Enhancing Zero-Shot Vision Models by Label-Free Prompt Distribution Learning and Bias Correcting

Xingyu Zhu[1][*]   Beier Zhu[2][*]   Yi Tan[1]   Shuo Wang[1][†]   Yanbin Hao[1]   Hanwang Zhang[2]

[1]University of Science and Technology of China
[2]Nanyang Technological University
xingyuzhu@mail.ustc.edu.cn,   shuowang.edu@gmail.com

Abstract

Vision-language models, such as CLIP, have shown impressive generalization capacities when using appropriate text descriptions. While optimizing prompts on downstream labeled data has proven effective in improving performance, these methods entail labor costs for annotations and are limited by their quality. Additionally, since CLIP is pre-trained on highly imbalanced Web-scale data, it suffers from inherent label bias that leads to suboptimal performance. To tackle the above challenges, we propose a label-**F**ree p**ro**mpt distribution **l**earning and b**i**as **c**orrection framework, dubbed as **Frolic**, which boosts zero-shot performance without the need for labeled data. Specifically, our Frolic learns distributions over prompt prototypes to capture diverse visual representations and adaptively fuses these with the original CLIP through confidence matching. This fused model is further enhanced by correcting label bias via a label-free logit adjustment. Notably, our method is not only training-free but also circumvents the necessity for hyper-parameter tuning. Extensive experimental results across 16 datasets demonstrate the efficacy of our approach, particularly outperforming the state-of-the-art by an average of 2.6% on 10 datasets with CLIP ViT-B/16 and achieving an average margin of 1.5% on ImageNet and its five distribution shifts with CLIP ViT-B/16. Codes are available in https://github.com/zhuhsingyuu/Frolic.

1   Introduction

Vision-language models (VLMs), such as CLIP [29], which are pre-trained on large-scale datasets using contrastive loss, effectively align visual and textual representations within a shared feature space. This capability enables the zero-shot inference on downstream tasks through prompting and achieves remarkable performance. For example, using a selection of 80 hand-crafted prompts, a zero-shot CLIP ViT-B/16 achieves an accuracy of 68.7%, and with prompts generated by language models [27], the accuracy increases to 69.9%.

The success of zero-shot capabilities heavily relies on the appropriate text descriptions of the classes, which has gained research interest in improving prompts. Recent studies propose learning prompts from a small set of labeled images in the downstream data [46, 45, 47]. Among these studies, Lu *et al.* [18] and Wang *et al.* [38] have found that learning the distribution of diverse prompts, which better captures the variance in visual representations, leads to improved performance. Although these methods have achieved significant improvements, they still depend on artificial prior knowledge for labeling downstream data and are limited by the quality of manual annotations, which may restrict the scalability of the original model.

---

[*]Equal contributions
[†]Corresponding author

38th Conference on Neural Information Processing Systems (NeurIPS 2024).
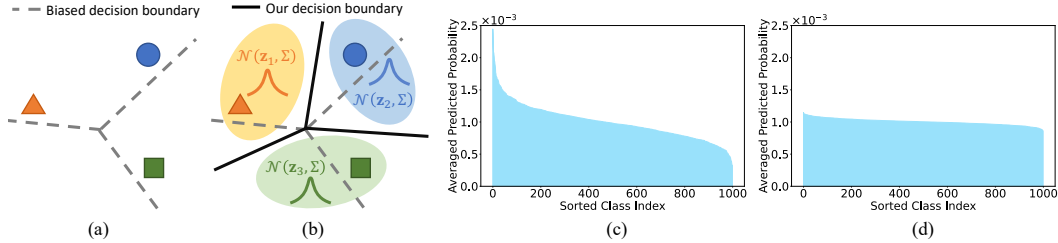
Figure 1: Illustration of prompt distribution learning and label bias correction on ImageNet using CLIP ViT-B/16. (a) Existing zero-shot models [1, 27]. (b) Our prompt distribution learning (c) Average probability prediction of original CLIP. (d) Average probability prediction of our Frolic.

