Delving into Semantic Scale Imbalance

Authors: Yanbiao Ma, Licheng Jiao, Fang Liu, Yuxin Li, Shuyuan Yang, Xu Liu, Yanbiao Ma

Date: 2023-02-16T00:00:00+00:00

Abstract

Model bias triggered by long-tailed data has been widely studied. However, measure based on the number of samples cannot explicate three phenomena simultaneously: (1) Given enough data, the classification performance gain is marginal with additional samples. (2) Classification performance decays precipitously as the number of training samples decreases when there is insufficient data. (3) Model trained on sample-balanced datasets still has different biases for different classes. In this work, we define and quantify the semantic scale of classes, which is used to measure the feature diversity of classes. It is exciting to find experimentally that there is a marginal effect of semantic scale, which perfectly describes the first two phenomena. Further, the quantitative measurement of semantic scale imbalance is proposed, which can accurately reflect model bias on multiple datasets, even on sample-balanced data, revealing a novel perspective for the study of class imbalance. Due to the prevalence of semantic scale imbalance, we propose semantic-scale-balanced learning, including a general loss improvement scheme and a dynamic re-weighting training framework that overcomes the challenge of calculating semantic scales in real-time during iterations. Comprehensive experiments show that dynamic semantic-scale-balanced learning consistently enables the model to perform superiorly on large-scale long-tailed and non-long-tailed natural and medical datasets, which is a good starting point for mitigating the prevalent but unnoticed model bias. In addition, we look ahead to future challenges.

Full Text

Preamble

Published as a conference paper at ICLR 2023

Delving into Semantic Scale Imbalance

Yanbiao Ma, Licheng Jiao, Fang Liu, Yuxin Li, Shuyuan Yang, Xu Liu

Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, Xidian University, Xi' an, 710071, China

{ybmamail,yxli_{12}}@stu.xidian.edu.cn, lchjiao@mail.xidian.edu.cn, f63liu@163.com, syyang@xidian.edu.cn, xuliu361@163.com

Abstract

Model bias triggered by long-tailed data has been widely studied. However, measures based on sample count cannot simultaneously explain three phenomena: (1) Given sufficient data, classification performance gains become marginal with additional samples; (2) Classification performance decays precipitously as training samples decrease when data is insufficient; and (3) Models trained on sample-balanced datasets still exhibit different biases across classes.

In this work, we define and quantify the semantic scale of classes to measure feature diversity. Experimentally, we discover a marginal effect of semantic scale that perfectly describes the first two phenomena. Furthermore, we propose a quantitative measurement of semantic scale imbalance that accurately reflects model bias across multiple datasets, even on sample-balanced data, revealing a novel perspective for studying class imbalance.

Given the prevalence of semantic scale imbalance, we propose semantic-scale-balanced learning, comprising a general loss improvement scheme and a dynamic re-weighting training framework that overcomes the challenge of calculating semantic scales in real-time during iterations. Comprehensive experiments demonstrate that dynamic semantic-scale-balanced learning consistently enables superior model performance on large-scale long-tailed and non-long-tailed natural and medical datasets, providing a strong starting point for mitigating this prevalent yet overlooked model bias. We also discuss future challenges.

1 Introduction

In practical tasks, long-tailed class imbalance is a common problem where numerical imbalance causes trained models to become biased toward dominant head classes and perform poorly on tail classes [**?**]. However, what is overlooked is that models trained on sample-balanced data still show different biases across classes. This model bias is not addressed by existing class imbalance studies and cannot be ameliorated by current methods proposed for long-tailed data [**?**]. For natural datasets, classes artificially divided by different semantic concepts correspond to different semantic scales, which can lead to varying degrees of optimization when the deep metric is single-scale [**?**]. In this study, we attempt to uncover more information from the data itself by introducing and quantifying semantic scale imbalance to represent more general model bias.

We further propose semantic-scale-balanced learning to improve loss functions and mitigate model bias. Classes corresponding to different semantic concepts

exhibit different feature diversity, which we equate with semantic scale. Typically, the finer the semantic concept of a class label, the less rich the feature diversity, the less information a model can extract, and the worse the model performs on that class [**?**]. The manifold distribution hypothesis [**?**] states that natural data from a specific class concentrates on a low-dimensional manifold. The larger the range of value variation along specific dimensions of the manifold (such as illumination and angle), the richer the features and the larger the manifold volume. For example, in Figure 1 [Figure 1: see original paper], since "Swan" is a subclass of "Bird," its semantic concept is finer, so "Bird" has richer feature diversity than "Swan," and its corresponding manifold volume is larger.

Semantic scale can be measured by manifold volume. We present a reliable and numerically stable quantitative measurement of semantic scale and further define semantic scale imbalance. To avoid confusion, we refer to the volume calculated in sample space as sample volume and that in feature space as feature volume. Our innovative study of semantic scale imbalance can simultaneously and naturally explain two phenomena that existing class imbalance studies cannot: (1) As sample count increases linearly, model performance shows rapid early improvement followed by plateauing [**?**]; and (2) Even models trained on sample-balanced datasets suffer from class bias.

Experiments demonstrate that semantic scale imbalance is more prevalent in natural datasets than sample count imbalance, extending the scope of class imbalance studies to arbitrary datasets.

How can we mitigate the adverse effects of semantic scale imbalance? Inspired by long-tailed classification methods, current solutions for class imbalance typically adopt re-sampling strategies [**?**] and cost-sensitive learning [**?**]. However, re-sampling may introduce numerous duplicate samples, making models susceptible to overfitting when oversampling, or discard valuable samples when undersampling. Therefore, drawing on classical re-weighting strategies [**?**], we propose dynamic semantic-scale-balanced learning. Its core idea is to dynamically measure the degree of imbalance between semantic scales in feature space rather than using fixed measures, enabling dynamic evaluation of weaker classes and assigning greater weights to their corresponding losses.

Our key contributions are: (1) We propose the novel idea of leveraging manifold volume to measure semantic scale (Sec 3.2). We innovatively discover that semantic scale exhibits marginal effects (Sec 3.3) and that dataset semantic scale trends are highly consistent with model performance trends. (2) We introduce and define semantic scale imbalance to measure class imbalance by semantic scale rather than sample count, revealing a new perspective for studying class imbalance. Experiments show semantic scale imbalance is prevalent in datasets and more accurately reflects model bias affecting performance (Sec 3.4). (3) We propose semantic-scale-balanced learning to mitigate model bias, including a general loss improvement scheme (Sec 4.1) and a dynamic re-weighting training framework (Sec 4.2) that overcomes the challenge of calculating semantic scales in real-time during iterations. Comprehensive experiments demonstrate

that semantic-scale-balanced learning applies to various datasets and achieves significant performance gains on multiple vision tasks (Sec 5).

2 Slow Drift Phenomenon of Features and Marginal Effect

Since model parameters change during training, [**?**] studied embedding drift speed by measuring feature differences for the same instance across training iterations. Experiments on the Stanford Online Products (SOP) dataset [**?**] show that features change drastically in early training stages, become relatively stable after traversing the dataset twice, and drift extremely slowly when the learning rate decreases. This justifies using historical features to calculate semantic scales.

Marginal effect [**?**] describes that in early training stages, networks learn features quickly, but due to information overlap among samples, data information gradually saturates as sample count increases, diminishing performance improvements from new samples. The effective number of samples represents data information, but its limitation is that it fails on sample-balanced datasets.

Assume each sample has unit volume 1 and define the set of all samples for a class as Ω with volume N and N$ $1. A new sample may overlap with previous samples, with overlap probability P and non-overlap probability 1−P. As data information increases, probability P becomes higher.

Define the effective number of samples as En, where En = 1 + β 1−βn−1 [**?**], n denotes sample count, and hyperparameter β = N −1 N [0, 1) controls En' s growth rate. When N = 1, β = 0 and En = 1, meaning all samples can be represented by a single prototype via data augmentation.

When N → ∞, β → 1, implying no overlap, then lim which indicates the effective number of samples does not increase faster than sample count, though this contradicts our experimental results. (1−βn) (1−β) = lim = 1−βn −nβn−1 = lim The effective number of samples En is an exponential function of sample count n. Hyperparameters β for classes of different granularity should differ. However, β selection requires more data information, which [**?**] does not address, forcing the assumption that β is identical for all classes. In this case, compared to sample count, En simply uses an exponential function to obtain smoother weights. Furthermore, when sample counts are balanced, En is identical for each class and cannot mitigate model bias, so we attempt to mine data information (or feature diversity) for each class to facilitate imbalance problem study.

3 Semantic Scale Imbalance

This section first defines sample volume, feature volume, and semantic scale imbalance. Next, we derive a quantitative measurement of feature volume from singular value decomposition of the data matrix and information theory perspectives. Then, we investigate the marginal effect of semantic scale. Finally, we discuss the relationship between semantic scale imbalance and model bias.

3.1 Definitions

Different semantic concepts correspond to different semantic scales; for example, "Bird" has a larger scale than "Swan." For each class, we equate its feature diversity with its semantic scale and measure semantic scale by the volume of the subspace spanned by samples or features. Deep neural networks can be viewed as a combination of feature mapping function f(x,) and trained downstream classifier g(z), i.e., x → z( ) → y. Let samples of a class be X = [x1, x2, . . . , xm], and embeddings learned by deep neural networks be Z = {zi|zi = f(xi,) Rd, i = 1, 2, . . . , m}.

Definition 3.1. (Sample volume) The volume of the subspace spanned by sample set X.

Definition 3.2. (Feature volume) The volume of the subspace spanned by feature vectors Z.

Definition 3.3. (Semantic scale imbalance) A phenomenon of imbalance in semantic scale sizes measured by sample volume or feature volume.

3.2 Quantification of Semantic Scale

Given data X = [x1, x2, . . . , xm] and learned embeddings Z = [z1, z2, . . . , zm] Rd×m, where zi = f(xi,) Rd, i = 1, 2, . . . , m, we derive the volume of the subspace spanned by random vector zi (i.e., feature volume) below; sample volume can be calculated similarly (Appendix E). The covariance matrix of random vector zi is estimated as Σ = E mZZ T Rd×d, with eigenvalues $ $1 ≥ $ $2 ≥ · · · ≥ λd > 0 for the real symmetric matrix. Singular value decomposition (SVD) of Z yields Z = U ΣV T with singular values σj = pλj, j = 1, 2, . . . , d. The volume of the space spanned by vector zi is proportional to the product of all singular values of feature matrix Z, i.e., Vol(Z) After determinant expansion, the characteristic polynomial of matrix Σ is Φ(λ) = det(λI − Σ) = λd − (a11 + a22 + · · · add)λd−1 + · · · + (−1)d detΣ, and $ 1 $2 · · · λd = detΣ. Therefore, the volume of the space spanned by vector zi is proportional to the square root of the covariance matrix determinant of Z:

Vol(Z) ZZ T). The same result can be derived from the volume of a parallel hexahedron defined by vectors (Appendix G).

