Transfer Knowledge from Head to Tail:
Uncertainty Calibration under Long-tailed Distribution

Jiahao Chen[1][2], Bing Su[1][2][†]

[1] Gaoling School of Artificial Intelligence, Renmin University of China
[2] Beijing Key Laboratory of Big Data Management and Analysis Methods

{nicelemon666, subingats}@gmail.com

Abstract

How to estimate the uncertainty of a given model is a crucial problem. Current calibration techniques treat different classes equally and thus implicitly assume that the distribution of training data is balanced, but ignore the fact that real-world data often follows a long-tailed distribution. In this paper, we explore the problem of calibrating the model trained from a long-tailed distribution. Due to the difference between the imbalanced training distribution and balanced test distribution, existing calibration methods such as temperature scaling can not generalize well to this problem. Specific calibration methods for domain adaptation are also not applicable because they rely on unlabeled target domain instances which are not available. Models trained from a long-tailed distribution tend to be more overconfident to head classes. To this end, we propose a novel knowledge-transferring-based calibration method by estimating the importance weights for samples of tail classes to realize long-tailed calibration. Our method models the distribution of each class as a Gaussian distribution and views the source statistics of head classes as a prior to calibrate the target distributions of tail classes. We adaptively transfer knowledge from head classes to get the target probability density of tail classes. The importance weight is estimated by the ratio of the target probability density over the source probability density. Extensive experiments on CIFAR-10-LT, MNIST-LT, CIFAR-100-LT, and ImageNet-LT datasets demonstrate the effectiveness of our method.

1. Introduction

With the development of deep neural networks, great progress has been made in image classification. In addition to performance, the uncertainty estimate of a given model is also receiving increasing attention, as the confidence of a model is expected to accurately reflect its performance.

A model is called *perfect calibrated* if the predictive confidence of the model represents a good approximation of its actual probability of correctness [9]. Model calibration is particularly important in safety-critical applications, such as autonomous driving, medical diagnosis, and robotics [1]. For example, if a prediction with low confidence is more likely to be wrong, we can take countermeasures to avoid unknown risks.

Most existing calibration techniques assume that the distribution of training data is balanced, i.e., each class has a similar number of training instances, so that each class is treated equally [9, 18, 25]. As shown in Fig. 1, the traditional calibration pipeline uses a balanced training set to train the classification model and a balanced validation set to obtain the calibration model, respectively. The target test set is in the same distribution as the training/validation set. However, data in the real-world often follows a long-tailed distribution, i.e., a few dominant classes occupy most of the instances, while much fewer examples are available for most other classes [6, 16, 24]. When tested on balanced test data, classification models trained from the training set with a long-tailed distribution are naturally more overconfident to head classes. Only imbalanced validation set with the same long-tailed distribution is available for calibrating such models since the validation set is often randomly divided from the training set.

Due to the different distributions between the imbalanced training data and the balanced test data [15], it is difficult for traditional calibration techniques to achieve balanced calibration among head classes and tail classes with different levels of confidence estimations. For instance, temperature scaling [9] with the temperature learned on a validation set obtains degraded performance on the test set if the two sets are in different distribution [29, 37]. As shown in Fig. 2, a balanced test set suffers heavier overconfidence compared with a long-tailed validation set. Although temperature scaling can relieve such phenomenon, there still exists overconfidence after calibration. Domain adaptation calibration methods [29, 40] aim to generalize

---
[†]Corresponding author.

(a) Calibration under balanced distribution.
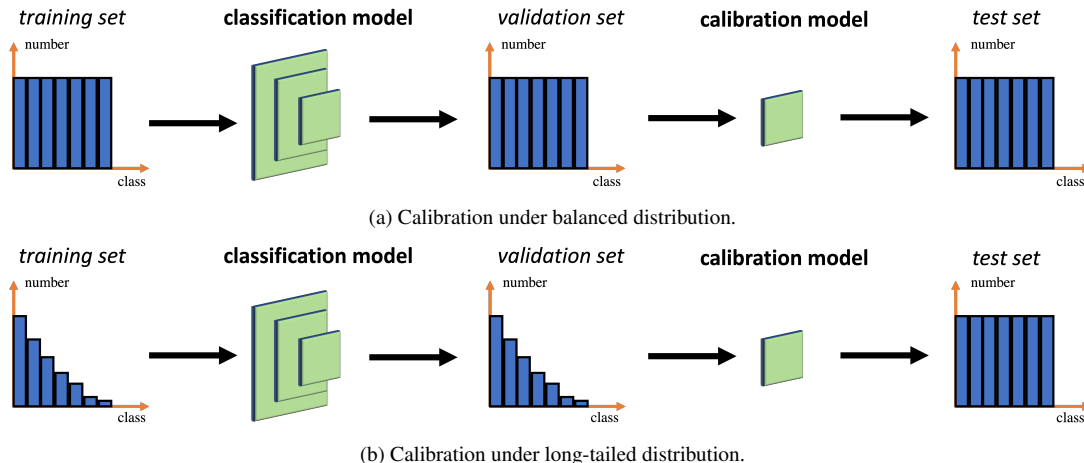

(b) Calibration under long-tailed distribution.

Figure 1. The difference between calibration under balanced distribution and calibration under long-tailed distribution. (a) The classification model and the calibration model are trained on the balanced training and validation sets, respectively, and the test set is balanced. (b) The classification model and the calibration model are trained on the long-tailed training and validation sets, while the test set is balanced.

(a) Validation set   (b) Test set   (c) Temperature scaling
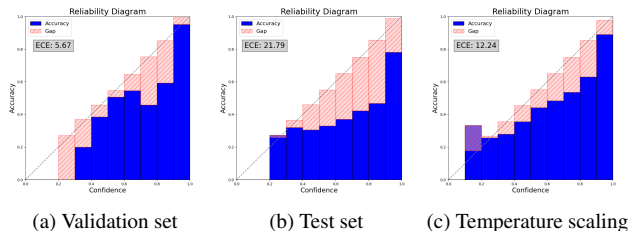
Figure 2. The reliability diagrams of (a) the validation set before calibration, (b) the test set before calibration, and (c) the test set after calibration with temperature scaling.

calibration across domains under a covariate shift condition but they utilize unlabeled target domain instances. Similarly, the domain generalization calibration method [8] uses a support set to bridge the gap between the source domain and the target domain, which also relies on extra instances. These methods cannot be applied to the long-tailed calibration since the balanced test domain is not available.

