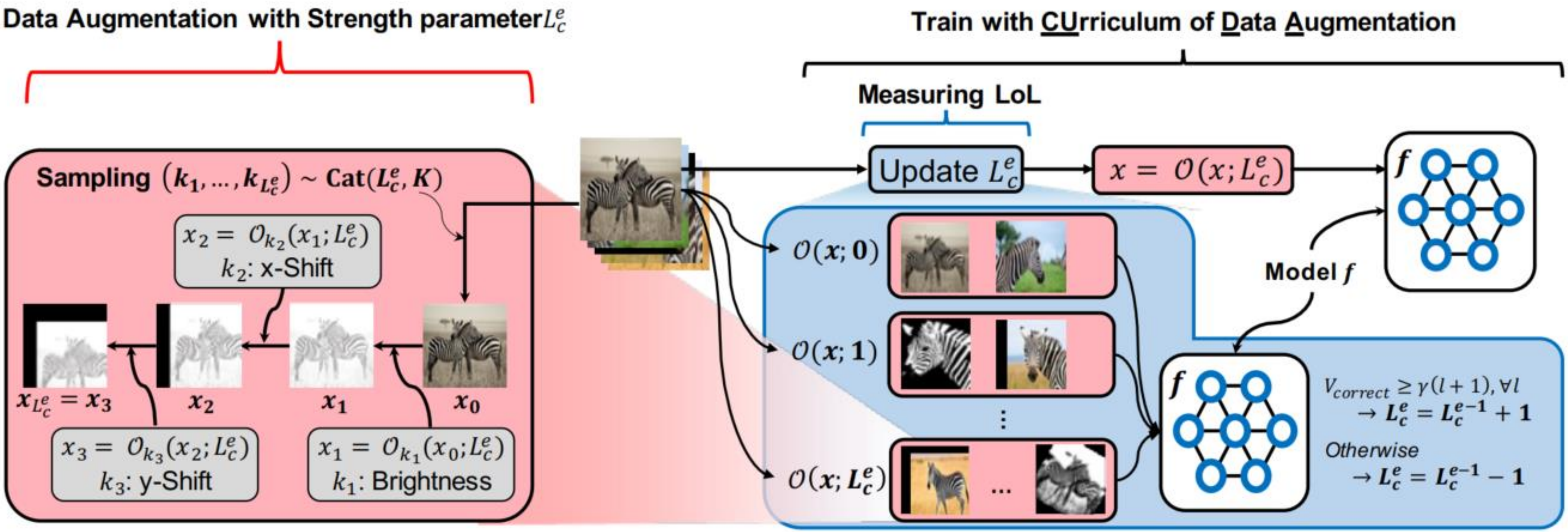
class-wise augmentation improves performance in the non-augmented classes while that for the augmented classes may not be significantly improved !

Methods

CUDA includes two parts:

(1) a method to generate the augmented samples based on the given strength parameter;

(2) a method to measure a Level-of-Learning (LoL) score for each class.



Alogrithm

Algorithm 1: CUrriculum of Data Augmentation

Input: LTR algorithm $\mathcal{A}_{\text{LTR}}(f, \mathcal{D})$, training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, train epochs E, aug. probability p_{aug}, threshold γ, number of sample coefficient T.
Output: trained model f_θ
Initialize: $L_c^0 = 0 \;\forall c \in \{1, ..., C\}$
for $e \le E$ **do**
    Update $L_c^e = V_{\text{LoL}}(\mathcal{D}_c, L_c^{e-1}, f_\theta, \gamma, T) \quad \forall c$ // Alg. $\boxed{2}$
    Generate $\mathcal{D}_{\text{CUDA}} = \{(\bar{x}_i, y_i) | (x_i, y_i) \in \mathcal{D}\}$ where

$$\bar{x}_i = \begin{cases} \mathcal{O}(x_i, L_{y_i}^e) & \text{with prob. } p_{\text{aug}} \\ x_i & \text{otherwise.} \end{cases}$$

    Run LTR algorithm using $\mathcal{D}_{\text{CUDA}}$, *i.e.*, $\mathcal{A}_{\text{LTR}}(f_\theta, \mathcal{D}_{\text{CUDA}})$.
end

Algorithm 2: V_{LoL}: Update LoL score

Input: $\mathcal{D}_c$, L, f_θ, γ, T
Output: updated L
Initialize: check $= 1$
for $l \le L$ **do**
    /* $V_{\text{correct}}(\mathcal{D}_c, l, f_\theta, T)$ */
    Sample $\mathcal{D}'_c \subset \mathcal{D}_c$ s.t. $|\mathcal{D}'_c| = T(l+1)$
    Compute $v = \sum_{x \in \mathcal{D}'_c} \mathbb{1}_{\{f(\mathcal{O}(x;l)=c\}}$
    if $v \le \gamma T(l+1)$ **then**
        check $\leftarrow 0$; **break**
    end
end
if check $= 1$ **then** $L \leftarrow L + 1$
else $L \leftarrow L - 1$

Experiment

Table 1: Validation accuracy on CIFAR-100-LT dataset. † are from Park et al. (2022) and ‡, ⋆ are from the original papers (Kim et al., 2020; Zhu et al., 2022). Other results are from our implementation. We format the first and second best results as **bold** and <u>underline</u>. We report the average results of three random trials.