Considering that real-world metric tools typically have dynamic range—for example, a ruler has multiple scales (1mm, 1cm, or 10cm) to measure objects of different scales—we expect the quantitative measurement of feature volume to have multi-scale metric capability and therefore implement it using the sphere packing method [**?**], normally adopted in information theory. = zi + wi, where wi N (0, 2 The feature volume is positively correlated with sphere count, and since sphere packing has error at subspace boundaries, the error of finite feature vectors is assumed to be independent additive Gaussian noise: zi n I). The estimate of spheres with radius needed to pack the space spanned by all vectors is N =Vol (Z)/Vol (w). Metric scale adjustment can be achieved by tuning

sphere radius , thus controlling feature volume measurement results. Then the covariance matrix of vector zi is Σ = 2 m ZZ T Rd×d, such that Vol(Z) n I). The feature volume is proportional to N : m ZZ T) and Vol(w) det(2 det(2 Vol (Z) N = Vol (Z) Vol (w) (cid:16) I + n (cid:17) m 2 ZZ T s det (cid:0) 2 m ZZ T (cid:1) n I(cid:1) det (cid:0) 2 The feature zi dimension is d, so let n = d. To increase numerical stability, we perform logarithmic transformation of the above equation, which does not affect function monotonicity, m 2 ZZ T (cid:1). In practical training, it is and we can obtain Vol (Z) log essential to normalize feature vectors to zero mean. In this work, we set = 1000, and value does not affect the relative size of the space spanned by feature vectors of each class. The volume of the space spanned by Z can be written as: m 2 ZZ T (cid:1) = 1 log det (cid:0)I + d det (cid:0)I + n Vol(Z) = det(I + d (Z − Zmean)(Z − Zmean)T), where Zmean is the mean of Z, and Vol(Z) > 0 when sample count m > 1. We measure semantic scale S by feature volume, i.e., S = Vol(Z), where larger S indicates richer feature diversity, which we verify in Appendix C using multiple Stanford point cloud manifolds.

3.3 Marginal Effect of Semantic Scale

The marginal effect describes that feature richness gradually saturates as sample count increases, so semantic scale changes should also conform to marginal effects. Figure 2 [Figure 2: see original paper] illustrates that as sample count increases, semantic scale S measured by sample volume gradually saturates, indicating that the quantitative measurement of semantic scale behaves as expected.

Additionally, semantic scale growth rates vary across classes, determined by the granularity of the class itself. This leads to different semantic scales even when all classes have identical sample counts.

To investigate why adding samples yields marginal performance improvements when training samples are sufficient, we generate new training datasets for classification experiments using the following method: assume the original dataset has C total classes, and randomly select m samples from each class to form a sub-dataset with total sample count C × m. Sub-datasets generated based on CIFAR-10, CIFAR-100 [**?**], and Mini-ImageNet [**?**] are detailed in Appendix D.1.

We train ResNet-18 and ResNet-34 [**?**] on each of the 31 training sets in Table 7 , with the sum of semantic scales for all classes and corresponding top-1 accuracy shown in Figure 2. We are pleasantly surprised to find that when semantic scale increases rapidly, model performance improves swiftly alongside it, and when semantic scale becomes saturated, improvement is minimal.

3.4.1 Quantification of Semantic Scale Imbalance

Figure 2: Left column: curves of semantic scales with increasing sample count for the first ten classes from different datasets. Right column: for different sub-

datasets, curves of the sum of semantic scales for all classes and top-1 accuracy curves of trained ResNet-18 and ResNet-34. All models are trained using the Adam optimizer [**?**] with initial learning rate 0.01 decayed by 0.98 per epoch.

Table 1 : Pearson correlation coefficients between class accuracy and semantic scales S with different α. N denotes sample count, S represents semantic scale without inter-class interference, and En denotes effective sample count.

Dataset Model α=1.5 α=2.5 CIFAR-10-LT CIFAR-10 ResNet-18 ResNet-34 ResNet-18 ResNet-34 The previous subsection shows that the sum of semantic scales for all classes in a dataset is highly correlated with model performance. We further investigate the relationship between semantic scale and model bias for different classes. When a class is closer to other classes, model performance on that class worsens [**?**]. Therefore, we additionally consider inter-class interference when quantifying the degree of imbalance between semantic scales of classes. When class i is closer to other classes, a smaller weight wi is applied to class i' s semantic scale.

Specifically, assume the semantic scales of m classes after maximum normalization are S = [S 1, S 2, . . . , S m]T, and the centers of all classes are O = [o1, o2, . . . , om]T. Define the distance between class i and class j centers as di,j = oi − oj 2. The weights of m classes are written as W = [w1, w2, . . . , wm]T. After maximum normalization and logarithmic transformation of W , we obtain W =log (α + W) , $\alpha \geq 1$, where α controls the smoothing degree of W. After considering inter-class distance, the semantic scale S = S W, and S ' s role in dominating imbalance degree increases with larger α. The weight wi = 1 / Σj=1 oi − oj 2.

To obtain the most appropriate α, we calculate Pearson correlation coefficients between semantic scale and accuracy of ResNet-18 and ResNet-34 trained on CIFAR-10-LT and CIFAR-10, as shown in Table 1. Experimental settings are in Appendix D.2. We find that S is more dominant on long-tailed data than on non-long-tailed data, and the improved S is better than S and far better than sample count in reflecting model bias. In subsequent experiments, we set α to 2 for long-tailed data and 1 for non-long-tailed data.

3.4.2 Semantic Scale Imbalance on Long-Tailed Data

Previous studies have roughly attributed model bias to sample count imbalance. Experimental results in the first row of Figure 3 [Figure 3: see original paper] show that even though MNIST-LT-1 has similar sample counts to MNIST-LT-2, class-wise accuracy on MNIST-LT-1 is closer to that on MNIST, just as their semantic scales are also more similar.

Additionally, Figure 3 indicates that models on certain classes with fewer samples outperform those on classes with more samples, and that semantic scale S reflects model bias more accurately. Furthermore, we observe that CIFAR-100-LT accuracy does not show a significant decreasing trend, which can be

explained by the marginal effect of semantic scale (Sec 3.3).

3.4.3 Semantic Scale Imbalance on Non-Long-Tailed Data

Figure 4 [Figure 4: see original paper] demonstrates model bias exists not only on long-tailed data but also on sample-balanced data. Typically, classes with smaller semantic scales have lower accuracies. Based on semantic scale size, weaker and dominant classes can be well differentiated.

It should be noted that weaker classes are not random; experiments in Figure 4 show models consistently perform worse on the same classes. More semantic scale imbalance results for datasets are shown in Appendix D.4. In summary, semantic scale imbalance can represent model bias more generally and appropriately. We therefore expect to improve overall model performance when facing semantic scale imbalance problems and propose dynamic semantic-scale-balanced learning inspired by re-weighting strategies.

4 Dynamic Semantic-Scale-Balanced Learning

Deep neural networks can be viewed as a combination of feature mapping function and classifier, and several studies have shown that model bias is mainly caused by classifier bias [**?**]. We therefore focus more on semantic scale imbalance in feature space, i.e., semantic scale measured by feature volume. This section proposes a general semantic-scale-based loss improvement scheme and designs a training framework for successful application.

4.1 Dynamic Semantic-Scale-Balanced Loss

During training, feature vectors corresponding to samples vary with model parameters, so semantic scale per class constantly changes. Compared with traditional re-weighting strategies, we propose calculating the degree of imbalance between semantic scales in real-time at each iteration to dynamically evaluate weaker classes and assign greater weights to their corresponding losses. Specifically, for class i at each iteration, normalized re-weighting terms αi $1/\alpha$i are introduced, inversely proportional to semantic scales that consider inter-class interference, where C is the total number of classes. Given embedding z of a sample and label yi, the dynamic semantic-scale-balanced (DSB) loss can be expressed as DSB(z, yi) = 1/L(z, yi), i = 1, 2, . . . , C, where yi is the sample label from class i.

How to combine general loss to generate DSB loss is described in Appendix F.1. Our approach has great potential to improve re-balancing loss methods and sampling rate adjustment based on sample count, because both semantic scale and sample count are natural measures independent of the model.

However, the number of samples per iteration is limited, making it impossible to obtain features of all samples for calculating semantic scales. Therefore, we

propose a dynamic re-weighting training framework enabling successful DSB loss application.

4.2 Dynamic Re-Weighting Training Framework

Inspired by the slow drift phenomenon of features [**?**], we design a storage pool Q to store and update historical features and propose a three-stage training framework. A mini-batch of features can be dynamically updated at each iteration, and each class' s semantic scale is calculated using all features in the storage pool. The three-stage training framework is shown in Figure 16 [Figure 16: see original paper] and Algorithm 2 (more details in Appendix F.2), with the following description: (1) In the first stage, all features and labels generated by the 1st epoch are stored in Q, but they cannot be directly used to calculate semantic scales due to large drift between historical and current features in early stages. (2) The second stage corresponds to epochs 2 through n. At each iteration, the oldest mini-batch features and labels in Q are removed and those generated by the current iteration are stored. The goal is to continuously update features in Q until feature drift becomes small enough. We set n to 5 in our experiments, and the original loss function is used in the first two stages. Figure 5 [Figure 5: see original paper] shows the effect of n on model performance. A larger n does not hurt model performance but only takes slightly more time. Experience suggests setting n to 5 is sufficient. (3) The third stage corresponds to epochs $>$ n. At each iteration, semantic scales are calculated using features in Q after updating Q, and the original loss is re-weighted.

Figure 5: Performance of different models with different losses on different datasets for different n values.

Comparison and analysis of video memory and training speed are in Appendix F.2. We answer possible questions about the methods section in detail in Appendix B.

5 Experiments

To validate the superiority and generality of proposed dynamic semantic-scale-balanced learning, we design four experiments. The first experiment on large-scale long-tailed datasets ImageNet-LT and iNaturalist2018 [**?**] confirms our approach' s superior performance on long-tailed data. The second experiment uses large-scale ImageNet [**?**] and sample-balanced CIFAR-100, the third selects CIFAR-100-LT and benchmark datasets commonly used in deep metric learning (CUB-2011 [**?**] and Cars196 [**?**]), and the fourth performs on MSCOCO-GLT [**?**] to demonstrate effectiveness in generalized long-tailed classification. Comprehensive experiments demonstrate our method' s generality and superiority. More results are in Appendix D.5, Appendix D.6 (fundus dataset OIA-ODIR [**?**]) and Appendix D.7 (remote sensing image scene classification). Ablation experiments and additional analyses are in Appendix H.

5.1 Results on ImageNet-LT and iNaturalist2018

ImageNet-LT is a long-tailed version of ImageNet containing 1,000 classes with 1,280 to 5 samples per class. iNaturalist2018 is a real-world, extremely unbalanced dataset containing 437,513 images from 8,142 classes. We adopt official training and validation splits [**?**].

Table 2 : Top-1 Acc(%) on ImageNet-LT and iNaturalist2018. We use ResNext-50 [**?**] on ImageNet-LT and ResNet-50 [**?**] on iNaturalist2018 as network backbones for all methods. Model training uses SGD optimizer with batch size 256 (ImageNet-LT) / 512 (iNaturalist), momentum 0.9, weight decay 0.0005, and learning rate 0.1 (linear LR decay).