In this paper, we investigate the problem of *calibration under long-tailed distribution*. Since the distributions of each tail class in the imbalanced validation set and the balanced target set are different, we utilize the importance weight strategy to alleviate the unreliable calibration for tail classes. The weight of each instance is the ratio between the target balanced probability density and the source imbalanced probability density. We explicitly model the distribution of each class as a Gaussian distribution. Different from the source distribution, the target balanced distribution cannot be estimated directly. Since there exists common information between head classes and tail classes [23], we transfer knowledge from head classes to estimate the target probability density. Normally, the more similar two classes are, the more information they share. Therefore, for each

tail class, we measure the similarities between the distributions of it and all head classes with the Wasserstein distance. The similarity considers both first-order and second-order statistics of the two classes and thus can better reflect the transferability of statistical knowledge. Then we estimate the target probability density of each tail class by combining its own distribution and the transferred information from all head classes referring to the similarities. Finally, we calibrate the model with the importance weights.

Our contributions are summarized as: 1) We explore the problem of calibration under long-tailed distribution, which has important practical implications but is rarely studied. We apply the importance weight strategy to enhance the estimation of tail classes for more accurate calibration. 2) We propose an importance weight estimation method by viewing distributions of head classes as prior for distributions of tail classes. For each tail class, our method estimates its probability density function from the distribution calibrated by head classes and calculates the importance weight to realize balanced calibration. 3) We conduct extensive experiments on the CIFAR-10-LT, CIFAR-100-LT [3], MNIST-LT [20], ImageNet-LT [24] datasets and the results demonstrate the effectiveness of our method.

2. Related work

Post-processing calibration. Current calibration techniques can be roughly divided into post-processing methods and regularization methods [4, 11]. Post-processing methods focus on learning a re-calibration function on a given model. Platt scaling [31] transforms outputs of a classification model into a probability distribution over classes. It can solve the calibration of non-probabilistic methods like SVM [5]. Temperature scaling [9] extends Platt scaling and is applied to multi-class classification problems. It opti-

mizes a parameter T to re-scale the output logits of a given model. Non-parametric isotonic regression [43] learns a piece-wise constant function that minimizes the residual between the calibrated prediction and the labels. In [44], three properties, accuracy-preserving, data-efficient, and expressive of uncertainty calibration are proposed. Experiments show that a combination of the non-parametric method and the parametric method can achieve better results.

Domain shift calibration. The most common case is that the validation and test sets are in different domains [39]. The re-calibration function learned by the validation domain cannot be generalized to the test domain. CPCS [30] utilizes importance weight to correct for the shift from the training domain to the target domain and achieves good calibration for the domain adaptation model. TransCal [40] achieves more accurate calibration with lower bias and variance in a unified hyperparameter-free optimization framework. In [8], a support set is applied to bridge the gap between the source domain and a target domain, and three calibration strategies are proposed to achieve calibration for domain generalization. In [37], current techniques have demonstrated that overconfidence is still existing under domain shift, and a simple strategy where perturbations are applied to samples in the validation set before performing the post-hoc calibration step is proposed.

Although long-tailed calibration also suffers the domain shift problem, it is different from domain shift calibration since unlabeled target domain instances or plenty of data to constitute a support set are not available for calibration. There are some works [42,45] to improve calibration under long-tailed distribution by modifying the training objective to train new models from scratch. In contrast, our method focuses on calibrating existing models using a small validation set. Previous methods such as Gistnet [21] use head classes' knowledge to enhance tail classes' generalization, but they concentrate on long-tailed classification while we address the calibration issue.

3. Method

3.1. Notation

We propose the problem of *calibration under long-tailed distribution*. Given a long-tailed distribution $p(\boldsymbol{x})$ and a corresponding balanced distribution $q(\boldsymbol{x})$, we hold the assumption that $p(\boldsymbol{x}) \neq q(\boldsymbol{x})$ while $p(y|\boldsymbol{x}) = q(y|\boldsymbol{x})$. Instances are i.i.d. sampled from $p(\boldsymbol{x})$ to construct a long-tailed training set $\mathcal{S} = \{(\boldsymbol{x}_i, y_i)\}$ and a validation set $\mathcal{V}$, where $y_i \in \{1, \cdots, C\}$ is the label of the ith instance $\boldsymbol{x}_i$, C is the number of classes, and n_c denotes the number of instances belongs to the cth class. Similarly, instances are i.i.d. sampled from $q(\boldsymbol{x})$ to construct a balanced test set $\mathcal{T}$. Assuming classes are sorted by decreasing cardinality, i.e., $n_1 \geq n_2 \geq ... \geq n_C$, the data follows a long-tailed

distribution where most instances belong to head classes, while each tail class has only a few instances. We divide all classes into head classes $\mathcal{A}_{head} = \{c|n_c \geq \zeta\}$ and tail class $\mathcal{A}_{tail} = \{c|n_c < \zeta\}$, where ζ is a threshold. Moreover, we have been given a classification model $\phi(\cdot)$ trained on $\mathcal{S}$, where the output of $\phi(\boldsymbol{x}_i)$ is denoted by $\boldsymbol{z}_i$ and the corresponding feature (the output of the layer before the classifier) is denoted by $\boldsymbol{f}_i$. The goal is to calibrate the model $\phi(\cdot)$ on the validation set $\mathcal{V}$ so that the model is calibrated on the balanced test data $\mathcal{T}$.