Algorithm	Imbalance Ratio (IR)			Statistics (IR 100)		
	100	50	10	Many	Med	Few
CE	$38.7_{\pm0.4}$	$43.4_{\pm0.3}$	$56.5_{\pm0.6}$	$66.2_{\pm0.5}$	$37.3_{\pm0.6}$	$8.2_{\pm0.3}$
CE + CMO (Park et al., 2022)	$42.0_{\pm0.4}$	$47.0_{\pm0.5}$	$60.0_{\pm0.4}$	$69.1_{\pm0.4}$	$41.2_{\pm0.6}$	$11.3_{\pm0.7}$
CE + CUDA	$42.7_{\pm0.4}$	$47.2_{\pm0.4}$	$59.6_{\pm0.4}$	$\mathbf{71.6_{\pm0.6}}$	$42.3_{\pm0.3}$	$9.4_{\pm0.7}$
CE + CMO + CUDA	$43.5_{\pm0.5}$	$48.7_{\pm0.6}$	$60.0_{\pm0.3}$	$70.0_{\pm0.7}$	$43.4_{\pm0.5}$	$12.7_{\pm0.8}$
CE-DRW (Cao et al., 2019)	$41.4_{\pm0.2}$	$45.5_{\pm0.6}$	$57.8_{\pm0.6}$	$62.8_{\pm0.5}$	$41.7_{\pm0.7}$	$16.1_{\pm0.4}$
CE-DRW + Remix (Chou et al., 2020)†	45.8	49.5	59.2	-	-	-
CE-DRW + CUDA	$47.7_{\pm0.4}$	$52.4_{\pm0.5}$	$61.6_{\pm0.5}$	$64.3_{\pm0.4}$	$49.2_{\pm0.6}$	$26.7_{\pm0.6}$
LDAM-DRW (Cao et al., 2019)	$42.5_{\pm0.2}$	$47.4_{\pm0.5}$	$57.6_{\pm0.1}$	$62.8_{\pm0.5}$	$42.3_{\pm0.6}$	$19.0_{\pm0.7}$
LDAM + M2m (Kim et al., 2020)‡	43.5	-	57.6	-	-	-
LDAM-DRW + CUDA	$47.6_{\pm0.7}$	$51.1_{\pm0.4}$	$58.4_{\pm0.1}$	$67.3_{\pm0.6}$	$50.4_{\pm0.5}$	$21.4_{\pm0.2}$
BS (Ren et al., 2020)	$43.3_{\pm0.4}$	$46.9_{\pm0.2}$	$58.3_{\pm0.4}$	$61.6_{\pm0.8}$	$42.3_{\pm0.5}$	$23.0_{\pm0.4}$
BS + CUDA	$47.7_{\pm0.3}$	$52.1_{\pm0.4}$	$61.7_{\pm0.5}$	$63.3_{\pm0.4}$	$48.4_{\pm0.4}$	$28.7_{\pm0.5}$
RIDE (3 experts) (Wang et al., 2021)†	48.6	51.4	59.8	-	-	-
RIDE (3 experts)	$49.7_{\pm0.2}$	$52.7_{\pm0.1}$	$60.2_{\pm0.2}$	$67.7_{\pm0.6}$	$51.5_{\pm0.5}$	$26.7_{\pm0.6}$
RIDE + CMO (Park et al., 2022)†	50.0	53.0	60.2	-	-	-
RIDE + CMO	$49.9_{\pm0.1}$	$53.0_{\pm0.1}$	$58.9_{\pm0.3}$	$67.3_{\pm0.3}$	$51.3_{\pm0.6}$	$28.1_{\pm0.4}$
RIDE (3 experts) + CUDA	$50.7_{\pm0.2}$	$53.7_{\pm0.4}$	$60.2_{\pm0.1}$	<u>$69.2_{\pm0.3}$</u>	$52.8_{\pm0.2}$	$27.3_{\pm0.8}$
BCL (Zhu et al., 2022)⋆	<u>51.0</u>	<u>54.9</u>	<u>64.4</u>	67.2	<u>53.1</u>	<u>32.9</u>
BCL + CUDA	$\mathbf{52.3_{\pm0.2}}$	$\mathbf{56.2_{\pm0.4}}$	$\mathbf{64.6_{\pm0.1}}$	$66.4_{\pm0.2}$	$\mathbf{54.2_{\pm0.6}}$	$\mathbf{33.9_{\pm0.8}}$

Experiment

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

Table 2: Validation accuracy on ImageNet-LT and iNaturalist 2018 datasets. † indicates reported results from the Park et al. (2022) and ‡ indicates those from the original paper (Kang et al., 2020). ⋆ means we train the network with the official code in an one-stage RIDE.

Algorithm	ImageNet-LT				iNaturalist 2018			
	Many	Med	Few	All	Many	Med	Few	All
CE†	64.0	33.8	5.8	41.6	73.9	63.5	55.5	61.0
CE + CUDA	**67.2**$_{\pm0.1}$	47.0$_{\pm0.2}$	13.5$_{\pm0.3}$	47.3$_{\pm0.2}$	**74.6**$_{\pm0.3}$	64.9$_{\pm0.1}$	57.2$_{\pm0.1}$	62.5$_{\pm0.2}$
CE-DRW (Cao et al., 2019)	61.7$_{\pm0.1}$	47.1$_{\pm0.3}$	29.0$_{\pm0.3}$	50.1$_{\pm0.1}$	68.2$_{\pm0.2}$	67.3$_{\pm0.2}$	66.4$_{\pm0.1}$	67.0$_{\pm0.1}$
CE-DRW + CUDA	61.8$_{\pm0.1}$	48.3$_{\pm0.1}$	30.3$_{\pm0.2}$	51.0$_{\pm0.1}$	68.8$_{\pm0.1}$	68.1$_{\pm0.3}$	66.6$_{\pm0.2}$	67.5$_{\pm0.1}$
LWS (Kang et al., 2020)‡	57.1	45.2	29.3	47.7	65.0	66.3	65.5	65.9
cRT (Kang et al., 2020)‡	58.8	44.0	26.1	47.3	69.0	66.0	63.2	65.2
cRT + CUDA	62.3$_{\pm0.1}$	47.2$_{\pm0.2}$	28.4$_{\pm0.5}$	50.2$_{\pm0.2}$	68.2$_{\pm0.1}$	67.8$_{\pm0.2}$	66.4$_{\pm0.1}$	67.3$_{\pm0.1}$
LDAM-DRW (Cao et al., 2019)†	60.4	46.9	30.7	49.8	-	-	-	66.1
LDAM-DRW + CUDA	63.1$_{\pm0.1}$	48.0$_{\pm0.3}$	31.1$_{\pm0.2}$	51.4$_{\pm0.1}$	67.8$_{\pm0.2}$	67.6$_{\pm0.2}$	66.7$_{\pm0.3}$	67.2$_{\pm0.2}$
BS (Ren et al., 2020)	61.1$_{\pm0.2}$	48.5$_{\pm0.2}$	31.8$_{\pm0.4}$	50.9$_{\pm0.1}$	65.5$_{\pm0.2}$	67.5$_{\pm0.1}$	67.5$_{\pm0.1}$	67.2$_{\pm0.2}$
BS + CUDA	61.9$_{\pm0.1}$	49.2$_{\pm0.0}$	32.3$_{\pm0.4}$	51.6$_{\pm0.1}$	67.6$_{\pm0.1}$	68.3$_{\pm0.1}$	68.3$_{\pm0.1}$	68.2$_{\pm0.1}$
RIDE (3 experts) (Wang et al., 2021)⋆	64.8$_{\pm0.1}$	50.8$_{\pm0.2}$	34.6$_{\pm0.2}$	53.6$_{\pm0.1}$	70.4$_{\pm0.1}$	71.8$_{\pm0.1}$	71.7$_{\pm0.1}$	71.6$_{\pm0.1}$
RIDE + CMO (Park et al., 2022)†	65.6	50.6	34.8	54.0	68.0	70.6	<u>72.0</u>	70.9
RIDE (3 experts) + CUDA⋆	65.9$_{\pm0.1}$	51.7$_{\pm0.2}$	34.9$_{\pm0.2}$	54.7$_{\pm0.1}$	70.7$_{\pm0.2}$	<u>72.5</u>$_{\pm0.1}$	**72.7**$_{\pm0.2}$	**72.4**$_{\pm0.2}$
BCL (Zhu et al., 2022)	65.3$_{\pm0.2}$	<u>53.5</u>$_{\pm0.2}$	<u>36.3</u>$_{\pm0.3}$	<u>55.6</u>$_{\pm0.2}$	69.5$_{\pm0.1}$	72.4$_{\pm0.2}$	71.7$_{\pm0.1}$	71.8$_{\pm0.1}$
BCL + CUDA	<u>66.8</u>$_{\pm0.1}$	**53.9**$_{\pm0.3}$	**36.6**$_{\pm0.2}$	**56.3**$_{\pm0.1}$	<u>70.9</u>$_{\pm0.2}$	**72.8**$_{\pm0.1}$	<u>72.0</u>$_{\pm0.1}$	<u>72.3</u>$_{\pm0.1}$

Experiment

Table 3: Comparison for CIFAR-LT-100 performance on ResNet-32 with 400 epochs.