Methods BBN [**?**] DIVE [**?**] CB-CE [**?**] DSB-CE DSB-CE+IFL [**?**] Focal [**?**] CB-Focal [**?**] DSB-Focal LDAM [**?**] DSB-LDAM BS [**?**] DSB-BS LADE [**?**] DSB-LADE PaCo [**?**] DSB-PaCo MBJ [**?**] DSB+MBJ RIDE [**?**] MBJ+RIDE [**?**] DSB+RIDE ImageNet-LT(ResNeXt50) iNaturalist 2018(ResNet50) Head Middle Overall 21.4(+13.7) 49.2(+4.8) 22.5(+14.8) 50.1(+5.7) 23.7(+10.6) 50.6(+3.4) 33.4(+1.5) 35.4(+3.3) 33.6(+2.4) 41.5(+2.3) 40.7(+1.7) 38.5(+2.5) 52.3(+1.2) 52.8(+1.6) 53.2(+1.3) 55.9(+1.5) 53.3(+1.2) 58.2(+2.1) Head Middle Overall 62.7(+6.5) 64.3(+2.6) 63.4(+7.2) 65.0(+3.3) 63.5(+2.4) 65.7(+1.1) 49.4(+2.7) 55.1(+1.9) 70.5(+0.8) 73.4(+1.1) 70.9(+0.9) 74.2(+1.1) 73.4(+0.8)

Table 2 shows that when CE, Focal [**?**] and RIDE are combined with our approach (Appendix D.1), overall model performance significantly improves. For example, DSB-CE overall accuracy is 4.8% and 2.6% higher than CE on ImageNet-LT and iNaturalist2018, respectively. We also report performance on three subsets (Head: >100 images, Middle: 20-100 images, Tail: <20 images). Our method achieves the largest improvement for the tail subset without compromising head subset performance, where DSB-CE and DSB-Focal improve 13.7% and 10.6%, respectively, over original methods on ImageNet-LT' s tail subset, effectively alleviating model bias. Additionally, IFL [**?**] considers intra-class long-tailed problems, and combining DSB-CE with it (i.e., DSB-CE-IFL) further enhances model performance. We therefore encourage researchers to focus on intra-class long-tailed problems.

5.2 Results on ImageNet and CIFAR-100

Table 3 : Comparison on ImageNet and CIFAR-100. On ImageNet, we use random clipping, mixup [**?**], and cutmix [**?**] for data augmentation, and all models are optimized by Adam with batch size 512, learning rate 0.05, momentum 0.9, and weight decay 0.0005. On CIFAR-100, batch size is 64 with random clipping, mixup, and cutmix augmentation. Adam optimizer with learning rate 0.1 (linear decay), momentum 0.9, and weight decay 0.005 trains all networks.

Methods VGG16 [**?**] BN-Inception [**?**] ResNet-18 ResNet-34 ResNet-50 DenseNet-201 [**?**] SE-ResNet-50 [**?**] ResNeXt-101 [**?**] ImageNet Top-1 Acc(%) DSB-CE CIFAR-100 Top-1 Acc(%) DSB-CE We use the ILSVRC2012 split

containing 1,281,167 training and 50,000 validation images. Each CIFAR-100 class contains 500 training and 100 test images. Table 3 results indicate our approach achieves >1% performance gains for various networks on both datasets. Notably, it enables VGG16 to improve 1.3% and 1.5% on ImageNet and CIFAR-100, respectively, compared to the original method. This implies semantic scale imbalance exists in non-long-tailed datasets and affects model performance.

5.3 Results on CUB-2011, Cars196 and CIFAR-100-LT

Since we also improve classical losses (NormSoftmax and SoftTriple [**?**]) in deep metric learning, we follow the widely adopted backbone network, experimental parameters, and dataset splits in this field (Appendix D.5). The two improved loss functions are denoted DSB-NSM and DSB-ST, with formulas given in Appendix F.1.

Table 4 : Results on CUB-2011 and Cars196. We evaluate model performance with Recall@K [**?**] and Normalized Mutual Information (NMI) [**?**].

Dataset Metric NormSoftmax DSB-NSM SoftTriple DSB-ST Circle NormSoftmax DSB-NSM SoftTriple DSB-ST 59.2(+1.4) 61.3 (+1.2) 65.1(+1.2) 66.4(+1.0) CUB-2011 70.7(+0.7) 72.7(+0.8) 76.3(+0.8) 77.0(+0.6) 66.5 (+1.2) 67.3(+1.1) 69.2(+0.9) 70.6(+1.3) 77.9(+1.1) 79.8(+1.2) 84.0(+0.8) 85.6(+1.1) Cars196 86.4(+0.8) 87.5(+0.9) 90.2(+0.7) 91.3(+0.6) 67.8(+1.1) 68.3 (+1.3) 70.9(+1.2) 71.1(+1.0)

Table 4 summarizes our method' s performance with 64 and 512 embeddings. Experiments show DSB loss consistently improves by >1% on R1 and NMI. DSB-ST with 512 embeddings performs superiorly on Cars196, where R@1 and R@2 exceed Circle loss [**?**] by 2.2% and 1.5%, respectively.

Table 5 : Results on CIFAR-100-LT. Dataset imbalance factor is defined as the ratio of training samples in the largest class to that in the smallest class.

Dataset Imbalance factor Metric NormSoftmax CB-NSM DSB-NSM SoftTriple CB-ST DSB-ST CIFAR-100-LT Results on CIFAR-100-LT. Class-balanced loss (CB loss) that performs well on long-tailed data and is also based on re-weighting strategy is selected for comparison with DSB loss. Analyzing Table 5 results, DSB loss outperforms CB loss overall. When long-tailed CIFAR-100 has imbalance factor 200, DSB-ST performs significantly better than CB-ST, exceeding SoftTriple on R@1, R@2 and NMI by 2.7%, 2.8% and 1.9%, respectively.

5.4 The Performance of Dynamic Semantic-Scale-Balanced Learning in Generalized Long-Tailed Learning

Invariant feature learning (IFL [**?**]) considers both inter-class and intra-class long tails, further defining generalized long-tailed classification. Intra-class long tail has not been previously considered, and invariant feature learning addresses

it for the first time, representing remarkable progress in solving long-tailed problems. IFL decomposes the classification problem's probabilistic model as P(y | x) = P(x|y)P(x)P(y) and defaults few-sample classes as weak classes. Notably, our study found that geometric properties of manifolds corresponding to different class distributions P(x) affect classification difficulty, breaking previous perceptions and indicating that inter-class long-tail problems still have huge research potential. Data manifold existence is already consensus, and data classification can be regarded as manifold unwinding and separation. Typically, a deep neural network consists of a feature extractor and classifier. Feature learning can be considered manifold unwinding, and a well-learned feature extractor can unwind multiple manifolds for the classifier to decode. In this view, all factors about manifold complexity may affect model classification performance. We therefore suggest future work explore inter-class long-tailed problems from a geometric perspective. Additionally, considering both inter-class and intra-class long tails will greatly alleviate long-tailed problems.

Table 6 : Evaluation on MSCOCO-GLT. Protocols < Accuracy | Precision > cRT [**?**] LWS [**?**] Deconfound-TDE [**?**] BLSoftmax [**?**] BBN [**?**] LDAM [**?**] DSB-LDAM BLSoftmax + IFL [**?**] DSB-BLSoftmax DSB-BLSoftmax + IFL cRT + IFL [**?**] DSB-cRT LWS + IFL [**?**] DSB-LWS Overall 73.64 | 75.84 72.60 | 75.66 73.79 | 74.90 72.64 | 75.25 73.69 | 77.35 75.57 | 77.70 76.63 | 78.95 73.72 | 77.08 73.96 | 77.37 74.64 | 78.06 76.21 | 79.11 76.82 | 79.95 75.98 | 79.18 76.55 | 80.06 Overall 64.69 | 68.33 63.60 | 68.81 66.07 | 68.20 64.07 | 68.59 64.48 | 70.20 67.26 | 70.70 68.15 | 71.87 64.76 | 70.00 65.03 | 70.15 65.47 | 70.83 66.90 | 71.34 67.26 | 71.73 66.55 | 71.49 67.03 | 72.15 Overall 49.97 | 50.37 50.14 | 50.61 50.76 | 51.68 49.72 | 50.65 51.83 | 51.77 55.52 | 56.21 56.16 | 56.87 52.97 | 53.52 50.24 | 51.36 53.08 | 53.75 52.07 | 52.85 51.41 | 51.94 52.07 | 52.90 51.64 | 51.16

Invariant feature learning estimates relatively unbiased feature centers by constructing resampling strategies and uses center loss for unbiased feature learning. We applied IFL to dynamic semantic-scale-balanced learning to consider both inter-class and intra-class long tails, validating it on ImageNet-LT and iNaturalist2018. Experiments show DSB-CE combined with IFL achieves further performance improvement, with results and analysis supplemented in Table 2.

We note that IFL proposes two datasets (ImageNet-GLT and MSCOCO-GLT) and three testing protocols. Since our paper already contains many experiments, we selected MSCOCO-GLT with the same experimental settings as IFL. Results are shown in Table 6. On CLT and GLT protocols, we significantly improve BL-softmax, LDAM, and BL-softmax+IFL performance. Our approach also promotes the above three methods' performance on the ALT protocol, possibly due to additional gain from stronger inter-class discriminability. Due to page limitations, this experiment is temporarily supplemented in the appendix; we will include it in the main text if the paper is accepted. Experiments show alleviating both inter-class and intra-class long tails can significantly improve model performance, so we encourage researchers to focus on intra-class long-tailed problems.

5.5 Experiment Summary

Extensive experiments confirm that dynamic semantic-scale-balanced learning achieves superior performance not only on long-tailed datasets but also on non-long-tailed and even sample-balanced datasets. This indicates that semantic scale imbalance warrants extensive attention.

6 Discussion

In this work, we pioneer the concept and quantitative measurement of semantic scale imbalance and make two important discoveries: (1) semantic scale has marginal effects, and (2) semantic scale imbalance can accurately describe model bias. Notably, our proposed semantic scale, like sample count, is a natural measure of class imbalance that does not depend on model predictions (see Related Work in Appendix A). Semantic scale can guide data augmentation—for example, semantic scale imbalance can evaluate which classes are weaker and need augmentation, and marginal effects can assist in selecting appropriate sample counts. We expect our work will bring more attention to this more prevalent model bias, improve model robustness, and promote fairer AI development.

References

[1] Elie Aljalbout, Vladimir Golkov, Yawar Siddiqui, Maximilian Strobel, and Daniel Cremers. Clustering with deep learning: Taxonomy and new methods. arXiv preprint arXiv:1801.07648, 2018.

[2] Dong Cao, Xiangyu Zhu, Xingyu Huang, Jianzhu Guo, and Zhen Lei. Domain balancing: Face recognition on long-tailed domains. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5671–5679, 2020.

[3] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. Advances in neural information processing systems, 32, 2019.

[4] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. Advances in neural information processing systems, 32, 2019.

[5] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16:321–357, 2002.

[6] Hsin-Ping Chou, Shih-Chieh Chang, Jia-Yu Pan, Wei Wei, and Da-Cheng Juan. Remix: Rebalanced mixup. In European Conference on Computer Vision, pages 95–110. Springer, 2020.

[7] Peng Chu, Xiao Bian, Shaopeng Liu, and Haibin Ling. Feature space augmentation for long-tailed data. In Computer Vision–ECCV 2020: 16th European

Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16, pages 694–710. Springer, 2020.

[8] Thomas M Cover. Elements of information theory. John Wiley & Sons, 1999.

[9] Ekin Dogus Cubuk, Ethan S Dyer, Rapha Gontijo Lopes, and Sylvia Smullin. Tradeoffs in data augmentation: An empirical study. 2021.