3.2. Background

Calibration. For each instance $\boldsymbol{x}_i$, we acquire its confidence score $\hat{p}_i$ and prediction result $\hat{y}_i$ from the output $\boldsymbol{z}_i$. Formally, if the following Eq. (1) is satisfied, the model $\phi(\boldsymbol{x}_i)$ is called perfect calibrated. The definitions of $\hat{p}_i$ and $\hat{y}_i$ are in Eq. (2) and $softmax(\boldsymbol{z}_i)$ denotes the softmax function $softmax(\boldsymbol{z}_i) = \frac{exp(\boldsymbol{z}_i)}{\sum_{j=1}^{C} exp(\boldsymbol{z}_j)}$.

$$\mathbb{P}(\hat{y}_i = y_i | \hat{p}_i = p) = p \qquad \forall p \in [0, 1] \qquad (1)$$

$$\hat{p}_i = \max softmax(\boldsymbol{z}_i) \quad \hat{y}_i = \underset{\{1, 2, \cdots, C\}}{\arg \max} \; softmax(\boldsymbol{z}_i) \tag{2}$$

This formulation means that for example 20% of all predictions with a confidence score of 80% should be false.

Temperature scaling. As shown in Eq. (3), temperature scaling [9] fits a single parameter T from the validation set and applies it to other test sets.

$$T^* = \underset{T}{\arg \min} \, \mathbb{E}_p[\mathcal{L}(s(\boldsymbol{z}_i/T), y_i)] \tag{3}$$

Similar to the training classification task, the loss function $\mathcal{L}(\cdot)$ for calibrating the temperature is the Cross Entropy loss. Since the validation set also follows long-tailed distribution while the test set does not, the learned parameter T is difficult to generalize well to the test set.

3.3. Calibration under long-tailed distribution

To tackle the generalization issue of the original temperature scaling in calibration under long-tailed distribution, we propose our knowledge-transferring-based temperature scaling method to achieve cross-distribution generalization. The calibration loss on the balanced target distribution $q(\boldsymbol{x})$ can be reformulated as the weighted calibration error of the source distribution $p(\boldsymbol{x})$:

$$
\begin{aligned}
\mathbb{E}_q[\mathcal{L}(s(\boldsymbol{z}_i/T), y_i)] &= \int_q q(\boldsymbol{x}_i)\mathcal{L}(s(\boldsymbol{z}_i/T), y_i)dx \\
&= \int_p \frac{q(\boldsymbol{x}_i)}{p(\boldsymbol{x}_i)}p(\boldsymbol{x}_i)\mathcal{L}(s(\boldsymbol{z}_i/T), y_i)dx \\
&= \mathbb{E}_p[w(\boldsymbol{x}_i)\mathcal{L}(s(\boldsymbol{z}_i/T), y_i)]
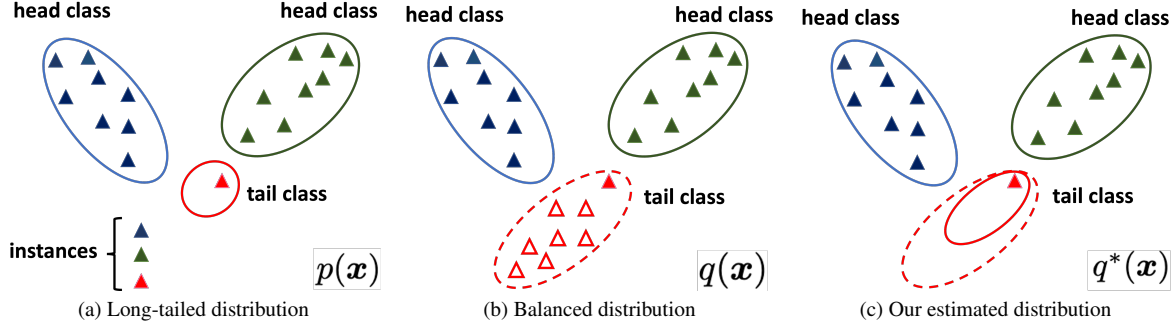\end{aligned}
\tag{4}
$$

Figure 3. (a) The long-tailed distribution $p(\boldsymbol{x})$. (b) The balanced distribution $q(\boldsymbol{x})$. Compared with (a), the distributions of head classes are the same, while the distributions of tail classes are not. (c) Our estimated distribution $q^*(\boldsymbol{x})$. With the help of head classes, we can estimate the distribution of tail classes and acquire their density.

As shown in Eq. (4), we can acquire the target distribution error $\mathbb{E}_q[\mathcal{L}(s(\boldsymbol{z}_i/T), y_i)]$ by estimating the ratio of probabilities $w(\boldsymbol{x}) = q(\boldsymbol{x})/p(\boldsymbol{x})$ for each instance. Domain adaptation calibration like TransCal [40] utilizes LogReg [2, 32] to estimate the ratio of density. It estimates the density by training a logistic regression classifier that realizes binary classification of source and target domains. Such methods cannot be used for the long-tailed calibration problem since the balanced distribution of test data is unknown and thus binary classification cannot be applied.

We model the distributions $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$ as mixtures of Gaussian distributions, respectively, by modeling each class as a Gaussian distribution. As shown in Fig. 3, because head classes have plenty of instances in both $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$, the distributions of head classes in $q(\boldsymbol{x})$ can be viewed as the same as those in $p(\boldsymbol{x})$, which can be easily estimated from the training set. However, each tail class only has a few instances in $p(\boldsymbol{x})$ while sufficient instances are available in $q(\boldsymbol{x})$, the distributions of tail classes are different in $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$. Since it is difficult to acquire the balanced distribution $q(\boldsymbol{x})$, we constitute the estimated distribution $q^*(\boldsymbol{x})$ to approach the truth distribution $q(\boldsymbol{x})$, where the key is to approximate the probability density value of each instance in tail classes under $q^*(\boldsymbol{x})$ from the estimated Gaussian distributions of tail classes in $p(\boldsymbol{x})$.