Algorithm	Imbalance Ratio	
	100	50
PaCo	52.0	56.0
BCL	52.6	57.2
NCL	54.2	58.2
BCL + CUDA	53.5	57.4
NCL + CUDA	**54.8**	**59.6**

Analysis

模式识别与神经计算研究组
PAttern Recognition and NEural Computing

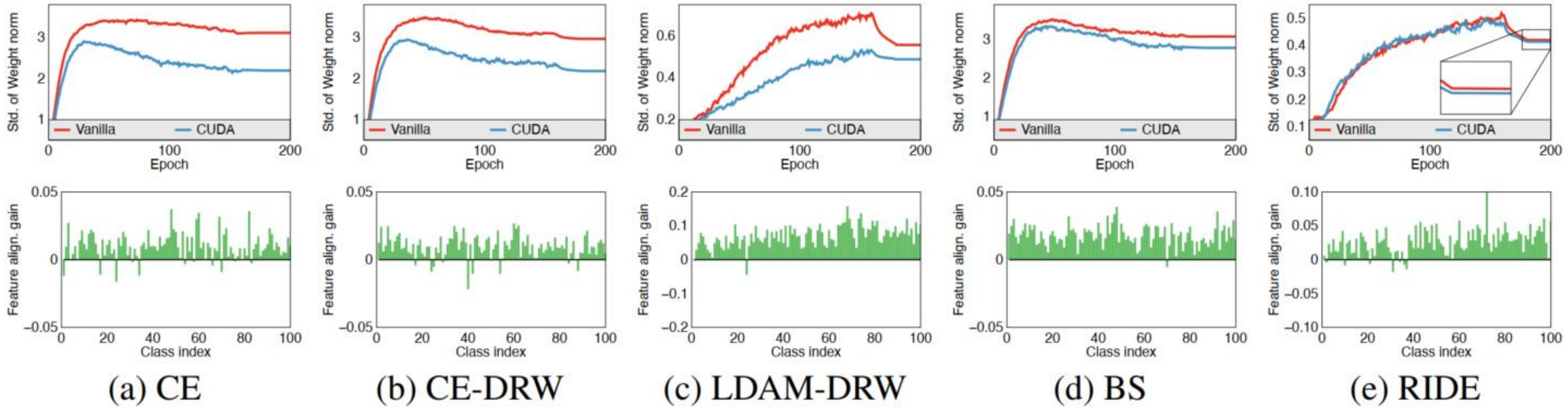


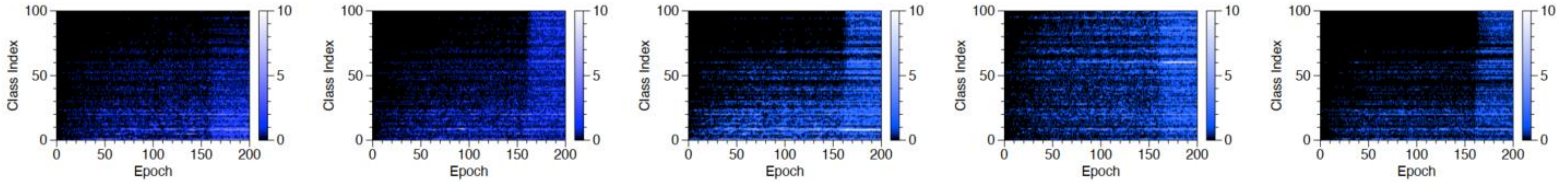
Figure 3: Analysis of how CUDA improves long-tailed recognition performance, classifier weight norm (top row) and feature alignment gain (bottom row) of the CIFAR-100-LT validation set. Notably that weight norm and feature alignment represent class-wise weight magnitude of classifier and ability of feature extractor, respectively. The detailed analysis is described in Section $\boxed{4.3}$.

Experiment

Table 4: Augmentation analysis on CIFAR-100-LT with IR 100. AA (Cubuk et al., 2019), FAA (Lim et al., 2019), DADA (Li et al., 2020b), and RA (Cubuk et al., 2020) with $n = 1, m = 2$ policies are used. C, S, I represent CIFAR, SVHN, and ImageNet policy.

	Vanilla	AA			FAA		DADA		RA	CUDA
		C	S	I	C	I	C	I		
CE	38.7	41.7	40.7	40.1	<u>42.3</u>	40.8	41.0	41.2	40.5	**42.7**
CE-DRW	41.4	<u>46.5</u>	44.7	45.5	46.3	44.8	45.6	45.7	45.8	**47.4**
LDAM-DRW	42.5	<u>47.0</u>	44.7	44.9	46.6	45.6	45.9	46.5	44.0	**47.2**
BS	43.3	<u>47.0</u>	46.1	45.5	46.5	45.0	45.0	46.9	45.2	**47.7**
RIDE (3 experts)	49.7	49.5	47.3	45.5	49.8	<u>50.6</u>	50.4	50.5	47.9	**50.7**

Experiment



(a) CE (b) CE-DRW (c) LDAM-DRW (d) BS (e) RIDE

Figure 4: Evolution of LoL score on various algorithms, CE, CE-DRW, LDAM-DRW, BS, and RIDE.

模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

Kill Two Birds with One Stone: Rethinking Data Augmentation for Deep Long-tailed Learning

Binwu Wang[2], Pengkun Wang[2,3*], Wei Xu[2], Xu Wang[2,3], Yudong Zhang[2], Kun Wang[2], Yang Wang[1,2,3*]
[1]Key Laboratory of Precision and Intelligent Chemistry, University of Science and Technology of China, Hefei, China
[2]University of Science and Technology of China (USTC), Hefei, China
[3]Suzhou Institute for Advanced Research, USTC, Suzhou, China
wbw1995@mail.ustc.edu.cn, pengkun@ustc.edu.cn
weixu1223@mail.ustc.edu.cn, wx309@ustc.edu.cn
{zyd2020,wk520529}@mail.ustc.edu.cn, angyan@ustc.edu.cn

ICLR 2024

Motivation

Whether it is optimal to utilize the same kind of augmentation strategy on all classes for long-tailed learning.