[10] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 715–724, 2021.

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

[12] Charles Elkan. The foundations of cost-sensitive learning. In International joint conference on artificial intelligence, volume 17, pages 973–978. Lawrence Erlbaum Associates Ltd, 2001.

[13] Andrew Estabrooks, Taeho Jo, and Nathalie Japkowicz. A multiple resampling method for learning from imbalanced data sets. Computational intelligence, 20(1):18–36, 2004.

[14] Chengjian Feng, Yujie Zhong, and Weilin Huang. Exploring classification equilibrium in long-tailed object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 3417–3426, 2021.

[15] Yuanfan Guo, Minghao Xu, Jiawen Li, Bingbing Ni, Xuanyu Zhu, Zhenbang Sun, and Yi Xu. Hcsc: Hierarchical contrastive selective coding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 9706–9715, 2022.

[16] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In International conference on intelligent computing, pages 878–887. Springer, 2005.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

[18] Yin-Yin He, Jianxin Wu, and Xiu-Shen Wei. Distilling virtual examples for long-tailed recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 235–244, 2021.

[19] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 6626–6636, 2021.

[20] Ting-I Hsieh, Esther Robb, Hwann-Tzong Chen, and Jia-Bin Huang. Droploss for long-tail instance segmentation. In AAAI, volume 3, page 15, 2021.

[21] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7132–7141, 2018.

[22] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4700–4708, 2017.

[23] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning, pages 448–456. PMLR, 2015.

[24] Muhammad Abdullah Jamal, Matthew Brown, Ming-Hsuan Yang, Liqiang Wang, and Boqing Gong. Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 7610–7619, 2020.

[25] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. Intelligent data analysis, 6(5):429–449, 2002.

[26] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. arXiv preprint arXiv:1910.09217, 2019.

[27] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. arXiv preprint arXiv:1910.09217, 2019.

[28] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

[29] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In Proceedings of the IEEE international conference on computer vision workshops, pages 554–561, 2013.

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

[31] Ning Li, Tao Li, Chunyu Hu, Kai Wang, and Hong Kang. A benchmark of ocular disease intelligent recognition: One shot for multi-disease detection. In International Symposium on Benchmarking, Measuring and Optimization, pages 177–193. Springer, 2020.

[32] Xiaotong Li, Yongxing Dai, Yixiao Ge, Jun Liu, Ying Shan, and Ling-Yu Duan. Uncertainty modeling for out-of-distribution generalization. arXiv preprint arXiv:2202.03958, 2022.

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

[34] Jialun Liu, Yifan Sun, Chuchu Han, Zhaopeng Dou, and Wenhui Li. Deep representation learning on long-tailed data: A learnable embedding augmentation perspective. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2970–2979, 2020.

[35] Jialun Liu, Jingwei Zhang, Wenhui Li, Chi Zhang, Yifan Sun, et al. Memory-based jitter: Improving visual recognition on long-tailed data with diversity in memory. arXiv preprint arXiv:2008.09809, 2020.

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

[37] Yi Ma, Harm Derksen, Wei Hong, and John Wright. Segmentation of multivariate mixed data via lossy data coding and compression. IEEE transactions on pattern analysis and machine intelligence, 29(9):1546–1562, 2007.

[38] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens Van Der Maaten. Exploring the limits of weakly supervised pretraining. In Proceedings of the European conference on computer vision (ECCV), pages 181–196, 2018.

[39] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 26, 2013.

[40] Erxue Min, Xifeng Guo, Qiang Liu, Gen Zhang, Jianjing Cui, and Jun Long. A survey of clustering with deep learning: From the perspective of network architecture. IEEE Access, 6:39501–39514, 2018.

[41] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4004–4012, 2016.

[42] Qi Qian, Lei Shang, Baigui Sun, Juhua Hu, Hao Li, and Rong Jin. Softtriple loss: Deep metric learning without triplet sampling. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 6450–6458, 2019.

[43] Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. Advances in neural information processing systems, 33:4175–4186, 2020.

[44] Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. Advances in Neural Information

Processing Systems, 33:4175–4186, 2020.

[45] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. International journal of computer vision, 115(3):211–252, 2015.

[46] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. Introduction to information retrieval, volume 39. Cambridge University Press Cambridge, 2008.

[47] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 761–769, 2016.

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[49] Adarsh Subbaswamy, Roy Adams, and Suchi Saria. Evaluating model robustness and stability to dataset shift. In International Conference on Artificial Intelligence and Statistics, pages 2611–2619. PMLR, 2021.

[50] Yanmin Sun, Mohamed S Kamel, Andrew KC Wong, and Yang Wang. Cost-sensitive boosting for classification of imbalanced data. Pattern recognition, 40(12):3358–3378, 2007.

[51] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 6398–6407, 2020.

[52] Yifan Sun, Yuke Zhu, Yuhan Zhang, Pengkun Zheng, Xi Qiu, Chi Zhang, and Yichen Wei. Dynamic metric learning: Towards a scalable metric space to accommodate multiple semantic scales. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5393–5402, 2021.

[53] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1–9, 2015.

[54] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 11662–11671, 2020.

[55] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. Advances in Neural Information Processing Systems, 33:1513–1524, 2020.

[56] Kaihua Tang, Mingyuan Tao, Jiaxin Qi, Zhenguang Liu, and Hanwang Zhang. Invariant feature learning for generalized long-tailed classification. In European Conference on Computer Vision, pages 709–726. Springer, 2022.

[57] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. science, 290(5500):2319–2323, 2000.

[58] Junjiao Tian, Niluthpol Chowdhury Mithun, Zachary Seymour, Han-Pang Chiu, and Zsolt Kira. Striking the right balance: Recall loss for semantic segmentation. In 2022 International Conference on Robotics and Automation (ICRA), pages 5063–5069. IEEE, 2022.

[59] Ivan Tomek et al. Two modifications of cnn. 1976.

[60] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 8769–8778, 2018.

[61] Grant Van Horn and Pietro Perona. The devil is in the tails: Fine-grained classification in the wild. arXiv preprint arXiv:1709.01450, 2017.

[62] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. Advances in neural information processing systems, 29:3630–3638, 2016.

[63] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[64] Tong Wang, Yousong Zhu, Chaoyang Zhao, Wei Zeng, Jinqiao Wang, and Ming Tang. Adaptive class suppression loss for long-tail object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3103–3112, 2021.

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

[66] Xun Wang, Haozhi Zhang, Weilin Huang, and Matthew R Scott. Cross-batch memory for embedding learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 6388–6397, 2020.

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

[68] Tong Wu, Qingqiu Huang, Ziwei Liu, Yu Wang, and Dahua Lin. Distribution-balanced loss for multi-label classification in long-tailed datasets. In European Conference on Computer Vision, pages 162–178. Springer, 2020.

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

[70] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1492–1500, 2017.

[71] Honggang Yang, Jiejie Chen, Rong Luan, Mengfei Xu, Lin Ma, and Xiaoqi Zhou. Base on megapixel color fundus photos for multi-label disease classification. In 2022 14th International Conference on Advanced Computational Intelligence (ICACI), pages 29–35. IEEE, 2022.

[72] Shuo Yang, Lu Liu, and Min Xu. Free lunch for few-shot learning: Distribution calibration. arXiv preprint arXiv:2101.06395, 2021.

[73] Yuzhe Yang, Hao Wang, and Dina Katabi. On multi-domain long-tailed recognition, imbalanced domain generalization and beyond. In European Conference on Computer Vision, pages 57–75. Springer, 2022.

[74] Yuzhe Yang and Zhi Xu. Rethinking the value of labels for improving class-imbalanced learning. arXiv preprint arXiv:2006.07529, 2020.

[75] Xi Yin, Xiang Yu, Kihyuk Sohn, Xiaoming Liu, and Manmohan Chandraker. Feature transfer learning for face recognition with under-represented data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5704–5713, 2019.

[76] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In Proceedings of the IEEE/CVF international conference on computer vision, pages 6023–6032, 2019.

[77] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412, 2017.

[78] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. arXiv preprint arXiv:2110.04596, 2021.

[79] Zizhao Zhang and Tomas Pfister. Learning fast sample re-weighting without reward data. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 725–734, 2021.

[80] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. 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, pages 9719–9728, 2020.

[81] Zhi-Hua Zhou and Xu-Ying Liu. Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Transactions on

knowledge and data engineering, 18(1):63–77, 2005.

A Related Work

Real-world datasets tend to be long-tailed. Extreme sample count imbalance in long-tailed data prevents classification models from adequately learning tail class distributions, leading to poor tail class performance. Therefore, re-balancing sample count methods [**?**] and balancing per-class loss methods [**?**]—i.e., re-sampling and cost-sensitive learning—are proposed. Cost-sensitive learning is most relevant to our work. [**?**] proposes using label frequency to adjust loss during training to mitigate class bias. [**?**] assigns loss weights to each class, giving higher weights to hard samples. Recent studies show that strictly re-weighting loss by inverse sample count is moderate [**?**]. Some methods generating weights more "smoothly" perform better, such as taking the square root of sample count [**?**] as weights. CB loss [**?**] attributes better "smooth" approach performance to marginal effects, while other studies [**?**] attribute it to negative gradient over-suppression. Distribution-balanced loss [**?**] proposes negative-tolerant regularization to mitigate gradient differences and recalculates loss by computing the ratio of expected to actual sampling frequency for each class. Seesaw Loss [**?**] leverages mitigation and compensation factors to dynamically suppress excessive negative sample gradients on tail classes while complementing penalties for misclassified samples. Furthermore, with decoupled training proposal [**?**], [**?**] adopts decoupled training using cross-entropy loss for feature learning in the first stage and re-balancing loss for classifier learning in the second stage.

Unlike the above re-balancing loss studies based on sample count or positive-negative gradient ratios, our work proposes a novel measure called semantic scale. Compared to sample count, semantic scale also considers sample distribution scope. In contrast to gradient-based measures, semantic scale does not depend on model output and gradient back-propagation, making it a natural measure similar to sample count. Model robustness work has focused on out-of-domain generalization performance, known as "out-of-distribution generalization." For example, [**?**] aims to maintain good model performance when test distribution deviates from training distribution. Similarly, [**?**] aims to let models learn more out-of-domain information. Unlike them, we address model bias introduced by unbalanced data that causes poor performance on certain classes.

B Explanation of a Few Key Points

B.1 How does the section "Slow drift phenomenon and marginal effects of characteristics" relate to the rest of the paper?

(1) Why introduce marginal effects? The effective number of samples discusses the relationship between sample count and feature diversity within a class, arguing that feature diversity has marginal effects. However, effective number of samples has major drawbacks (introduced in Section 2): it fails on sample-balanced datasets and does not provide quantitative

feature diversity measurement. Therefore, we extend the effective number of samples mechanism and propose "semantic scale" that can effectively measure feature diversity on sample-balanced datasets. Our approach is more general than effective number of samples. Additionally, marginal effects prove our extension is reasonable and appropriate.

Our approach simultaneously explains three phenomena that other methods cannot, indicating our proposed method' s reliability. The paper' s logic is: • CB loss introduced effective number of samples concept based on marginal effects. • We extend effective number of samples and propose semantic scale concept. • Experiments show semantic scale still has marginal effects (Section 3.3). • Based on step 3's properties, we can explore many applications requiring marginal effects as theoretical support (e.g., representative sample selection in Appendix I).