For each class $c \in \{1, 2, \cdots, C\}$, the corresponding distribution is $p_c(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$, where the mean $\boldsymbol{\mu}_c$ and the variance $\boldsymbol{\Sigma}_c$ are calculated by the set of output features belonging to class c in the training set. When $c \in \mathcal{A}_{tail}$, its distribution $p_c(\boldsymbol{x})$ is not reliable due to the limited instances. It is crucial to estimate the probability density under balanced distribution. Since there exists common information between head classes and tail classes [21–23], it is rational to transfer knowledge from all head classes to the cth tail class and recover its balanced distribution to the utmost. Normally, the more similar two classes are, the more information they share. We measure the similarity of the

two classes by the Wasserstein distance to consider the first-order and second-order statistics. We construct a distance vector $\boldsymbol{d}_c \in \mathbb{R}^{|\mathcal{A}_{head}|}$, where $\boldsymbol{d}_c^k$ is the kth element of $\boldsymbol{d}_c$ and denotes the similarity between class c and head class k.

$$\boldsymbol{d}_c^k = Wasserstein(p_c(\boldsymbol{x}), p_k(\boldsymbol{x})) \qquad (5)$$

Following the attention mechanism [38], we calculate the attention score $\boldsymbol{s}_c$ from the similarities as follows,

$$\boldsymbol{s}_c = softmax(-\frac{\boldsymbol{d}_c}{\sqrt{dim(\boldsymbol{f})}}) \qquad (6)$$

where the $dim()$ function acquires the feature dimension. A head class will be assigned a large attention score if its distribution is similar to the distribution of tail class c.

We calibrate the distribution of each tail class by transferring knowledge from all head classes based on $\boldsymbol{s}_c$. Specifically, for the cth tail class, the statistics of its calibrated distribution are estimated as follows.

$$\boldsymbol{\mu}_{c^*} = \alpha\boldsymbol{\mu}_c + (1-\alpha) \sum_{k \in \mathcal{A}_{head}} \boldsymbol{s}_c^k \boldsymbol{\mu}_k$$
$$\sqrt{\boldsymbol{\Sigma}}_{c^*} = \alpha\sqrt{\boldsymbol{\Sigma}}_c + (1-\alpha) \sum_{k \in \mathcal{A}_{head}} \boldsymbol{s}_c^k \sqrt{\boldsymbol{\Sigma}}_k \qquad (7)$$

The estimated distribution $\mathcal{N}(\boldsymbol{\mu}_{c^*}, \boldsymbol{\Sigma}_{c^*})$ contains the information of all head classes according to their similarity scores in $\boldsymbol{d}_c$, where α is a hyper-parameter.

For each instance $\boldsymbol{x}_i$ in tail classes, we can then acquire its probability under the estimated distribution $q^*_{y_i}(\boldsymbol{x}_i) = \mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_{y_i^*}, \boldsymbol{\Sigma}_{y_i^*})$. Based on the estimated $q^*_{y_i}(\boldsymbol{x}_i)$, the importance weight is defined in Eq. (8).

$$w^*(\boldsymbol{x}_i) = \begin{cases} 1 & y_i \in \mathcal{A}_{head} \\ min(max(\frac{q^*_{y_i}(\boldsymbol{x}_i)}{p_{y_i}(\boldsymbol{x}_i)}, \eta_1), \eta_2) & y_i \in \mathcal{A}_{tail} \end{cases} \qquad (8)$$

For each instance in head classes, the importance weight equals 1 since head classes in the two distributions are the

same. For each instance in tail classes, the importance weight equals $q^*_{y_i}(\boldsymbol{x}_i)/p_{y_i}(\boldsymbol{x}_i)$. In practice, we restrict the value of the weight from η_1 to η_2 to avoid abnormal values. Empirically, we set $\eta_1 = 0.3$ and $\eta_2 = 5.0$.

By using the importance weight to bridge the training long-tailed distribution and the test balanced distribution, we learn the temperature T in the final softmax layer on the validation set to calibrate the classification confidence. The final optimization function is shown in Eq. (9).

$$T^* = \arg\min_T \mathbb{E}_p[w^*(\boldsymbol{x}_i)\mathcal{L}(s(\boldsymbol{z}_i/T), y_i)] \tag{9}$$

To better understand our calibration method, we summarize the detailed procedures in Appendix 3.

3.4. Analysis

Theorem 3.1 *We denote the distribution of the kth class in the long-tailed distribution, the ground truth balanced distribution, and the estimated distribution by $p_k(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_{p_k}, \boldsymbol{\Sigma}_{p_k})$, $q_k(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_{q_k}, \boldsymbol{\Sigma}_{q_k})$, and $q^*_k(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_{q^*_k}, \boldsymbol{\Sigma}_{q^*_k})$, respectively. The absolute error $|\mathbb{E}_{p_k}[w_k(\boldsymbol{x})\mathcal{L}(s(\boldsymbol{z}/T), y)] - \mathbb{E}_{p_k}[w^*_k(\boldsymbol{x})\mathcal{L}(s(\boldsymbol{z}/T), y)]|$ is sensitive to $\epsilon = \mathbb{E}_{p_k}[(w_k(\boldsymbol{x}) - w^*_k(\boldsymbol{x}))^2]$ and the bound of ϵ is shown as follows, where the formula $d_2(\cdot||\cdot)$ presents the exponential in base 2 of the Renyi-divergence [34].*

$$\epsilon \in [(\sqrt{d_2(q_k||p_k)} - \sqrt{d_2(q^*_k||p_k)})^2, d_2(q_k||p_k) + d_2(q^*_k||p_k)] \tag{10}$$

The proof is provided in Appendix 1. Theorem 3.1 presents the error bound of our method, which is closely related to Renyi-divergence. It is obvious that when $q_k = q^*_k$, the lower bound reaches the minimum and equals 0. Therefore, our estimation method aims to keep q^*_k approaching q_k to reduce the calibration error on the test data.