Another significant approach to enhancing zero-shot performance involves correcting the label bias inherent in skewed web-scale pre-training data [1, 25, 49]. This bias leads to highly imbalanced predictions and suboptimal performance. As illustrated in Figure 1(c), the average predicted probability on ImageNet using ViT-B/16 reveals an imbalanced distribution: the highest class probability exceeds 0.002, whereas the lowest is below 0.0005. Existing methods correct this bias by allowing access to a portion of the pre-training data [1, 25], or by using labeled downstream data [49]. However, the pre-training data is often inaccessible due to privacy or copyright concerns, and debiasing without labeled data is challenging.

In this paper, we introduce a label-**F**ree p**ro**mpt distribution **l**earning and b**i**as **c**orrection framework, dubbed as **Frolic**, which eliminates the need for data annotations to enhance zero-shot performance. First, unlike previous methods [1, 27, 46, 39, 43], which use a single class prototype for each class to define the decision boundary (as shown in Figure 1(a)), our approach employs Gaussian distributions to model the varied visual representations of text prototypes, as illustrated in Figure 1(b). It is worth noting that estimating such a distribution is non-trivial, since classical maximum likelihood estimation requires the annotation of each sample. Fortunately, we demonstrate that it is possible to infer distribution for each class directly from the first and second moments of the marginal distribution of downstream data without label information. Second, to prevent the use of pre-training data or labeled samples in downstream tasks, we develop a bias estimation mechanism, which transitions the sampling process from the pre-training data distribution to a class-conditional sampling from downstream distribution. By incorporating the estimated label bias into zero-shot models, we can achieve a balanced prediction, as illustrated in Figure 1 (d). Furthermore, we explore the possibility of combining the original CLIP predictions with those from the Gaussian-based models to enhance zero-shot performance. To this end, we have developed a confidence-matching technique that dynamically balances the contributions of the two models, eliminating the need for hyperparameter tuning. Notably, our framework is training-free, which enhances both flexibility and ease of implementation.

The main contributions of this work are:

- We enhance zero-shot performance by estimating a distribution over prompt prototypes to capture the variance in visual appearances. We demonstrate that this process can be implemented entirely without labels.

- We propose a confidence matching technique that fuses the original CLIP model with a Gaussian distribution-based model to further enhance zero-shot performance. This process eliminates the need for hyper-parameter searching, in stark contrast to previous studies.

- We develop an unsupervised method to correct pre-training label bias. Unlike existing methods that require access to pre-training data, our Proposition 2 suggests that we can avoid sampling from the pre-training distribution for estimating and correcting this bias. Instead, our method utilizes only downstream images.

- We demonstrate the effectiveness of our proposed method Frolic by conducting experiments across 16 datasets, which has a consistent and significant improvement over existing baselines. For example, our method surpasses the state-of-the-art zero-shot models by a margin of 2.6% on average with CLIP ViT-B/16.

2

2 Related Works

Zero-shot vision models. Vision models pre-trained with auxiliary language supervision, such as CLIP [29] and OpenCLIP [6], facilitate zero-shot inference through prompting. Enhancing zero-shot performance has gained increasing research interest: (1) One approach involves prompt engineering, which includes designing hand-crafted prompts based on human priors [29] or automatically generating prompts via language models [35]. (2) Another promising approach seeks to improve classifiers, *e.g.*, ZPE [1] scores the importance of candidate prompts for prompt ensembling. InMaP [28] reduces the modality gap between vision and text. Several studies [32, 31] optimize prompt at test time by encouraging consistent predictions across augmented samples. Our work aims to enhance zero-shot models by learning the prompt distribution and mitigating the pre-training label bias.

Prompt distribution learning. Automatically learning prompts from downstream data has shown potential in improving zero-shot models [46, 45, 47]. These methods typically optimize prompts via minimizing the classification loss on the target task. However, as pointed out in Lu *et al.* [18], learning prototype prompts overlook the diversity of visual representations. To this end, they estimate a distribution over the prompts to capture the variance of visual representations. Recently, Wang *et al.* [38] propose training-free prompt distribution learning to improve efficiency. Contrary to existing methods [18] that estimate distributions through supervised approaches, our method circumvents the necessity for labels by inferring the variance of distributions from the statistics of unlabeled data.