In summary, we explain that semantic scale measurement (or semantic scale imbalance based on semantic scale) was inspired by effective number of samples with marginal effects. Discarding marginal effects would weaken our early motivation and later practical applications theoretically. (2) Association with feature slow drift: Since experiments show very high correlation between semantic scale and model bias, we propose re-weighting the loss function by inverse semantic scale. Features change dynamically during training, and all feature vectors are needed to calculate each class' s semantic scale. Obviously, one batch' s data is insufficient, so we propose dynamically updating and storing historical features to calculate semantic scale, and the feature slow drift phenomenon ensures this operation' s feasibility.

Section 2 is indispensable for the whole paper, ensuring coherence.

B.2 Why focus on the relationship between semantic scale and accuracy?

Note that we address model bias introduced by unbalanced data, causing poor model performance on some classes. Previously, researchers believed models performed poorly on few-sample classes, defining them as tail classes and many-sample classes as head classes, proposing long-tailed identification tasks. However, we observe that models do not necessarily perform poorly on few-sample classes, explaining why some tail classes are "overbalanced" in many long-tailed identification methods. The higher similarity between our proposed semantic scale and model performance allows us to redefine the imbalance problem by replacing sample count with semantic scale. Superior performance on sample-balanced datasets shows our proposed semantic scale imbalance is reliable. Semantic scale measurement does not depend on the model and is calculated directly from data. Even more surprising, class semantic scale can predict class performance, leading to further understanding of what models learn from data and facilitating data-driven artificial intelligence development.

B.3 Why should the loss function be dynamically weighted?

DSB loss works as follows: in each iteration, each class' s semantic scale is calculated in feature space, and the loss function is re-weighted by inverse semantic scale.

Why "dynamically"weight the loss? Because semantic scales change continuously as features change during training, requiring us to update features each iteration and re-calculate semantic scales to re-weight the loss function. "Dynamic" refers to semantic scale updates each iteration.

However, dynamic weighting implementation is difficult because calculating semantic scale requires all features, and extracting features of all samples each iteration would be time-consuming. Therefore, we analyze the "feature slow drift" phenomenon in Section 2 and propose calculating semantic scale by dynamically storing and updating historical features (i.e., dynamic re-weighting training framework). Comparative experiments on training speed and memory consumption are presented in Appendix F.2, showing our approach is efficient.

B.4 What is the point of proposing various cost-sensitive learning methods? Why not directly use each class' s accuracy to weight the loss?

What is the point of proposing various cost-sensitive learning methods? For example, using inverse sample count or effective number of samples to reweight loss rather than directly using each class' s accuracy. After careful consideration, we believe there are several reasons:

(1) Reweighting loss with class accuracy may cause models to over-focus on weak classes, ignoring other classes. Recent studies show that strictly reweighting loss by inverse sample count has modest effect [**?**]. Some "smoother" methods perform better, such as taking the square root of sample count [**?**] as weights. [**?**] argues that "smoother" methods perform better due to marginal effects. Our approach can be understood as a smoothed version of class accuracy because our proposed semantic scale has marginal effects and high correlation with class accuracy. We note a recent work (CDB loss) published in IJCV that measures class difficulty. Additionally, domain balancing also measures class-level difficulty, so we compare semantic-scale-balanced learning with them. Introduction and comparison experiments for these two methods are in Appendix H.2.

(2) Weighting by model performance cannot bring new cognition. Why do models perform poorly on some data and well on others? For example, face recognition models usually do not perform well in dark environments. When encountering such problems, we first consider whether lack of dark environment data causes inadequate model learning. Since much data is available, this problem is not caused by few samples, so is there another explanation? We argue that face patterns in dark environments

are not rich enough, causing many samples to cluster around manifolds with smaller volumes, making faces difficult to distinguish. Our approach not only addresses performance imbalance but also advances researchers' understanding of deep neural networks. Scientific advances are usually accompanied by new cognition establishment.

(3) Our proposed semantic scale has great application potential. In engineering applications, how many samples should be collected for each class? Too few samples cause under-representation, while too many consume huge costs. Our approach can effectively solve this by stopping collection when semantic scales tend to saturate. When communicating with technology companies, we find they have 100 million data points but no proper way to select representative data. We therefore design an idea to select representative data using semantic scales, detailed in Appendix I.

C Experiments on Stanford Point Cloud Manifolds

Since (Z − Zmean)(Z − Zmean)T is a real symmetric matrix, it is semi-positive definite. Further, I + p Zmean)(Z − Zmean)T is a positive definite matrix and therefore det(I + p (Z − Zmean)(Z − Zmean)T) > 0.

Semantic scale measurement is derived from singular value decomposition of the data matrix, jointly determined by most samples. Therefore, our method is insensitive to noisy samples, i.e., semantic scale measurement is numerically stable. We calculate and plot semantic scales of multiple Stanford point cloud manifolds with different sizes in Figure 6 [Figure 6: see original paper]. Let bunny' s center point be Cbunny. We increase bunny' s volume by performing w ∗ (bunny − Cbunny), scaling other point clouds similarly. As object manifold scales up, calculated volume increases slowly and monotonically, indicating our method can accurately measure relative manifold volume size and is numerically stable, an advantage that helps mitigate noisy sample effects.

D Experimental Details

D.1 Marginal Effect of Semantic Scale

We generate new training datasets for classification experiments using the following method: assume the original dataset has C total classes, and randomly select m samples from each class to form a sub-dataset with total sample count C × m. Details of sub-datasets generated based on CIFAR-10, CIFAR-100 and Mini-ImageNet are in Table 7.

D.2 Quantification of Semantic Scale Imbalance

We train ResNet-18 and ResNet-34 on CIFAR-10-LT and CIFAR-10 with imbalance factor 200, respectively, with CIFAR-10-LT' s test set consistent with CIFAR-10. During training, batch size is fixed to 64 and optimizer adopts Adam. Learning rate starts at 0.01 and becomes 0.98×the previous learning rate after

each epoch. We do not employ other additional tricks or data augmentation strategies.

D.3 Semantic Scale Imbalance on Long-Tailed Data

We artificially produce two long-tailed versions of the MNIST dataset, called MNIST-LT-1 and MNIST-LT-2. Sample counts per class are listed in Table 8 .

Figure 3 shows class-wise accuracies of ResNet-18 and ResNet-34 trained on CIFAR-10-LT and CIFAR-100-LT with the same training settings as Appendix D.2. Taking CIFAR-10 as an example, labels 1 to 10 correspond to: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck. Classification experiment prediction scores on CIFAR-10-LT and CIFAR-10 find that cat (label 4) is most easily confused with dog (label 6), as shown by their lowest accuracy on CIFAR-10 (Figure 4). However, cat and dog accuracies on CIFAR-10-LT are higher than deer (label 5), due to S ' s dominant role in S for long-tailed data.

Table 8: The two long-tailed MNIST datasets resampled from MNIST.

Dataset MNIST-LT-1 Class label Number 5,923 3,590 2,940 2,518 2,256 1,972 1,700 1,300 1,100 Dataset MNIST-LT-2 Class label Number 5,923 3,090 2,540 1,818 1,356

D.4 Semantic Scale Imbalance for More Datasets

We have demonstrated semantic scale imbalance on MNIST, MNIST-LT, CIFAR-10, CIFAR-10-LT, CIFAR-100 and CIFAR-100-LT in Section 3.4. Figure 7 [Figure 7: see original paper] additionally shows semantic scale imbalance degree on CUB-2011, Cars196, and Mini-ImageNet, indicating that semantic scale imbalance is indeed prevalent across all kinds of datasets.

D.5 Experimental Settings for Section 5.3 and More Experiments

D.5.1 Experimental Settings for Section 5.3 Backbone Network and Experimental Parameters. Since we improve classical loss in deep metric learning, we follow the widely adopted backbone network, experimental parameters, and dataset splits in this field. BN-Inception [**?**] pre-trained on ImageNet is adopted as the backbone network, with training set augmented by random horizontal flipping and random cropping. All images are cropped to 224$\times$224 as network input. After global average pooling, network output is fed into a single fully connected layer to obtain 64- or 512-dimensional feature embeddings, then all embeddings are clustered by K-means. The model is optimized by Adam with batch size 32 and 50 epochs. We evaluate learned embedding performance with Recall@K and Normalized Mutual Information (NMI). Remaining experimental parameters are consistent with those reported in NormSoftmax and SoftTriple [**?**].

Dataset Introduction (CUB-2011, Cars196 and CIFAR-100-LT). CUB-2011 dataset has 5,864 images in the first 100 classes for training and 5,924

images in the second 100 classes for testing. Cars196 dataset consists of 196 classes totaling 16,185 images, with the first 98 classes for training and remaining classes for testing. CIFAR-100 has 100 classes, each containing 600 images. We create three long-tailed CIFAR-100 versions with the first 60 classes for training (see Figure 8a [Figure 8: see original paper]) and test on remaining classes [**?**].

D.5.2 More Experiments Long-tailed Cars196 is created using the first 98 classes for training (see Figure 8b) and the test set is the remaining classes. Both Mini-ImageNet and CIFAR-100 datasets contain 100 classes with 600 samples each. For Mini-ImageNet, the first 64 classes are the training set and the last 36 classes are the test set; for CIFAR-100, the first 60 classes are the training set and remaining classes are the test set. Note that CIFAR-100 experiments in this section differ from classification experiments in Sec 5.3. This experiment' s purpose is to complement our proposed method' s effectiveness on both long-tailed and sample-balanced datasets in deep metric learning.

Table 9 : Comparison on long-tailed Cars196.

Dataset Long-tailed Cars196 Imbalance factor Metric NormSoftmax CB-NSM DSB-NSM SoftTriple CB-ST DSB-ST

Table 9 shows DSB-NSM and DSB-ST performance comparison on long-tailed Car196. When imbalance factor is 10, DSB-NSM outperforms NSM (NormSoftmax) by 3.1% on R@1 and DSB-ST outperforms ST (SoftTriple) by 2.1%. When imbalance factor is 50, DSB-NSM and DSB-ST improve 2.8% and 2.5%, respectively, on R@1 compared to original methods. Additionally, DSB loss performs better than CB loss on all metrics.

Table 10 shows that compared with original losses, DSB-NSM and DSB-ST consistently improve R@1 by 1.3% on average and NMI by 1-2% for both sample-balanced datasets, with all other metrics outperforming originals. Tables 8 and 9 results further confirm that our proposed dynamic semantic-scale-balanced learning applies to long-tailed and sample-balanced datasets in deep metric learning, with broad application prospects.

Table 10: Comparison on Mini-ImageNet and CIFAR-100.

Dataset Mini-ImageNet CIFAR-100 Metric NormSoftmax DSB-NSM 87.1(+1.4) 92.0(+0.8) 75.5(+1.4) 61.4(+1.3) 72.2(+0.7) 50.6(+1.2) SoftTriple DSB-ST 88.0(+1.1) 92.8(+0.8) 78.8(+1.5) 63.5(+1.4) 73.9(+0.6) 53.0(+1.0)

D.6 Results on the Fundus Datasets OIA-ODIR and OIA-ODIR-B

D.6.1 Dataset Introduction The OIA-ODIR dataset [**?**] was made public in 2019, containing 10,000 fundus images across 8 classes. As shown in Figure 9 [Figure 9: see original paper], the eight classes are: Normal(N), hypertensive retinopathy(D), glaucoma(G), cataract(C), age-related macular degeneration(A), hypertension complication (H), pathologic myopia (M), other disease/abnormality(O). Considering O usually appears with other diseases, to

reduce ambiguity we adopt the data splitting scheme of [**?**], using only the first 7 classes' data, with training and test sample counts per class shown in Figure 10 [Figure 10: see original paper].