We also explore how the importance weight influences the calibration compared with temperature scaling. For simplicity, we constitute one-dimensional Gaussian distribution $p(x) = \mathcal{N}(x|\mu_a, \sigma_a^2)$ and $q(x) = \mathcal{N}(x|\mu_b, \sigma_b^2)$, where $\mu_a \neq \mu_b$ and $\sigma_a^2 < \sigma_b^2$. We draw the conclusion that the value of $w(x) > 1$ if $\tau_1 > x$ or $x > \tau_2$, and $w(x) < 1$ if $\tau_1 < x < \tau_2$, where τ_1 and τ_2 are calculated as:

$$\Delta = \sqrt{(\mu_a - \mu_b)^2 + (\sigma_b^2 - \sigma_a^2)(\ln \sigma_b^2 - \ln \sigma_a^2)}$$
$$\tau_1 = \frac{\mu_a \sigma_b^2 - \mu_b \sigma_a^2 - \sigma_a \sigma_b \Delta}{\sigma_b^2 - \sigma_a^2} \tag{11}$$
$$\tau_2 = \frac{\mu_a \sigma_b^2 - \mu_b \sigma_a^2 + \sigma_a \sigma_b \Delta}{\sigma_b^2 - \sigma_a^2}$$

The proof is presented in Appendix 2. Normally, instances clustered around the mean are more likely to be classified correctly. Therefore, the instances whose importance weight $w(x) < 1$ are likely to be correctly classified while

$w(x) > 1$ on the contrary. In practice, a model trained with imbalanced data generalizes well for head classes but easily overfits tail classes, and hence obtains degraded performances on balanced test data. Our importance weight estimation method assigns larger weights to instances of tail classes that are classified incorrectly.

4. Experiment

4.1. Datasets

CIFAR-10-LT. CIFAR-10-LT [3] is simulated from balanced CIFAR-10 [17]. We conduct experiments with different imbalance factors (IF) and generate three imbalanced datasets with IF=100, IF=50, and IF=10, respectively. For each dataset, we randomly split 80% instances as the training set and 20% as the validation set. For comparison, we use four test sets: (1) original CIFAR-10 test set, (2) CIFAR-10.1 [33], (3) CIFAR10.1-C [12]: 95 synthetics datasets generated on CIFAR-10.1 with different transformations, (4) CIFAR-F [36]: 20 real-word test sets collected from Flickr. **MNIST-LT**. MNIST-LT is simulated from MNIST [20]. Similar to CIFAR-10-LT, we generate three imbalanced datasets with IF=100, IF=50, and IF=10, respectively. For comparison, we use four test sets: (1) original MNIST test set, (2) SVHN [26], (3) USPS [14], (4) Digital-S [36]: 5 test sets that are searched from Shutterstock based on different options of color. Note that the original MNIST test set is slightly imbalanced, which is closer to reality. **CIFAR-100-LT**. CIFAR-100-LT [3] is generated from the CIFAR-100 dataset. We generate imbalanced datasets with IF=10 and conduct experiments on the original CIFAR-100 test set. **ImageNet-LT**. ImageNet-LT [24] is simulated from ImageNet [7]. We merge the long-tailed training set and balanced validation set from the original ImageNet-LT. Following the principle of CIFAR-10-LT, we generate a long-tailed training set and a long-tailed validation set. We conduct experiments on a balanced test set. More details are presented in Appendix 4.

4.2. Experiments set up

Classification model. We use our method to calibrate different classification models. For CIFAR-10-LT and MNIST-LT, we use ResNet-32 [10] and LeNet-5 [20] as the classification model, respectively. To verify our method can be applied to different models, we apply ResNet-32, DenseNet-40 [13], VGG-19 [35] as classification models and test on CIFAR-100-LT, respectively. We do experiments on the large-scale dataset, ImageNet-LT, with ResNet-50. Details about training strategies are presented in Appendix 4. **Metrics/Baselines**. The most popular evaluation metric for calibration is ECE [25]. Besides, we also use SCE [28] and ACE [27] as evaluation metrics. We compare our method with temperature scaling

IF	Dataset	Method								
		Base	TS	ETS	TS-IR	IR	IROvA	SBC	GPC	Ours
IF=100	CIFAR-10	21.79	12.24	12.16	11.64	12.36	13.36	12.13	11.65	**9.84**
	CIFAR-10.1	28.97	16.75	16.70	16.65	17.13	17.93	16.78	15.71	**13.86**
	CIFAR-10.1-C	58.22	43.01	43.00	43.05	43.34	43.83	42.53	41.98	**39.58**
	CIFAR-F	29.22	15.27	15.24	15.52	15.75	16.23	15.45	14.18	**12.15**
IF=50	CIFAR-10	17.36	7.65	8.04	8.22	9.75	9.45	7.55	7.78	**3.99**
	CIFAR-10.1	22.79	10.36	10.99	11.72	13.35	12.70	10.32	10.82	**5.74**
	CIFAR-10.1-C	55.52	38.66	39.9	40.16	41.58	40.76	38.94	39.39	**33.09**
	CIFAR-F	25.37	11.30	12.21	12.67	14.39	13.37	11.4	11.76	**6.64**
IF=10	CIFAR-10	8.39	2.23	1.64	2.03	2.29	2.42	2.49	2.01	**1.00**
	CIFAR-10.1	13.80	4.87	4.25	4.54	5.38	5.23	5.63	4.66	**3.95**
	CIFAR-10.1-C	48.31	32.77	31.07	32.11	32.29	31.94	33.16	31.37	**29.98**
	CIFAR-F	19.73	8.15	6.80	8.42	8.97	8.13	8.54	7.10	**5.97**

Table 1. The ECE (%) on CIFAR-10-LT.