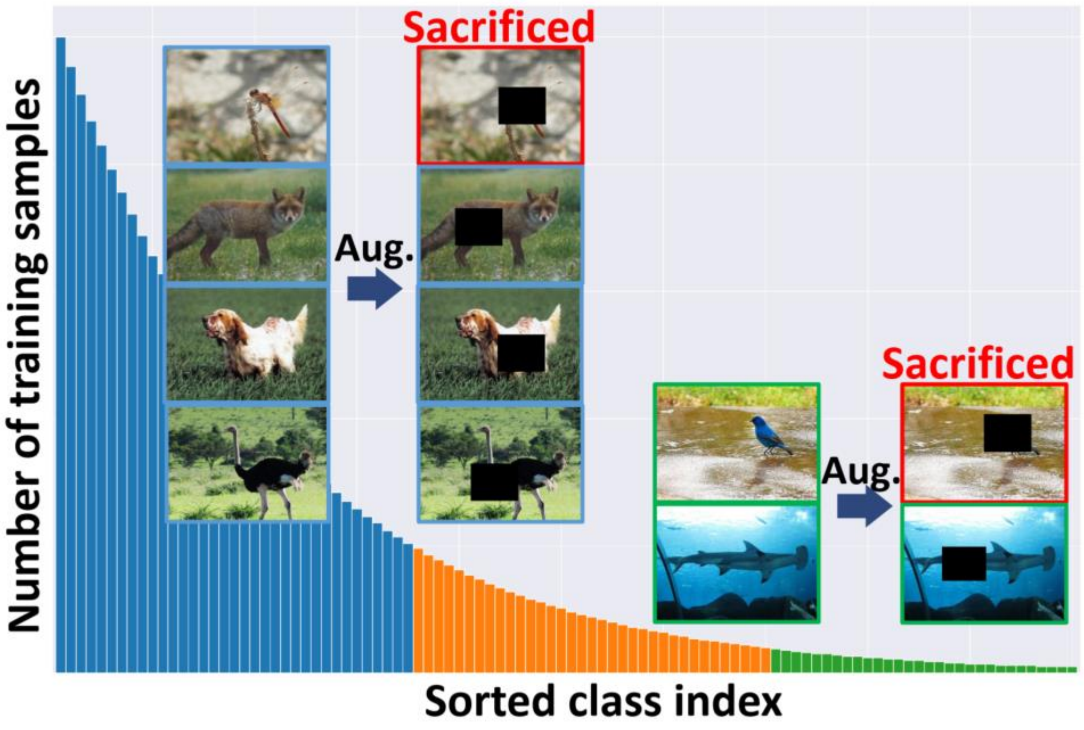


Figure 1: Motivation of DODA. In deep long-tailed learning, traditional data augmentation can significantly improve the average performance of the model, but it will potentially sacrifice certain classes (i.e., red box), especially tail classes.

Two ‘birds’ co-exist in long-tailed learning:
(1) inherent data-wise imbalance;
(2) extrinsic augmentation-wise imbalance.

The former is the inherent property of real-world data distributions, while the latter is the side effect caused by DA when improving model generalization.

Analysis

This overall improvement is encouraging, but it also prompts us to think about how DA brings gains to the problem.

Theorem 1. *Augmented samples produced by $\mathcal{O}(\cdot)$ do not respect the level-set of f^*. When we approximate the ideal model f^* by minimizing the training loss (i.e., 0 training error), the latter tends to zero, while the former is greater than zero due to the augmented samples deviate from the level set of f^*. In this case, the trained model f_θ is biased compared to the ideal model f^*.*

$$\sum_{(x,y)\in\mathcal{D}} \mathbb{E}[||y - f^*(\mathcal{O}(x))||_2^2] > 0 \wedge \sum_{(x,y)\in\mathcal{D}} \mathbb{E}[||y - f_\theta(\mathcal{O}(x))||_2^2] = 0 \Longrightarrow \textit{bias} \tag{1}$$

Theorem 2. *Under the long-tailed distribution, minimizing the training loss (i.e., 0 training error) is equivalent to minimizing the training loss for each class. For class c and augmented samples $\{(\mathcal{O}(x), y)|y = c, (\mathcal{O}(x), y) \in \mathcal{D}\}$, the ideal trained model can minimize the training loss of class c, but when $\mathcal{O}(x)$ deviates from the level-set of f^*, DA will cause irreducible class-wise bias in f_θ.*

$$\sum_{(x,y)\in\mathcal{D}|y=c} \mathbb{E}[||y - f^*(\mathcal{O}(x))||_2^2] > 0 \wedge \sum_{(x,y)\in\mathcal{D}|y=c} \mathbb{E}[||y - f_\theta(\mathcal{O}(x))||_2^2] = 0 \Longrightarrow \textit{class-wise bias} \tag{2}$$

Analysis

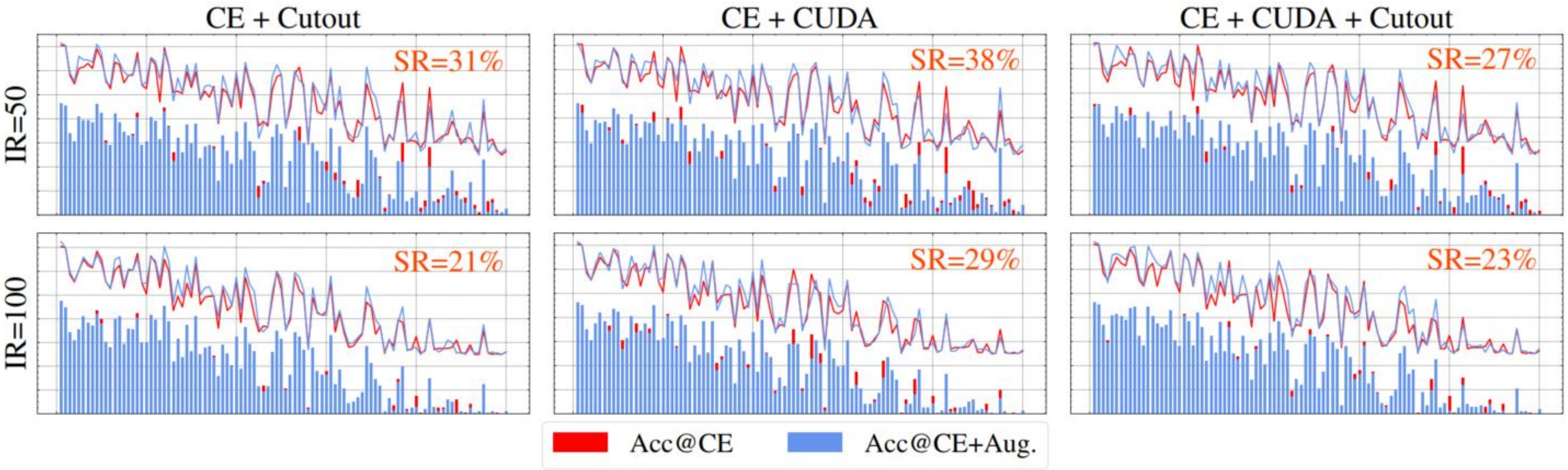


Figure 2: Impact of class-independent long-tailed DAs on classification accuracy on CIFAR-100-LT.

Key Observations:

① All three types of DA can improve the average model performance;

② The improvement in average performance inevitably sacrifices some classes, whether they are head or tail classes;

③ The phenomenon of sacrificing classes is especially evident on tail classes.

Analysis

DA demonstrates its effectiveness by pleasing the 'strong' and exploiting the 'weak'!