The OIA-ODIR dataset suffers from unbalanced sample counts. To fully validate our method, we produced a balanced version, OIA-ODIR-B, using the class with fewest samples as benchmark. As shown in Figure 10, each OIA-ODIR-B class contains 103 training and 46 test samples.

In addition to sample counts, we plot semantic scale imbalance degree for OIA-ODIR and OIA-ODIR-BS training sets in Figure 10.

D.6.2 Backbone Network and Experimental Parameters We used ResNet-50 pre-trained on ImageNet as the backbone network. Adam optimizer with learning rate 0.1 (linear decay), momentum 0.9, and weight decay 0.005 trained all networks. Following [**?**], average precision (AP) was used as model performance metric.

D.6.3 Results on OIA-ODIR We improved advanced class rebalancing methods (BS [**?**], Focal loss [**?**], LDAM [**?**]) and plot classification results in Figure 11 [Figure 11: see original paper]. Experimental findings are summarized as follows:

Although class H has the fewest samples, all methods outperform on class H compared to classes C, M, and A. This again shows sample count is not the best measure of class imbalance. Sample count-based methods usually give larger boosts to smallest-sample classes and thus fail to pay more attention to classes C, M, and A. Our method gives the most significant boosts to these three classes, indicating semantic scale imbalance more accurately reflects class difficulty.

D.6.4 Results on OIA-ODIR-B Since class rebalancing methods based on sample count cannot be applied to sample-balanced datasets, we additionally adopted VGG-16, ResNet-18 and SE-ResNet-50 as backbone networks to test DSB-CE's enhancement effect on CE, with experimental results shown in Figure 12. Experimental findings are summarized as follows:

With balanced sample counts, models still perform poorly on classes C, M and A. Figure 10 shows these three classes' semantic scales are significantly smaller than other classes. Our approach yields significant performance gains for all models on classes C, M, and A, promoting more balanced model performance across all classes, which is important in medical AI.

Experiment Summary. We validated semantic-scale-balanced learning effectiveness on both balanced and long-tailed fundus image datasets. Results show semantic scale imbalance exists in medical image datasets and significantly limits deep neural network performance, making it necessary to introduce semantic-scale-balanced learning in medical image classification.

D.7 Remote Sensing Image Scene Classification

This section validates semantic-scale-balanced learning effectiveness in sample-balanced remote sensing image classification tasks, demonstrating the necessity of introducing semantic scale imbalance into remote sensing image recognition.

D.7.1 Dataset Introduction

- **RSSCN7 dataset** contains 2,800 remote sensing images classified into 7 typical scene categories: grassland, forest, farmland, parking lot, residential region, industrial region, and river and lake. Figure 13 [Figure 13: see original paper] shows the seven scenarios. Following official split, training and testing images are each 50% of total.
- **NWPU-RESISC45 dataset** contains 31,500 images with pixel size 256$×$256, covering 45 scene classes with 700 images per class. This dataset has large intra-class variability and inter-class similarity due to large differences in image spatial resolution, untitled pose, and illumination. Following official split, 20% of images are used for training and 80% for testing.

D.7.2 Backbone Network and Experimental Parameters We select VGG-16, GoogLeNet, and ResNet-34 as backbone networks. Adam optimizer (default parameters) updates the model until convergence, with learning rate decaying 10 times every 50 epochs. Batch size is set to 100 and no data augmentation is used throughout training.

D.7.3 Results on RSSCN7 We trained all backbone networks with dynamic semantic-scale-balanced learning. Results are shown in Figure 14 [Figure 14: see original paper]. Our method improves all models' performance. When our method is not employed, all backbone networks are significantly weaker in recognizing industrial regions than other scenes. Our method makes model recognition ability for different scenes more balanced, thus improving overall model performance.

Specifically, dynamic semantic-scale-balanced learning improves VGG-16' s recognition accuracy for industrial regions and parking lots by 4% and 3%, respectively, significantly reducing model bias. Dynamic semantic-scale-balanced learning also performs well on GoogLeNet and ResNet-34, improving GoogLeNet and ResNet overall accuracy by 1% and 0.6%, respectively.

D.7.4 Results on NWPU-RESISC45 We significantly improved multiple backbone networks' performance by employing dynamic semantic-scale-balanced learning on NWPU-RESISC45, with results illustrated in Figure 15 [Figure 15: see original paper].

VGG-16-DSB overall performance is 1.8% higher than VGG-16. Meanwhile, dynamic semantic-scale-balanced learning improves GoogLeNet and ResNet-34 overall performance by 1.6% and 1.2%, respectively.

Experiment Summary. On two sample-balanced remote sensing image datasets, our method shows significant improvements on common backbone networks. Results show semantic scale imbalance exists in remote sensing image datasets and affects deep neural network performance to some extent. Remote sensing images hold great promise for agriculture, industry, and military applications, so promoting deep neural network fairness on remote sensing images is crucial.

E Pseudo Code for Sample Volume

An image can be considered a point in sample space, with sample space dimension equal to image pixel count. The manifold distribution law considers that multiple images from a class are distributed around a low-dimensional manifold in sample space. We calculate each class' s corresponding manifold volume and call it sample volume. We provide sample volume calculation pseudo code in Algorithm 1. In this work, we resize images to (16, 16, 3) then calculate sample volume after flattening.

F Dynamic Semantic-Scale-Balanced Learning

F.1 DSB-NSM, DSB-ST and DSB-Focal Loss

Given embedding z of a sample and label yi, dynamic semantic-scale-balanced (DSB) loss can be expressed as:

DSB(z, yi) = L(z, yi), i = 1, 2, . . . , C,

To show how to combine general loss to generate dynamic semantic-scale-balanced loss, we improve NormSoftmax (NSM) cross-entropy loss and SoftTriple (ST) loss. NormSoftmax removes the bias term in the last linear layer and adds an L2 normalization module to inputs and weights before SoftMax loss. With [w1, w2, ⋯, wC] Rd×C as the last fully connected layer, DSB-NSM with temperature σ generated by embedding z can be written as:

DSB-NSM(z, yi) = -exp(wT exp(wT j z/σ)

The SoftTriple loss combined with semantic-scale-balanced term is expressed as DSB-ST(z, yi) = -exp(λ(D exp(λ(D z,yi - δ)) + P z,yi - δ)) exp(λD where λ is a scaling factor and δ is a hyperparameter. The relaxed similarity between embedding z and class c is defined as D , where k is the number of centers per class. exp(1 z,c = P γ zT wk γ zT wk exp(1

Focal loss purpose is applying small loss weights to high-classification-confidence samples, thus increasing hard low-confidence samples' loss proportion to overall loss. The α-balanced variant of Focal loss regulates loss proportion among

samples while assigning different weights to each class, denoted as FL(pt) = -αt(1 - pt)γ log(pt), where pt is the probability the sample belongs to the true class. When αt = 1, Focal loss transforms into DSB-Focal loss.

F.2 Dynamic Re-Weighting Training Framework

Given training samples X = [x1, x2, . . . , xN] containing C classes and corresponding labels Y = [y1, y2, . . . , yN], with Ni samples per class (i = 1, 2 . . . , C) and total sample count N. d-dimensional features extracted by CNNs are denoted Z = [z1, z2, . . . , zN] Rd×N. We conduct experiments for two task types: image classification and deep metric learning. In deep metric learning, 64-dimensional features are generally adopted, while network-extracted features in image classification tasks tend to be high-dimensional. For example, ResNet-50 extracts 2048-dimensional features, which occupies more video memory. Therefore, when saving historical features in classification tasks, one-dimensional average pooling is performed on all features to reduce feature dimension to 64, consistent with common deep metric learning feature dimensions while preserving distribution geometry (because pooling operations are translation invariant, rotation invariant, and scale invariant).

Below we describe the three-stage training framework in detail.

The three-stage training framework is shown in Figure 16: (1) In the first stage, all features and labels generated by the 1st epoch are stored in Q, denoted (cid:21) (cid:20) Z · · · Z1N · · · ZdN · · · R(d+1)×N. Q contains all samples' features and labels, but in early training stages, historical features have large drift from current features and cannot be directly used to calculate semantic scale. (2) The second stage corresponds to epochs 2 through n. At each iteration, the oldest mini-batch features and labels in Q are removed and those generated by the current iteration are stored. The goal is to continuously update features in Q until feature drift becomes small enough. We set n to 5 in experiments, and the original loss function is used in the first two stages. Figure 5 shows n' s effect on model performance. A larger n does not hurt model performance but only takes slightly more time. Experience suggests setting n to 5 is sufficient. (3) The third stage corresponds to epochs > n. At each iteration, semantic scales are calculated using features in Q after updating Q, and the original loss is re-weighted.

Algorithm 2 shows how to apply the dynamic re-weighting training framework using DSB-ST loss as an example.

The proposed three-stage training framework overcomes the difficulty of calculating class-wise semantic scales during training due to limited samples per batch. In fact, a simple brute-force method to achieve real-time class-wise semantic scale calculation is extracting features of all samples using the current model after each iteration. However, this would take considerable time—for example, when training ImageNet with batch size 512, one epoch contains about 2,500 iterations, meaning all features would need extraction 2,500 times, which

is unacceptable.

Table 11 : Comparison of DSB-ST and SoftTriple in memory consumption and training speed. Speed is measured by average iterations per second. Additional video memory consumption due to our method is almost negligible.

Dataset ImageNet-LT iNaturalist2018 Cars196 CUB-2011 Mini-ImageNet GPU Memory Training speed SoftTriple

3491 MB

Volume Formula for Low-Dimensional Parallel Hexahedra in High-Dimensional Space

In Section 3.2 of the main paper, we deduced from the singular value decomposition of the matrix $Z = [z_1, z_2, \dots, z_m] \in \mathbb{R}^{d\times m}$ composed of features that the volume $\mathrm{Vol}(Z)$ of the subspace spanned by z_i is proportional to $\sqrt{\det(ZZ^T)}$. Here, we assume that in $\mathbb{R}^d$, given m d-dimensional vectors, these vectors define a parallel hexahedron in $\mathbb{R}^n$. The problem is how to calculate this parallel hexahedron. For example, consider two vectors $z_1 = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$ and $z_2 = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$ in $\mathbb{R}^3$. The parallel hexahedron defined by these two vectors is a parallelogram in $\mathbb{R}^3$. We want to find a formula to calculate the area of this parallelogram. (Note that the true three-dimensional volume of this planar parallelogram is 0, just as the length of a point is 0 and the area of a line is 0. Here, we are trying to measure the two-dimensional "volume" of the parallelogram.)