IF	Dataset	Method								
		Base	TS	ETS	TS-IR	IR	IROvA	SBC	GPC	Ours
IF=100	MNIST	2.52	1.27	1.84	2.82	2.84	1.84	1.92	1.76	**1.08**
	SVHN	16.06	7.20	11.62	21.25	22.18	14.93	9.59	13.67	**6.09**
	USPS	15.00	9.52	12.25	13.25	13.62	10.58	10.10	11.44	**8.40**
	Digital-S	32.10	22.13	27.35	30.13	31.01	27.48	23.34	27.60	**20.28**
IF=50	MNIST	1.12	0.85	1.14	1.53	1.54	1.02	1.01	1.12	**0.79**
	SVHN	**2.32**	3.95	3.33	11.42	12.15	2.63	9.43	**2.32**	4.53
	USPS	11.21	8.14	12.81	11.89	11.91	10.54	8.57	11.21	**8.02**
	Digital-S	15.22	10.81	17.81	20.96	21.81	13.64	16.74	15.18	**10.34**
IF=10	MNIST	0.56	0.23	**0.21**	0.50	0.52	0.23	0.25	0.41	0.36
	SVHN	5.75	6.76	6.94	8.10	**4.51**	5.31	7.00	5.31	7.43
	USPS	8.29	4.81	4.60	6.59	6.98	4.76	5.12	5.88	**4.55**
	Digital-S	13.55	8.21	8.09	15.37	13.34	8.31	7.67	8.24	**7.37**

Table 2. The ECE (%) on MNIST-LT.

(TS) [9], ETS [44], TS-IR [44], IROvA [43], IRM [44], Scaling-binning calibrator (SBC) [19] and GPC [41]. As for our method, all the experiments are conducted with hyper-parameter $\alpha = 0.998$ if not specified. Code is available: https://github.com/JiahaoChen1/Calibration

4.3. Results

CIFAR-10-LT. As shown in Tab. 1, our method achieves the best performance on the CIFAR-10-LT dataset. Usually, the model trained with a heavier imbalanced dataset suffers heavier miscalibration. Our method can realize competitive results in different situations. Since CIFAR-10.1 and CIFAR-F are collected from the real-world, the excellent results indicate that our method can generalize to different domains. The results of CIFAR-10.1-C also verify that our method can handle datasets containing different transformations. **MNIST-LT**. Tab. 2 demonstrates the effectiveness of our method on MNIST-LT. Except for (IF=50: SVHN) and

(IF=10: MNIST, SVHN), our method achieves the best performance. Although our method obtains negative optimization on (IF=50: SVHN) and (IF=10: SVHN), and lower performance compared with ETS on (IF=10: MNIST), the performance of our method is still acceptable. Overall, our method outperforms other methods significantly in most cases. **CIFAR-100-LT**. For the CIFAR-100-LT benchmark, our method achieves the best results on calibrating different models as shown in Tab. 3. We do experiments on three different architectures including ResNet, DenseNet, and VGG. Compared with DenseNet, the performance gains of ResNet and VGG are even higher, and our method achieves the smallest ECE on ResNet. **ImageNet-LT**. As shown in Tab. 4, our method achieves the best performance on the ImageNet-LT benchmark. This indicates that our method can be scaled to large-scale datasets. For detail, our method reduces the ECE value from 10.18% to 3.45% while the second best method, GPC, can only reduce it to 5.46%. For all datasets, the results of SCE and ACE are presented in

Model	Dataset	Method								
		Base	TS	ETS	TS-IR	IR	IROvA	SBC	GPC	Ours
ResNet-32	CIFAR-100	20.38	2.50	2.10	6.07	9.35	5.92	6.74	3.27	**1.50**
DenseNet-40	CIFAR-100	16.00	3.43	2.51	5.57	8.42	5.76	5.96	2.73	**2.37**
VGG-19	CIFAR-100	27.86	3.81	2.36	6.35	10.35	6.66	8.03	3.82	**1.99**

Table 3. The ECE (%) on CIFAR-100-LT.

Model	Dataset	Method								
		Base	TS	ETS	TS-IR	IR	IROvA	SBC	GPC	Ours
ResNet-50	ImageNet	10.18	6.72	6.06	10.23	11.15	7.63	9.12	5.46	**3.45**

Table 4. The ECE (%) on ImageNet-LT.
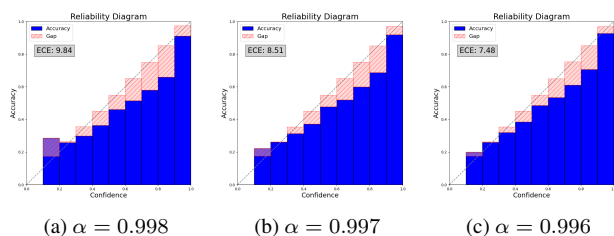
(a) $\alpha = 0.998$ (b) $\alpha = 0.997$ (c) $\alpha = 0.996$

Figure 4. The reliability diagram of our method with (a) $\alpha = 0.998$, (b) $\alpha = 0.997$, and (c) $\alpha = 0.996$.

Appendix 6 and similar conclusions can be observed.

4.4. Ablation study

Reliability diagram. We also visualize the reliability diagram of our method with different α on the CIFAR-10 test set. As shown in Fig. 4, our method can give more reliable results and alleviate the overconfidence problem. The reliability diagrams of the baseline and TS are shown in Fig. 2, and our method achieves competitive results compared with them. Specifically, a higher value of α achieves better results in the reliability diagram on this benchmark. Compared with TS, our method well calibrates the predictions for instances with high confidence. We also observe that our method leads to slight underconfidence in instances with very small confidence values, this may be because these samples themselves are more difficult to classify.