The distribution span can be expressed as follows:

$$\mathbb{S}_c \Rightarrow (\mathbb{X}-\mathbb{X}_c)^2-(\mathbb{Y}-\mathbb{Y}_c)^2=\mathbb{R}_c{}^2$$

The distribution span after DA can be defined as follows:

$$\bar{\mathbb{S}}_{c_h} \Rightarrow (\mathbb{X}-\mathbb{X}_{c_h})^2-(\mathbb{Y}-\mathbb{Y}_{c_h})^2=(\mathbb{R}_{c_h}+\Delta_{c_h})^2$$
$$\bar{\mathbb{S}}_{c_t} \Rightarrow (\mathbb{X}-\mathbb{X}_{c_t})^2-(\mathbb{Y}-\mathbb{Y}_{c_t})^2=(\mathbb{R}_{c_t}+\Delta_{c_t})^2$$

The augmentation sensitivity ψ can be defined as the ratio of the marginal space to the base space, Under the same DA, tail classes have a higher augmentation sensitivity, indicating that tail classes are more sensitive to the marginal space.

$$\psi_{c_t}-\psi_{c_h}=2\Delta_{c_h}\cdot\frac{\mathbb{R}_{c_h}-\mathbb{R}_{c_t}}{\mathbb{R}_{c_h}\mathbb{R}_{c_t}}+\Delta_{c_h}{}^2\cdot\frac{\mathbb{R}_{c_h}{}^2-\mathbb{R}_{c_t}{}^2}{\mathbb{R}_{c_h}{}^2\mathbb{R}_{c_t}{}^2}>0$$

Methods

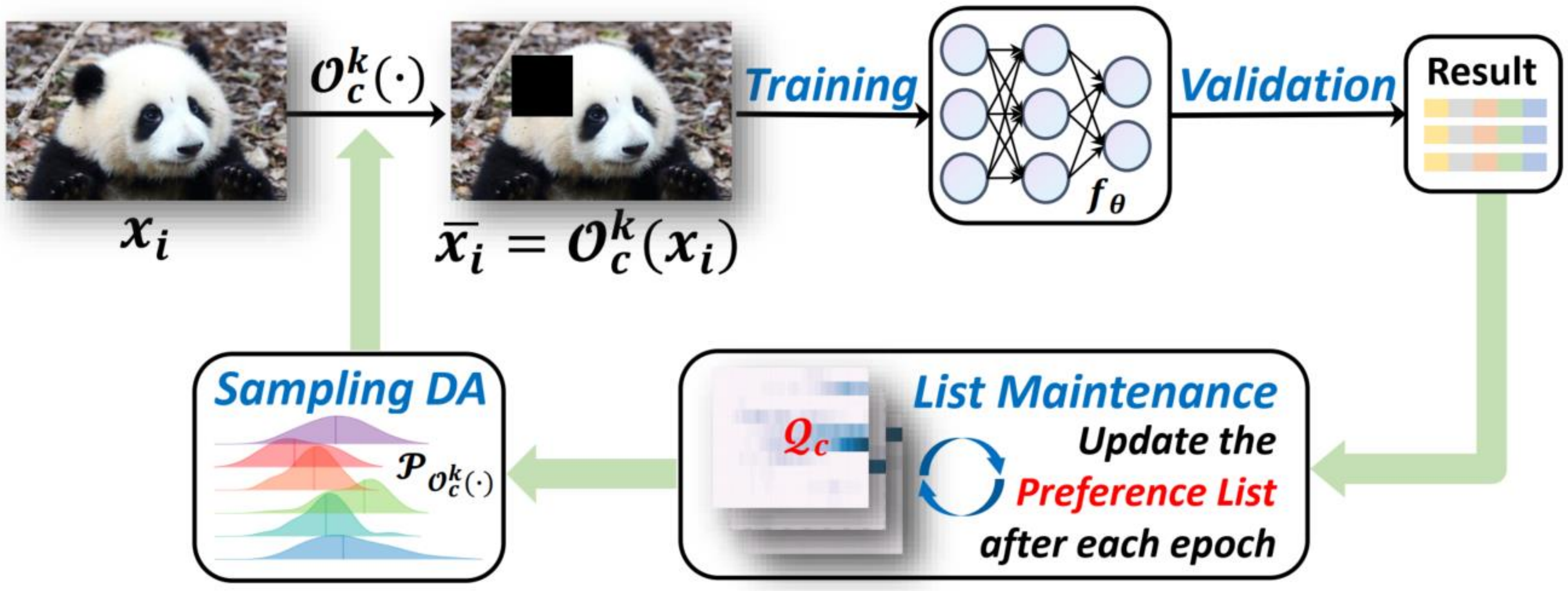


Figure 3: Overview of DODA. DODA allows each class to choose appropriate augmentation methods during training by maintaining a 'preference list' for each class.

Methods

for $c \leq C$ **do**

Randomly select a DA $\mathcal{O}_c^k(\cdot)$ for class c according to the weight distribution $\mathcal{Q}_c$.

$$\mathcal{P}_{\mathcal{O}_c^k(\cdot)} = \frac{|\mathcal{Q}_c^k|}{\sum_{j \in \mathcal{Q}_c} |j|}$$

where $\mathcal{Q}_c = \{\mathcal{Q}_c^1, \mathcal{Q}_c^2, \ldots, \mathcal{Q}_c^K\}$

end

Generate $\bar{\mathcal{D}} = \{(\bar{x_i}, y_i) | (x_i, y_i) \in \mathcal{D}\}$ where

$$\bar{x_i} = \begin{cases} \mathcal{O}_c^k(x_i), & \text{with prob. } p_{aug} \\ x_i, & \text{otherwise.} \end{cases}$$

Run LTL algorithm $\mathcal{F}$ using $\bar{\mathcal{D}}$, i.e., $\mathcal{F}(f_\theta, \bar{\mathcal{D}})$.

for $c \leq C$ **do**

Compute positive sample size $\nabla_{z_c^k}^{pos}$ for class c

$$\nabla_{z_c^k}^{pos} = \sum_{(\bar{x_i}, y_i) \in \bar{\mathcal{D}} | y_i = c} \mathbb{1}_{\{f_\theta(\bar{x_i}) = c\}}$$

if $\nabla_{z_c^k}^{pos} > temp_c$ **then** $\mathcal{Q}_c^k \longleftarrow \mathcal{Q}_c^k + 1$

// Up-weight the positive DA

elif $\nabla_{z_c^k}^{pos} = temp_c$ **then** $\mathcal{Q}_c^k \longleftarrow \mathcal{Q}_c^k$

else // Down-weight the negative DA

if $\mathcal{Q}_c^k > 1$ **then** $\mathcal{Q}_c^k \longleftarrow \mathcal{Q}_c^k - 1$

else $\mathcal{Q}_c^k \longleftarrow 1$

$temp_c = \nabla_{z_c^k}^{pos}$

end

Experiment

Table 1: Accuracy (%) on CIFAR-100-LT dataset (Imbalance ratio={10, 50, 100}) wtih state-of-the-art methods. **Blod** indicates the best performance while underline indicates the second best. (+) and (-) indicate the the relative gain. We report the average results of three random trials.