Next, we introduce two special cases of parallel hexahedral volume. For a single vector $z = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \in \mathbb{R}^n$, its parallel hexahedron is itself. Here, "volume" means the length of the vector, and according to the Pythagorean theorem, its volume is $\sqrt{a_1^2 + a_2^2 + \cdots + a_n^2}$. Another case is when we have n vectors in $\mathbb{R}^n$. Suppose these n vectors are $z_1 = \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{n1} \end{bmatrix}, \dots, z_n = \begin{bmatrix} a_{1n} \\ a_{2n} \\ \vdots \\ a_{nn} \end{bmatrix}$. We know that the volume of the resulting parallel hexahedron is $\left| \det \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \right|$.

In the non-special case, the formula for the volume of a low-dimensional parallel hexahedron in a high-dimensional space will incorporate results (7) and (8). Here, we first present the final formula and then discuss why it is reasonable. Write the k vectors $z_1, \dots, z_k$ in $\mathbb{R}^n$ as column vectors. Let $Z =$

$[z_1, \dots, z_k] \in \mathbb{R}^{n\times k}$, and the volume of the parallel hexahedron derived from the vectors $z_1, \dots, z_k$ is $\sqrt{\det[Z^T Z]}$.

We now discuss why $\sqrt{\det[Z^T Z]}$ must be the volume in the general case.

Lemma 1. For a matrix $Z = [z_1, \dots, z_k]$, we have $Z^T Z = \begin{bmatrix} |z_1|^2 & z_1 \cdot z_2 & \cdots & z_1 \cdot z_k \\ z_2 \cdot z_1 & |z_2|^2 & \cdots & z_2 \cdot z_k \\ \vdots & \vdots & \ddots & \vdots \\ z_k \cdot z_1 & z_k \cdot z_2 & \cdots & |z_k|^2 \end{bmatrix}$, where $z_i \cdot z_j$ denotes the dot product of vectors z_i and z_j, and $|z_i| = \sqrt{z_i \cdot z_i}$ denotes the length of the vector.

The proof of Lemma 1 needs to focus only on $Z^T Z = [z_1, \cdots, z_k]^T [z_1, \cdots, z_k]$. If we apply any linear transformation that preserves angles and lengths in $\mathbb{R}^n$ (in other words, if we perform a rotation operation on $\mathbb{R}^n$), the numbers $|z_i|$ and $z_i \cdot z_j$ do not change. The set of all linear transformations that preserve angle and length in $\mathbb{R}^n$ forms a group, called the orthogonal group and denoted as $O(n)$. This allows us to reduce the problem to that of finding the volume of a parallel hexahedron in $\mathbb{R}^k$.

Proof. It is known that $\det[Z^T Z] = \begin{vmatrix} |z_1|^2 & z_1 \cdot z_2 & \cdots & z_1 \cdot z_k \\ z_2 \cdot z_1 & |z_2|^2 & \cdots & z_2 \cdot z_k \\ \vdots & \vdots & \ddots & \vdots \\ z_k \cdot z_1 & z_k \cdot z_2 & \cdots & |z_k|^2 \end{vmatrix}$. To prove that the above equation must be the formula for the volume, we first consider the standard basis of $\mathbb{R}^n$: $e_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, e_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \dots, e_n = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$. According to Lemma 1, we are able to find a rotation of $\mathbb{R}^n$ that maintains both length and angle, and also rotates our vectors $z_1, \dots, z_k$ such that they can be fully represented linearly by the first k standard vectors $e_1, \dots, e_k$ (which is geometrically reasonable). After the rotation, the latter $n-k$ dimensions of each vector z_i are 0. Therefore, we can think of our parallel hexahedron as consisting of k vectors in $\mathbb{R}^k$, and we already know how to calculate its volume, which is $\sqrt{\det[Z^T Z]}$.

H. Additional Analysis

In this section, we provide additional experimental results addressing three questions: (1) The effectiveness of dynamic semantic-scale-balanced learning without considering inter-class interference; (2) Comparison with other methods for measuring class-level difficulty; and (3) Dividing ImageNet into three subsets based on semantic scale, and demonstrating the performance of dynamic semantic-scale-balanced learning on these subsets.

H.1. Effectiveness of Dynamic Semantic-Scale-Balanced Learning Without Considering Inter-Class Interference

We have weakened the effect of inter-class interference when designing the measurement of semantic scale imbalance. The semantic scale of m classes after maximum normalization is assumed to be $S' = [S'_1, S'_2, \dots, S'_m]^T$, and the centers of all classes are $O = [o_1, o_2, \dots, o_m]^T$. We define the distance between the centers of class i and class j as $d_{i,j} = \|o_i - o_j\|_2$. The weights of m classes are written as $W' = [w_1, w_2, \dots, w_m]^T$. After maximum normalization and logarithmic transformation of W', we obtain $W = \log(\alpha + W')$, where $\alpha \geq 1$ is used to control the smoothing degree of W. After considering inter-class distance, the semantic scale $S = S' \odot W$, and the role of S' in dominating the degree of imbalance is greater when α is larger. The second-order derivative of the function $W = \log(\alpha + W')$ is less than 0, so the increment of W decreases as α increases. When the value of α is large, W' hardly works.

The Pearson correlation coefficients between class accuracy and inter-class interference, semantic scale, and semantic scale considering inter-class interference are shown in Table 1. It can be seen that the Pearson correlation coefficient between semantic scale and class accuracy on CIFAR-10-LT without considering inter-class interference still reaches 0.8688, while the correlation coefficient between inter-class distance W and class accuracy is only 0.2957, which illustrates the importance of semantic scale. In addition, we have added the correlation coefficients between the effective sample numbers and class accuracies in Table 1. It can be observed that the correlation between effective sample number and class accuracy is almost the same as the correlation between sample number and class accuracy, which is due to the fact that effective sample number is a monotonic function of sample number.

To demonstrate the performance of dynamic semantic-scale-balanced learning without considering inter-class interference in more detail, we conducted experiments on ImageNet-LT. The experimental settings are the same as those in Table 2. The dynamic semantic-scale-balanced loss without considering inter-class interference is denoted as DSB-CE-1 and DSB-Focal-1. The experimental results are shown in Figure 17 [Figure 17: see original paper]. It can be observed that DSB-CE-1 and DSB-Focal-1 have almost no performance degradation compared to DSB-CE and DSB-Focal. This observation is as expected, since our recent study shows that the correlation between the separation degree of feature manifolds and the accuracy of the corresponding class decreases during training, and that the existing model can eliminate the main effect of separation degree between feature manifolds on model bias.

H.2. Comparison with Other Methods of Measuring Class-Level Difficulty

Difficult example mining [47; 58] is an instance-level approach, while we focus on class-level difficulty. We note that a recent work on measuring class difficulty

(CDB loss [37]) was published in IJCV, which can be compared with our work. In addition, LOCE [14] and domain balancing [2] also measure class-level difficulty, but LOCE is designed for object detection tasks, so we compare semantic-scale-balanced learning with CDB loss and domain balancing. The description of CDB loss, LOCE, and domain balancing is as follows:

- The imbalance in class performance is referred to as the "bias"of the model, and [37] defines the model bias as bias $= \max\left(\frac{\max_{c=1}^{C} A_c}{\frac{1}{C}\sum_{c'=1}^{C} A_{c'}+\epsilon} - 1, 0\right)$, where A_c denotes the accuracy of the c-th class. When the accuracy of each class is identical, bias $= 0$. [37] computes the difficulty of class c using $1 - A_c$ and calculates the weights of the loss function using a nonlinear function of class difficulty.
- LOCE [14] uses the mean classification prediction score to monitor the learning status for different classes and applies it to guide class-level margin adjustment for enhancing tail-class performance [78].
- Domain balancing [2] studied a long-tailed domain problem, where a small number of domains (containing multiple classes) frequently appear while other domains exist less frequently. To address this task, this work introduced a novel domain frequency indicator based on the inter-class compactness of features, and uses this indicator to re-margin the feature space of tail domains [78].

We implemented CDB loss [37], LOCE [14], and domain balancing [2] on ImageNet-LT. As shown in Figure 18 [Figure 18: see original paper], our proposed semantic scale balanced learning outperforms these three approaches. In addition to the comparison with the class-level difficulty weighting method, we add the results of improvements to PaCO [10] in Table 2. Balanced softmax is included in PaCO, and although we have shown in Table 2 that our method significantly improves Balanced softmax, we still improved PaCO and conducted experiments to allay researchers' concerns. All experiments adopted the same training strategy and parameters as in Table 2.

H.3. Performance of Dynamic Semantic-Scale-Balanced Learning on Three Subsets of ImageNet

We divided ImageNet into Head, Middle, and Tail subsets based on semantic scale, which contain 333, 333, and 334 classes, respectively. The performance of DSB-CE and CE on the three subsets when the backbone networks are VGG-16 and ResNet-18 is shown in Figure 19 [Figure 19: see original paper].

The experimental results show that semantic-scale-balanced learning significantly improves the performance of CE on the Tail subset. Meanwhile, DSB-CE also outperforms CE on Head and Middle subsets, which may be caused by the performance gain from better feature learning. In addition to the classification problem, we hope to introduce semantic scale imbalance in the fields of object detection, semantic segmentation, etc. to promote the fairness of the model.

I. Applying Semantic Scale to Solve Other Problems

I.1. Selecting Well-Represented Data

Downsampling the head class is one of the methods to alleviate the long-tail problem, which balances the number of samples but leads to the loss of head class information. Therefore, it is important to develop a downsampling method that preserves head information. We propose an idea to select well-represented data based on the geometric meaning of semantic scale.

The existence of data manifolds is a consensus: the same class of data is usually distributed around a low-dimensional manifold. Different dimensions of the manifold represent different physical characteristics, and samples located at the edges of the manifold often tend to overlap with other manifolds. Therefore, we believe that the following two principles should be obeyed when downsampling:

- **Uniform sampling inside the manifold**: This ensures that the volume of the manifold does not shrink significantly after downsampling.
- **Increase the sampling rate of samples at the edges of the manifold**: This makes the sampled distribution have significant bounds, which helps improve the robustness of the classification model.

As shown in Figure 20 [Figure 20: see original paper], we refer to the strategy that obeys the above sampling principles as "pizza" sampling.

Uniform sampling is easy to implement, but how do we sample as many samples as possible from the edges of the manifold? We propose to randomly sample k subsets in the original sample set and calculate the semantic scales of the subsets. Then repeat the above operation several times and select the subsets with the largest semantic scales as the final samples.

I.2. Guiding Data Collection

When collecting data that has never been studied before, we do not know how many samples to collect to represent their corresponding class well because of the lack of prior knowledge. When too few samples are collected, the class is under-represented. Sampling too many samples will consume huge costs.

The marginal effect of the semantic scale can help us judge whether the currently collected samples have enough feature diversity, and we can stop collecting samples when the feature diversity tends to be saturated.

Specifically, the data collection process is as follows: 1. For class c, m samples are collected each time. 2. After the $(n-1)$-th collection of samples, there are $(n-1) \times m$ samples, and the semantic scales of these samples are calculated. 3. After the n-th collection of samples, there are $n \times m$ samples, and the semantic scales of these samples are calculated. 4. Calculate the increment of the semantic scale for the n-th time relative to the $(n-1)$-th time. 5. Calculate $(S_n - S_{n-1})/S_n$. If the increment of semantic scale is less than $\alpha\%$ of S_n, it

means that the feature diversity of class c has not changed significantly and the sample collection can be stopped.