The visualization of attentions. Our method utilizes the distance between different distributions and acquires the attention score by the softmax function. As shown in Fig. 5, each tail class has multiple similar head classes and their knowledge will be used. For example, the top two similar classes of tail class "woman" are "girl" and "boy", while the not similar classes are "cloud" and "castle". It is obvious that "woman" has much similar information with "girl" and "boy" and we utilize more information from such classes is
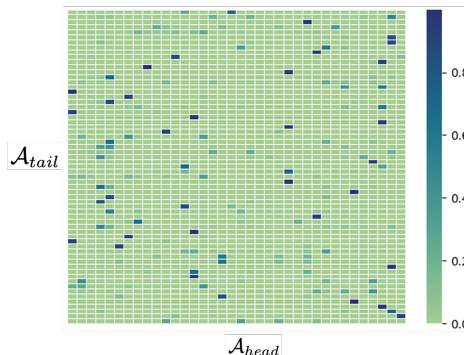

More ablation studies are in Appendix 5

Figure 5. Experiment on CIFAR-100-LT. The shape of the matrix is $|\mathcal{A}_{tail}| \times |\mathcal{A}_{head}|$. Each row denotes the attention vector $\boldsymbol{s}$.

rational. In addition, we do experiments with two different knowledge-transferring strategies. One strategy is that we only transfer knowledge from the most similar head class and discard other head classes, which is called OneHot. The other strategy is that we transfer knowledge from all head classes equally and ignore their inherent differences, which is called Uniform. The results in Tab. 5 show the utility of our strategy and prove the importance to consider the knowledge of all head classes according to their relevance.

The distribution of $w^*(\boldsymbol{x})$. Our method is heavily influenced by the value of $w^*(\boldsymbol{x})$, so we explore the distribution of $w^*(\boldsymbol{x})$ on different datasets. We do experiments on CIFAR-10-LT with IF=100, IF=50, and IF=10, respectively. As shown in Fig. 6, the overall distribution of w is clustered around the value $w = 1$. The more imbalanced the dataset, the more instances with larger w values. More instances have larger values of w at IF=100 than at IF=10. Since the dataset with IF=100 suffers a heavier imbalance, it faces a more serious domain shift, and more instances equipped with a large value of w are rational.

Ablation study on hyper-parameter α. The most important hyper-parameter of our method is α, which controls how much the information of the head class is transferred.
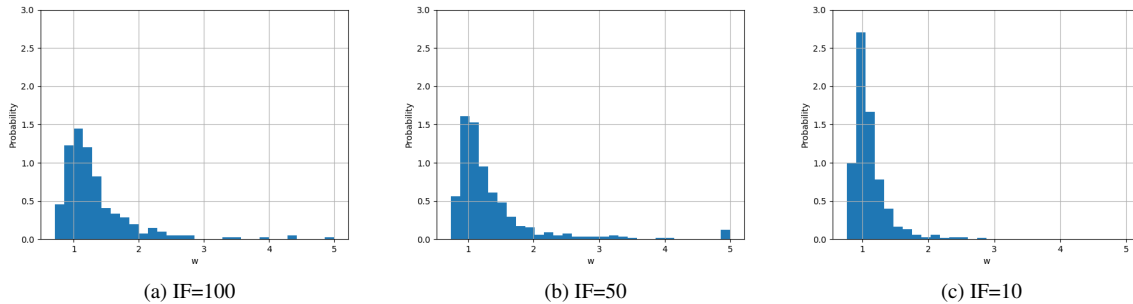
Figure 6. The distribution histograms of $w^*(\boldsymbol{x})$ with $\alpha = 0.998$. The horizontal axis represents the value of $w^*(\boldsymbol{x})$ and the vertical axis represents the probability density. The model is trained on (a) CIFAR-10-LT with IF=100, (b) CIFAR-10-LT with IF=50, and (c) CIFAR-10-LT with IF=10, respectively.
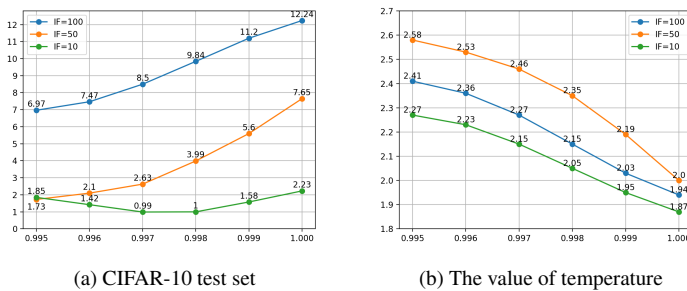

Figure 7. The blue line, orange line, and green line denote results on CIFAR-10-LT with IF=100, IF=50, and IF=10, respectively. The horizontal axis represents the hyper-parameter α. (a) The vertical axis represents the ECE value, which is tested on the original CIFAR-10 test set. (b) The vertical axis represents the temperature value, which is tested on the original CIFAR-10 test set.

Dataset	Uniform	OneHot	Ours
CIFAR-10	8.10	7.42	6.97
CIFAR-10.1	11.72	10.91	10.4
CIFAR-10.1-C	37.09	36.17	35.59
CIFAR-F	9.91	9.02	8.46

Table 5. The ECE (%) on CIFAR-10-LT with IF=100, α=0.995. Uniform denotes that we transfer knowledge from all head classes equally, i.e., all head classes give the same contribution. OneHot denotes that $\boldsymbol{s}$ is a one-hot attention vector and the weight of the most similar head class is 1 while others are 0.

Normally, a smaller value of α means we utilize more information from head classes. As shown in Fig. 7, with the growth of value α, the value of our temperature exhibits a downtrend. Note that $\alpha = 1.0$ represent traditional temperature scaling since $w(\boldsymbol{x}) = 1$ for all instances. A larger temperature can relieve the overconfidence phenomenon effectively. In addition, a heavily imbalanced dataset (IF=100) needs a larger temperature value compared with a slightly imbalanced dataset (IF=10).

As shown in Fig. 7, for the performance on the CIFAR-10 test set, different α achieve different performances. Actually, a heavily imbalanced dataset (IF=100) achieves the best performance on $\alpha = 0.995$ while a slightly imbalanced dataset (IF=10) achieves the best on $\alpha = 0.997$. This indicates that we need to utilize more information from head classes if fewer instances of tail classes are available. It is interesting that there is a downtrend and uptrend in the green curve in Fig. 7 (a). The reason is that the model is underconfident when $\alpha < 0.997$ because of the larger value of temperature. Therefore, it is not a good choice to calibrate every model with a small value of α and it is proper to apply a smaller value of α on a heavily imbalanced dataset and a larger value of α on a slightly imbalanced dataset.