Method	IR=10				IR=50				IR=100			
	Head	Medium	Tail	All	Head	Medium	Tail	All	Head	Medium	Tail	All
CE He et al. (2016)	63.2	40.3	-	56.5 (+0.0)	63.9	36.2	15.2	43.8 (+0.0)	65.6	36.2	8.2	38.1 (+0.0)
CE + CMO Park et al. (2022)	67.0	45.0	-	60.2 (+3.7)	68.6	37.8	18.7	47.0 (+3.2)	70.1	40.6	10.3	41.8 (+3.7)
CE + CUDA Ahn et al. (2023)	66.8	43.1	-	59.5 (+3.0)	68.3	38.4	13.7	46.2 (+2.4)	70.7	41.4	9.3	42.0 (+3.9)
CE + CMO + CUDA Ahn et al. (2023)	65.7	43.0	-	58.7 (+2.2)	67.6	39.2	17.6	47.0 (+3.2)	71.1	43.4	11.7	43.6 (+5.5)
CE + DODA	67.0	44.0	-	59.9 (+3.4)	**71.2**	40.3	12.6	48.0 (+4.2)	**74.8**	43.8	10.0	44.5 (+6.4)
CE + CMO + DODA	**67.1**	42.5	-	59.5 (+3.0)	70.4	41.2	19.7	49.3 (+5.5)	73.2	44.4	12.4	44.9 (+6.8)
CE-DRW Cao et al. (2019)	62.5	48.6	-	58.2 (+0.0)	60.6	39.0	22.9	45.0 (+0.0)	63.4	41.2	15.7	41.4 (+0.0)
CE-DRW + CUDA Ahn et al. (2023)	64.2	56.2	-	61.7 (+3.5)	63.8	48.0	37.0	52.5 (+7.5)	63.5	48.9	25.3	46.9 (+5.5)
CE-DRW + DODA	63.6	56.7	-	61.5 (+3.3)	63.4	47.4	38.9	52.5 (+7.5)	60.2	**51.9**	29.6	48.1 (+6.7)
LDAM-DRW Cao et al. (2019)	62.7	46.1	-	57.5 (+0.0)	63.0	41.2	25.1	47.2 (+0.0)	62.8	42.6	21.1	43.2 (+0.0)
LDAM-DRW + CUDA Ahn et al. (2023)	63.6	45.2	-	57.9 (+0.4)	66.2	46.2	26.4	50.8 (+3.6)	66.0	49.5	22.1	47.1 (+3.9)
LDAM-DRW + DODA	63.3	45.8	-	57.9 (+0.4)	64.7	46.3	27.5	50.5 (+3.3)	65.4	50.8	25.5	48.3 (+5.1)
BS Ren et al. (2020)	61.5	50.6	-	58.1 (+0.0)	60.3	41.3	34.3	47.9 (+0.0)	59.6	42.3	23.7	42.8 (+0.0)
BS + CUDA Ahn et al. (2023)	64.2	55.5	-	61.5 (+3.4)	63.6	48.4	37.3	52.7 (+4.8)	62.5	49.1	29.4	47.9 (+5.1)
BS + DODA	64.0	**56.8**	-	61.8 (+3.7)	62.2	**51.2**	**41.5**	**54.0** (+6.1)	63.1	49.3	31.2	48.7 (+5.9)
RIDE (3 experts) Wang et al. (2021)	66.4	49.4	-	61.1 (+0.0)	65.7	47.7	31.8	52.2 (+0.0)	65.7	48.6	25.0	47.5 (+0.0)
RIDE + CMO Park et al. (2022)	65.3	48.5	-	60.1 (-1.0)	67.8	47.0	33.4	53.1 (+0.9)	67.9	51.2	27.6	50.0 (+2.5)
RIDE (3 experts) + CUDA Ahn et al. (2023)	65.6	47.2	-	59.9 (-1.2)	68.2	46.1	29.3	52.1 (-0.1)	68.7	50.9	25.7	49.6 (+2.1)
RIDE (3 experts) + DODA	65.7	49.9	-	60.8 (-0.3)	67.0	46.5	33.6	52.6 (+0.4)	68.4	51.1	27.8	50.2 (+2.7)
BCL Zhu et al. (2022)	62.2	51.8	-	58.9 (+0.0)	61.6	43.1	34.3	49.1 (+0.0)	63.1	42.9	23.9	44.2 (+0.0)
BCL + CUDA Ahn et al. (2023)	65.3	56.6	-	62.6 (+3.7)	64.0	47.4	39.4	52.7 (+3.6)	64.7	49.7	29.1	48.8 (+4.6)
BCL + DODA	65.6	56.1	-	**62.7** (+3.8)	64.9	48.0	40.6	53.6 (+4.5)	66.0	50.7	**33.8**	**51.0** (+6.8)

Experiment

Table 2: Accuracy (%) on ImageNet-LT and iNaturalist 2018 datasets wtih state-of-the-art methods.

Method	ImageNet-LT				iNaturalist 2018			
	Head	Medium	Tail	All	Head	Medium	Tail	All
CE He et al. (2016)	64.0	33.8	5.8	41.6 (+0.0)	73.9	63.5	55.5	61.0 (+0.0)
CE + CUDA Ahn et al. (2023)	67.1	47.1	13.4	47.2 (+5.6)	74.6	65.0	57.2	62.5 (+1.5)
CE + DODA	**67.4**	47.5	13.9	48.1 (+6.5)	**74.9**	66.0	58.4	63.6 (+2.6)
CE-DRW Cao et al. (2019)	61.7	47.3	28.8	50.1 (+0.0)	68.2	67.3	66.4	67.0 (+0.0)
CE-DRW + CUDA Ahn et al. (2023)	61.7	48.4	30.5	51.1 (+1.0)	68.8	67.9	66.5	67.4 (+0.4)
CE-DRW + DODA	62.4	48.5	31.3	52.2 (+2.1)	69.0	68.2	67.8	68.2 (+1.2)
LWS Kang et al. (2020)	57.1	45.2	29.3	47.7 (+0.0)	65.0	66.3	65.5	65.9 (+0.0)
cRT Kang et al. (2020)	58.8	44.0	26.1	47.3 (+0.0)	69.0	66.0	63.2	65.2 (+0.0)
cRT + CUDA Ahn et al. (2023)	62.3	47.2	28.1	50.2 (+2.9)	68.2	67.9	66.4	67.3 (+2.1)
cRT + DODA	62.8	47.7	28.9	51.3 (+3.6)	69.2	68.3	67.6	68.5 (+3.3)
LDAM-DRW Cao et al. (2019)	60.4	46.9	30.7	49.8 (+0.0)	-	-	-	66.1 (+0.0)
LDAM-DRW + CUDA Ahn et al. (2023)	63.2	48.2	31.2	51.5 (+1.7)	68.0	67.5	66.8	67.3 (+1.2)
LDAM-DRW + DODA	63.7	48.6	31.9	52.4 (+2.6)	68.6	68.1	67.9	68.7 (+2.6)
BS Ren et al. (2020)	60.9	48.8	32.1	51.0 (+0.0)	65.7	67.4	67.5	67.3 (+0.0)
BS + CUDA Ahn et al. (2023)	61.8	49.1	31.8	51.5 (+0.5)	67.6	68.2	68.3	68.2 (+0.9)
BS + DODA	61.9	49.5	32.4	52.0 (+1.0)	68.1	68.9	69.5	69.4 (+2.1)
RIDE (3 experts) Wang et al. (2021)	64.9	50.4	34.4	53.6 (+0.0)	70.4	71.8	71.8	71.6 (+0.0)
RIDE + CMO Park et al. (2022)	65.6	50.6	34.8	54.0 (+0.4)	68.0	70.6	72.0	70.9 (-0.7)
RIDE (3 experts) + CUDA Ahn et al. (2023)	66.0	51.7	34.7	54.7 (+1.1)	70.6	72.6	72.7	72.4 (+1.4)
RIDE (3 experts) + DODA	66.6	51.9	35.9	55.8 (+2.2)	70.9	72.4	**73.9**	**73.7** (+2.8)
BCL Zhu et al. (2022)	65.3	53.5	36.3	55.6 (+0.0)	69.4	72.4	71.8	71.8 (+0.0)
BCL + CUDA Ahn et al. (2023)	66.8	53.9	36.6	56.3 (+0.7)	70.8	72.7	72.0	72.2 (+0.4)
BCL + DODA	66.9	**54.1**	**37.4**	**56.9** (+1.3)	71.2	**73.2**	73.4	**73.7** (+1.9)