The parameter α can be adjusted according to the needs of the task.

Geometric analysis of data manifolds can bring new perspectives to data science. We will open source the toolkit for measuring the information geometry of data, which includes the application of semantic scale in various scenarios, such as data collection and representative data selection.

J. Further Explanation of Figure 2

To see it more clearly, we zoomed in on Figure 2 and plotted it in Figure 24 [Figure 24: see original paper]. Previous studies have observed that: (1) given sufficient data, the classification performance gain is marginal with additional samples; (2) when data is insufficient, classification performance drops sharply as the number of training samples decreases. We speculate that phenomenon 1 may be caused by the marginal effect of feature diversity. It should be noted that CB loss considers marginal effects, but it only qualitatively describes the gradual flattening of feature diversity with increasing number of samples.

Taking CIFAR-10 as an example, we first select a few samples for each class, train the model, and test the accuracy. Then new samples are continuously added to the original samples instead of re-selecting more samples to train the model. The experiments corresponding to each point in Figure 2 are trained from scratch. While increasing the data, we find that there are marginal effects of semantic scale, which indicates that our proposed measurement is as expected. The marginal effects of feature diversity explain phenomenon 1.

However, phenomenon 2 is not explained by marginal effects, and the effective number of samples from CB loss does not predict phenomenon 2 at all, because the effective number of samples does not grow faster than the number of samples (which we have analyzed in Section 2). We experimentally find that when samples are few, the feature diversity measured by the semantic scale increases rapidly with the number of samples, and this increase is faster than linear. The rapid increase of feature diversity measured by the semantic scale explains phenomenon 2.

K. Can the Semantic Scale Capture the Hierarchical Structure?

HCSC [15] constructs the hierarchical structure of classes by bottom-up k-means, and we use the example shown by HCSC to validate our approach. Given the following seven classes: Poodles, Samoyeds, Labradors, Persian cats, Siamese cats, Chimpanzees, and Gorillas, each class contains 1,000 samples, and the hierarchical structure of the seven classes is shown in Figure 25 [Figure 25: see original paper].

We collect 1,000 images for each of the three parent classes (Dogs, Cats, and Monkeys), which can adequately represent the three parent classes (i.e., the feature richness is sufficient). Can the semantic scale be used to match the correct parent classes for the seven classes? According to our theory, the manifolds of the child classes should be within the manifold of the corresponding parent class, and they have an inclusion relationship. Therefore, when the data of the child classes are mixed into the data of the parent class, the manifold volume of the parent class will not change significantly. We propose a matching method of semantic hierarchy based on this property. The specific steps are as follows:

1. Train a ResNet-18 classification model on seven child classes. We set the batch size to 64 and adopt the Adam optimizer with a learning rate of 0.01 (linear decay), a momentum of 0.9, and a weight decay factor of 0.005.
2. Extract the features of all samples from seven child classes and three parent classes.
3. Calculate the semantic scales of the three parent classes.
4. Select a child class c from the seven child classes.
5. Mix the data of child class c into the data of each parent class and calculate the semantic scale of the mixed data, obtaining three values.
6. Calculate the changes in the semantic scales of the three parent classes and sort them.
7. Match the parent class with the smallest change in semantic scale for child class c.
8. Perform steps (3) to (7) for the remaining six child classes.

We summarize the ratio of the semantic scales of the parent classes after mixing to before mixing in Table 12 . If the change in the semantic scale of a parent class is small after a child class is mixed into that parent class, they are considered to have a nested relationship. Based on the above method, we successfully match each child class to its parent class. Experimental results show that our proposed measure of semantic scales can capture the semantic hierarchy of classes. Our study can inspire hierarchical feature learning as well as facilitate its performance in downstream tasks.

L. Future Work and Challenges

L.1. Model-Independent Measure of Data Difficulty

The performance of models varies across classes. In the past, it was believed that model bias was caused by an imbalance in sample numbers, but a growing body of research suggests that sample numbers are not the only factor affecting model bias. Of course, model bias is also not introduced by the model structure, but by the characteristics of the data itself that affect model performance. Therefore, it is very important to propose model-independent measurements to represent the data itself, and this work will greatly contribute to our understanding of deep neural networks. In this paper, the effect of the volume of the data manifold on model bias is explored from a geometric perspective. This provides a new

direction for future work, namely the geometric analysis of deep neural networks. The geometric characteristics of the data manifold will help us further reveal how neural networks learn and inspire the design of neural network structures.

L.2. A Geometric Perspective on Data Classification

Natural datasets have intrinsic patterns that can be generalized to the manifold distribution principle: the distribution of a class of data is close to a low-dimensional manifold. As shown in Figure 23 [Figure 23: see original paper], data classification can be regarded as the unwinding and separation of manifolds. When a data manifold is entangled with other perceptual manifolds, the difficulty of classifying that manifold increases. Typically, a deep neural network consists of a feature extractor and a classifier. Feature learning can be considered as manifold unwinding, and a well-learned feature extractor is often able to unwind multiple manifolds for the classifier to decode. In this view, all factors about manifold complexity may affect the model' s classification performance. Therefore, we suggest that future work can explore the inter-class long-tailed problem from a geometric perspective.

L.3. Introducing Semantic Scale Imbalance in Object Detection

Long-tailed distribution is one of the main difficulties faced by object detection algorithms in real-world scenarios. Classical object detection algorithms are generally trained on manually designed datasets with relatively balanced data distribution. In contrast, the accuracy of these algorithms tends to suffer significantly on long-tailed distributed datasets. So far, methods for foreground-background imbalance and class imbalance have been proposed extensively, but these methods define the degree of imbalance based on the number of objects and cannot explain more phenomena.

In the field of object detection, it is often encountered that although a class does not appear frequently, the model can always detect such instances efficiently. It is easy to observe that classes with simple patterns are usually easier to learn, even if the frequency of such classes is low. Therefore, classes with low frequency in object detection are not necessarily always harder to learn. We believe that it is a valuable research direction to analyze the richness of the instances contained in each class, and then pay more attention to the hard classes. The dimensionality of all images or feature embeddings in the image classification task is the same, which facilitates the application of the semantic scale proposed in this paper. However, the non-fixed dimensionality of each instance in the field of object detection brings new challenges, so we have to consider the effect of dimensionality on the semantic scale, which is a direction worthy of further study.

L.4. Challenges of Class Imbalance in Deep Learning

Class imbalance remains a major challenge in the field of deep learning. Data imbalance classification, although widely studied, still lacks effective and clear methods and guidelines. The problem of object detection for class imbalance is still in its infancy and requires greater investment of attention. In the following, we summarize the important future challenges and research directions in this field.

(1) More Precise Measure of Class Difficulty. An increasing number of studies have shown that sample number does not accurately reflect the accuracy of the model in recognizing classes. Therefore, more extensive measures should be proposed to redefine the long-tail distribution to facilitate classification and object detection tasks and further expand the scope of research on long-tailed recognition. For example, a dataset with perfectly balanced sample numbers may not be balanced under other measures.

(2) Long-Tailed Distribution of Properties Within Classes. As shown in Figure 24 [Figure 24: see original paper], previous studies have focused on the imbalance between classes and ignored the imbalance of properties within each class. For example, most pandas have black and white fur, and only a small proportion of pandas are brown. In visual recognition tasks, we should not only pursue the overall accuracy of the class but also pay attention to whether samples with sparse properties in a class can be classified accurately.

In medical image classification, this point is particularly important. For example, pulmonary diseases contain many different types of diseases, and generally the more severe diseases tend to have smaller sample numbers, suggesting that there is an imbalance of properties under the label of pulmonary disease. We hope to be able to recognize more severe diseases more accurately so that patients do not miss the best time for treatment.

(3) Generalization Performance of the Model Outside the Training Domain of the Tail Class. As shown in Figure 25 [Figure 25: see original paper], tail classes often have very few samples, so these samples do not well represent the true distribution of the tail classes, which results in the model consistently failing to learn and adapt to the tail classes correctly. Obviously, recovering the underlying distribution of the tail classes helps the generalization performance of the model outside the training domain of the tail classes. It is currently shown that similar classes have similar distribution statistics (variance), which can lead researchers to recover the underlying distribution of tail classes. However, the current research is still in its infancy, and it is not a sufficiently stringent assumption that similar classes have similar variances. Therefore, we hope that in the future researchers will be able to help recover the true distribution of tail classes by more means.

(4) How to Choose the Appropriate Long-Tailed Recognition Method for the Task. Up to now, a large number of visual recognition methods on long-

tail distribution have been proposed. While individual methods have positive performance in long-tailed recognition tasks, some combinations of methods may have negative effects. Few studies have focused on the selection and combination of different training techniques and methods. In the future, it is possible to explore how to select existing methods on specific tasks, and further, effective combinations of different methods are important.

(5) Multi-Domain Deep Long-Tailed Learning. Past research has typically focused on the problem of long-tailed distribution over a single domain, which has limited research ideas. As shown in Figure 26 [Figure 26: see original paper], data from multiple domains can complement each other to alleviate the long-tailed distribution of classes. For example, in plant and animal classification, cameras are placed in different places to capture animals, but some animals only appear in a fixed area, which leads to different label distributions for animals captured by different cameras. By combining the data from all cameras, a more balanced class distribution can be obtained. Similarly, a similar situation occurs in other practical applications. For example, in a visual recognition problem, the few classes from "photo" images can be complemented by potentially rich samples from "sketch" images. In autonomous driving, a few classes of "real" life accidents can be enriched by accidents generated in "simulations" . In addition, in medical diagnosis, data from different populations can be mutually augmented, e.g., a small sample from one institution can be combined with the majority of possible instances from other institutions.

In these examples, different data types can act as different domains, and such multi-domain data can also be utilized effectively to address data imbalances.

(6) Recognition of Unbalanced Data Streams. Continuous learning aims to process new data that is continuously generated in order to dynamically update and adapt the model to the latest data domain. Challengingly, as new data is generated, the degree of imbalance between classes changes and what used to be a tail class may become a head class. The long-tailed distribution of properties within classes can also affect the performance of the model if concept drift occurs. Thus, the key to handling unbalanced data streams is to evaluate the class-level difficulty and the long-tailed distribution of properties within classes in real time, which is a huge challenge.

(7) Augmentation Methods for Other Modalities of Unbalanced Data. Methods for multi-sample synthesis are widely used in image data augmentation, such as Mixup and Cutout, but there is still a lack of data augmentation methods for other modal data (e.g., speech and tabular). Researchers can design more general methods to generate samples of any type of data.

(8) Other Long-Tailed Visual Recognition Tasks. Current research focuses on long-tail image classification, while less attention has been paid to long-tail object detection, image segmentation, and regression tasks. In object detection, there are multiple imbalances, such as foreground-background imbalance and imbalance between classes belonging to the foreground, which

are unresolved challenges. With further applications of deep learning, research on imbalance learning in various fields will be of great benefit for real-world applications.

This study suggests some future avenues of inquiry to further deepen and expand the study of unbalanced learning. Of course, the scope of future inquiry into unbalanced learning is not limited to the eight challenges mentioned above, and we believe that new questions will arise in the course of inquiry into these challenges, but that researchers will eventually address them over time.

In everything balance has to be gained. Through balance you will come nearer to truth, because truth is the ultimate balance.

Note: Figure translations are in progress. See original paper for figures.