5. Conclusion

In this paper, we propose a novel importance weight-based strategy to achieve post-processing calibration under long-tailed distribution. The tackled problem differs from traditional calibration tasks as the validation set follows a long-tailed distribution, while the test data distribution is balanced. We use the importance weight strategy to re-weight instances of tail classes and model the distribution of each class as a Gaussian distribution. We enhance the estimation of tail class distributions by transferring knowledge from head classes. Extensive experiments on four benchmarks demonstrate the effectiveness of our method.

Acknowledgements

References

[1] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016. 1

[2] Steffen Bickel and Tobias Scheffer. Dirichlet-enhanced spam filtering based on biased samples. *Advances in neural information processing systems*, 19, 2006. 4

[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. 2, 5

[4] Jiacheng Cheng and Nuno Vasconcelos. Calibrating deep neural networks by pairwise constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13709–13718, 2022. 2

[5] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995. 2

[6] 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. 1

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[8] Yunye Gong, Xiao Lin, Yi Yao, Thomas G Dietterich, Ajay Divakaran, and Melinda Gervasio. Confidence calibration for domain generalization under covariate shift. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8958–8967, 2021. 2, 3

[9] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017. 1, 2, 3, 6

[10] 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. 5

[11] Ramya Hebbalaguppe, Jatin Prakash, Neelabh Madan, and Chetan Arora. A stitch in time saves nine: A train-time regularizing loss for improved neural network calibration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16081–16090, 2022. 2

[12] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019. 5

[13] 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. 5

[14] Jonathan J. Hull. A database for handwritten text recognition research. *IEEE Transactions on pattern analysis and machine intelligence*, 16(5):550–554, 1994. 5

[15] 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. 1

[16] Bingyi Kang, Yu Li, Sa Xie, Zehuan Yuan, and Jiashi Feng. Exploring balanced feature spaces for representation learning. In *International Conference on Learning Representations*, 2020. 1

[17] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5

[18] Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. In *Advances in Neural Information Processing Systems*, pages 12295–12305, 2019. 1

[19] Ananya Kumar, Percy S Liang, and Tengyu Ma. Verified uncertainty calibration. *Advances in Neural Information Processing Systems*, 32, 2019. 6

[20] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2, 5

[21] Bo Liu, Haoxiang Li, Hao Kang, Gang Hua, and Nuno Vasconcelos. Gistnet: a geometric structure transfer network for long-tailed recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8209–8218, 2021. 3, 4

[22] Hong Liu, Jeff Z HaoChen, Adrien Gaidon, and Tengyu Ma. Self-supervised learning is more robust to dataset imbalance. *arXiv preprint arXiv:2110.05025*, 2021. 4

[23] 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. 2, 4

[24] 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. 1, 2, 5

[25] Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015. 1, 5

[26] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011. 5

[27] Lukas Neumann, Andrew Zisserman, and Andrea Vedaldi. Relaxed softmax: Efficient confidence auto-calibration for safe pedestrian detection. 2018. 5

[28] Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. Measuring calibration in deep learning. In *CVPR Workshops*, volume 2, 2019. 5

[29] Anusri Pampari and Stefano Ermon. Unsupervised calibration under covariate shift. *arXiv preprint arXiv:2006.16405*, 2020. 1

[30] Sangdon Park, Osbert Bastani, James Weimer, and Insup Lee. Calibrated prediction with covariate shift via unsupervised domain adaptation. In *International Conference on Artificial Intelligence and Statistics*, pages 3219–3229. PMLR, 2020. 3

[31] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999. 2

[32] Jing Qin. Inferences for case-control and semiparametric two-sample density ratio models. *Biometrika*, 85(3):619–630, 1998. 4

[33] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do cifar-10 classifiers generalize to cifar-10? *arXiv preprint arXiv:1806.00451*, 2018. 5

[34] Alfréd Rényi et al. On measures of entropy and information. In *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability*, volume 1. Berkeley, California, USA, 1961. 5

[35] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[36] Xiaoxiao Sun, Yunzhong Hou, Hongdong Li, and Liang Zheng. Label-free model evaluation with semi-structured dataset representations. *arXiv preprint arXiv:2112.00694*, 2021. 5

[37] Christian Tomani, Sebastian Gruber, Muhammed Ebrar Erdem, Daniel Cremers, and Florian Buettner. Post-hoc uncertainty calibration for domain drift scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10124–10132, 2021. 1, 3

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[39] Yoav Wald, Amir Feder, Daniel Greenfeld, and Uri Shalit. On calibration and out-of-domain generalization. *Advances in neural information processing systems*, 34:2215–2227, 2021. 3

[40] Ximei Wang, Mingsheng Long, Jianmin Wang, and Michael Jordan. Transferable calibration with lower bias and variance in domain adaptation. *Advances in Neural Information Processing Systems*, 33:19212–19223, 2020. 1, 3, 4

[41] Jonathan Wenger, Hedvig Kjellström, and Rudolph Triebel. Non-parametric calibration for classification. In *International Conference on Artificial Intelligence and Statistics*, pages 178–190. PMLR, 2020. 6

[42] Zhengzhuo Xu, Zenghao Chai, and Chun Yuan. Towards calibrated model for long-tailed visual recognition from prior perspective. *Advances in Neural Information Processing Systems*, 34:7139–7152, 2021. 3

[43] Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 694–699, 2002. 3, 6

[44] Jize Zhang, Bhavya Kailkhura, and T Yong-Jin Han. Mix-n-match: Ensemble and compositional methods for uncertainty calibration in deep learning. In *International conference on machine learning*, pages 11117–11128. PMLR, 2020. 3, 6

[45] Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16489–16498, 2021. 3