Experiment

Does DODA make fewer classes be ‘sacrificed’?

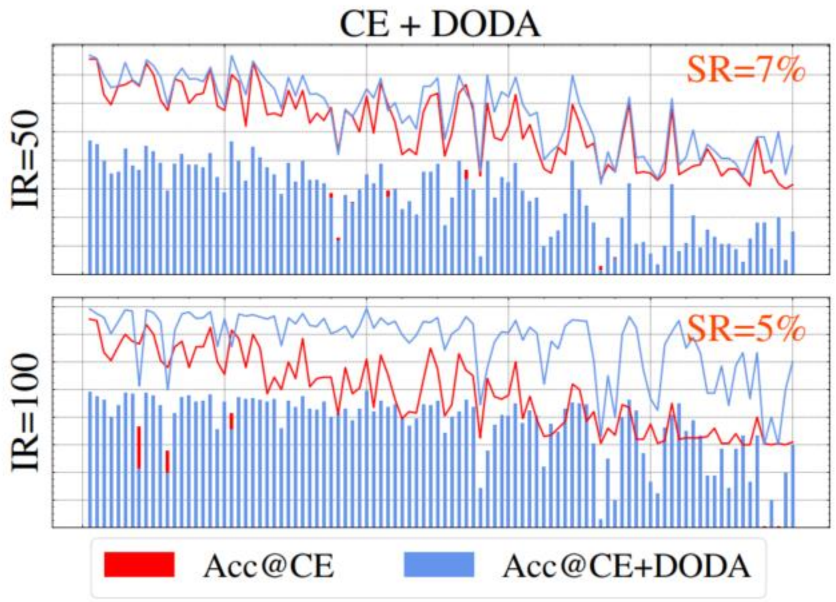


Figure 4: Visualization of the accuracy of each class between CE and CE with CUDA.

Experiment

Why DODA can alleviate the long-tailed problem?

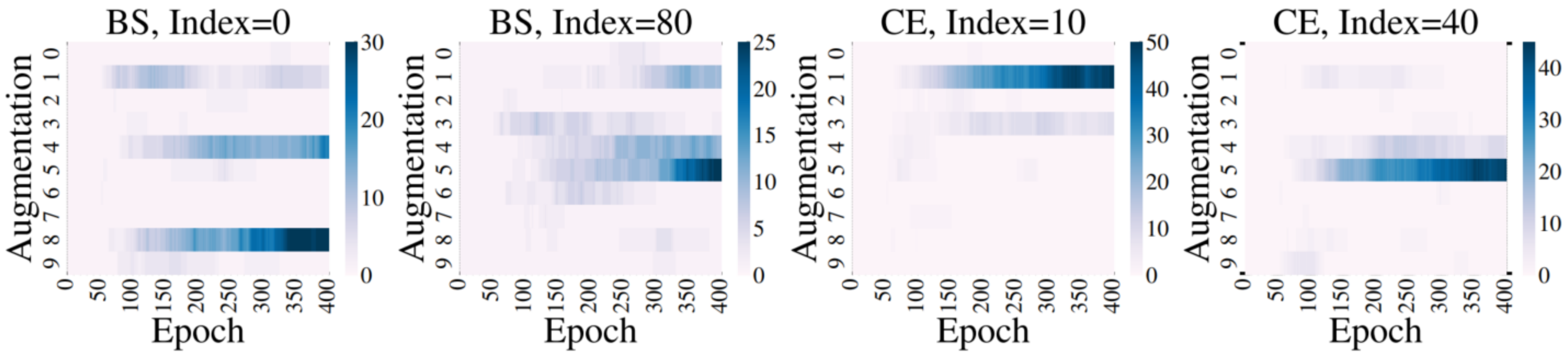


Figure 5: Trend of the selection hierarchies during training.

Experiment

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

What are the trends in DAs favored by classes?

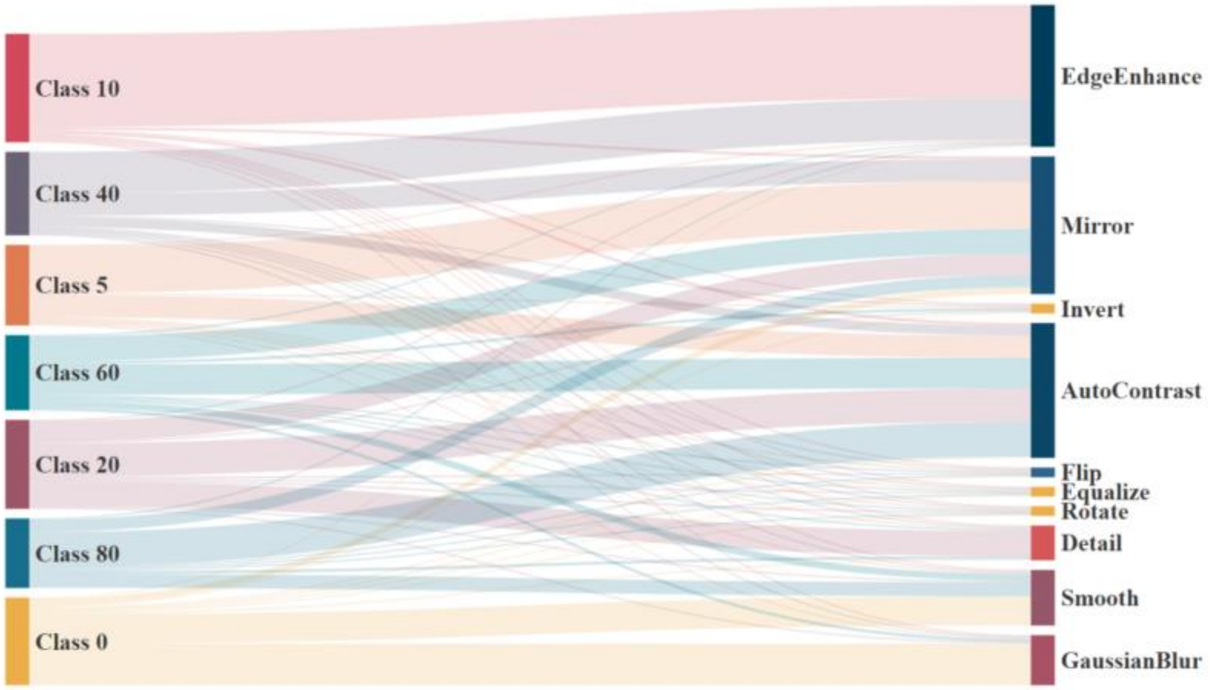


Figure 6: Class and 'its' preferred DA on BS and CIFAR-100-LT (IR=100).

Experiment

Would other augmentation methods be better?

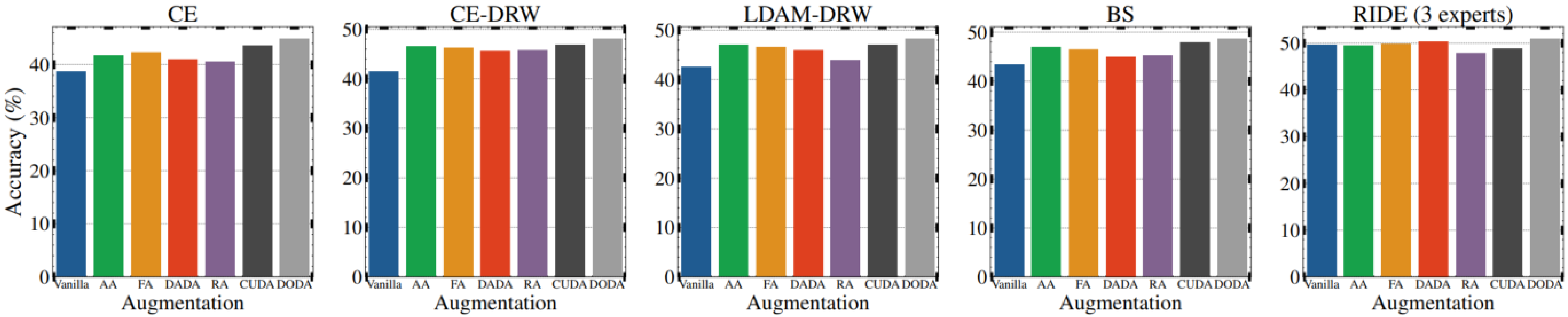
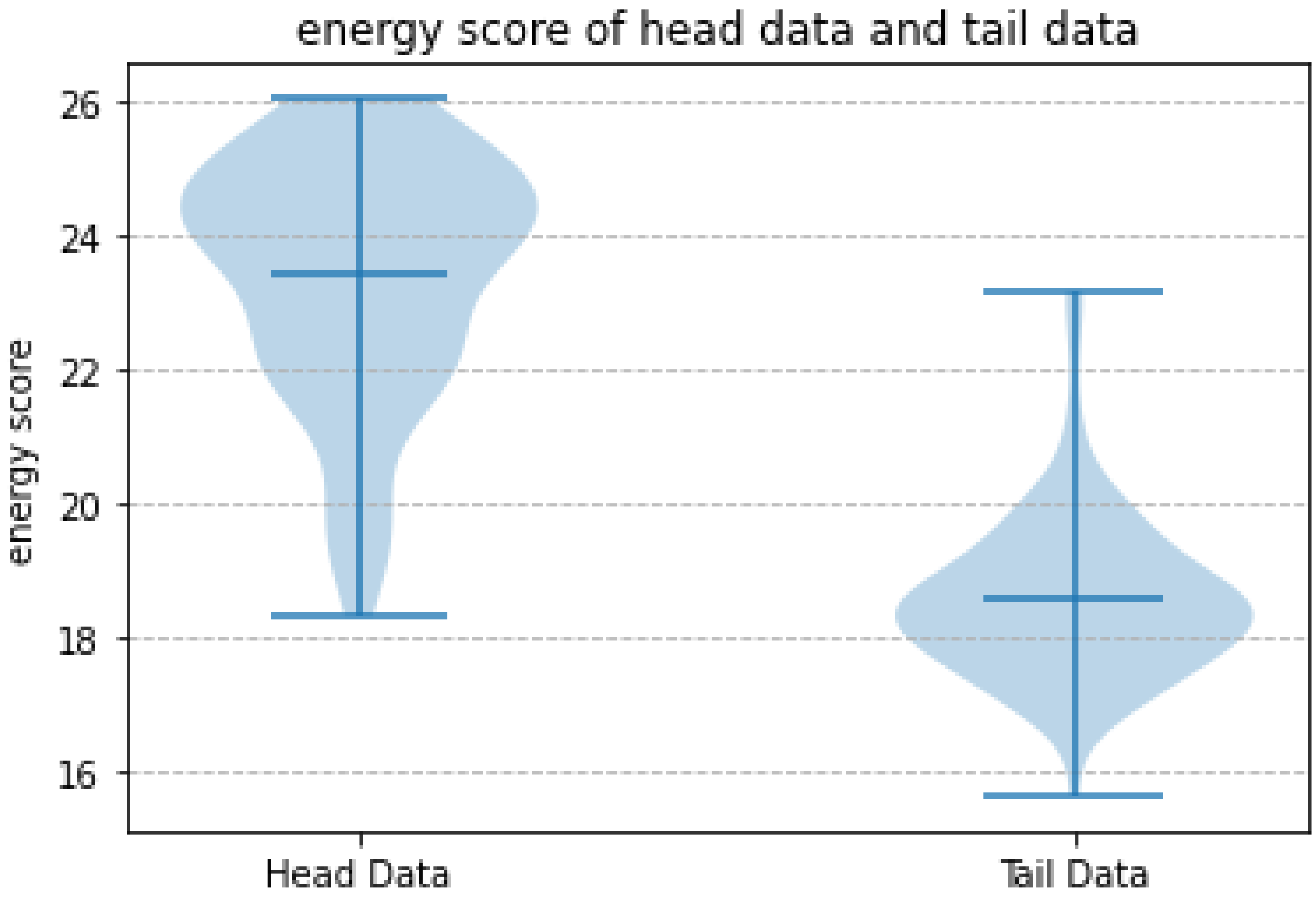


Figure 7: DA analysis on various algorithms, CE, CE-DRW, LDAM-DRW, BS, and RIDE.

Thanks


energy score of head data and tail data
energy score
26
24
22
20
18
16
Head Data
Tail Data