Correcting label bias. Label bias generally occurs in the presence of skewed or imbalanced training data. In response to this challenge, Logit Adjustment (LA) [34, 14, 21, 49] has emerged as a prominent technique in long-tailed learning, specifically designed to adjust the decision boundary of classifiers to mitigate label bias. Menon *et al.* [21] derives the theoretically optimal adjustment for logits. Zhu *et al.* [49] extents LA to fine-tune zero-shot models by removing the pre-trained label bias. Unlike approaches that rely on the label distribution of the training set [34, 14, 21, 48] or the labels of fine-tuning data [49], our method adjusts the logits using unlabeled test data.

3 Methods

In this section, we present our prompt distribution learning, adaptive fusion, and logit adjustment techniques for adapting zero-shot models. Without loss of generality, we adopt CLIP [29] as our zero-shot model. To begin with, we emphasize three advantages of our framework:

Training-free: Our Frolic is training-free without optimizing the backbone of the zero-shot models, enhancing both flexibility and ease of implementation.

Label-free: Our method Frolic requires no external labeled data, making it suitable for zero-shot scenarios.

No hyper-parameters searching: Our method Frolic eliminates hyper-parameter tuning on validation datasets, in stark contrast to [38, 44]

3.1 Setup

The zero-shot model consists of a visual encoder $\Phi_{\mathsf{v}}(\cdot)$ and a text encoder $\Phi_{\mathsf{t}}(\cdot)$. Given a set of unlabeled image data $\{x_i\}_{i=1}^N$ and the unique text set of the class description $\{z_j\}_{j=1}^K$, their visual and text representation can be computed as:

$$\mathbf{x}_i = \Phi_{\mathsf{v}}(x_i); \quad \mathbf{z}_j = \Phi_{\mathsf{t}}(z_j), \tag{1}$$

where $\mathbf{x}_i$ and $\mathbf{z}_j$ share the same dimension ($\mathbf{x}, \mathbf{z} \in \mathbb{R}^d$). N is the sample size and K is the class size. $\mathbf{z}_j$ can be considered as the prototype for class j. With an image $\mathbf{x}$ and all prototypes $\{\mathbf{z}_j\}_{j=1}^K$, zero-shot CLIP predicts the label as:

$$y = \underset{j}{\operatorname{argmax}} \, f_{\mathsf{c}}(\mathbf{x})_j = \underset{j}{\operatorname{argmax}} \, \mathbf{z}_j^\top \mathbf{x}, \tag{2}$$

where $f_{\mathsf{c}}(\mathbf{x})_j = \mathbf{z}_j^\top \mathbf{x}$ is the score for class j.

3

3.2 Label-Free Prompt Distribution Learning

In order to express the diverse visual variations, our approach aims to learn the distribution of the class prototypes. Previous studies [18, 38] show that the Gaussian distribution is effective to model the distribution of the CLIP features and achieves impressive improvement. However, these methods require *extra labeled training data*, which is not applicable to our zero-shot setting.

Specifically, we follow [38] to assume $\mathcal{N}(\mathbf{z}_{1:K}, \Sigma)$ with identical covariance is the underlying distribution. In classical maximum likelihood estimation [3], the shared covariance Σ is computed by averaging the empirical covariances of K classes: $\hat{\Sigma} = \frac{1}{K} \sum_j \hat{\Sigma}_j$, where $\hat{\Sigma}_j = \frac{1}{|\mathcal{C}_j|-1} \sum_{\mathbf{x} \in \mathcal{C}_j} (\mathbf{x} - \mathbf{z}_j)(\mathbf{x} - \mathbf{z}_j)^\top$. Here, one need the label information of each image to compute $\hat{\Sigma}_j$. Fortunately, to avoid using label information, we can infer Σ directly from the expectation and the second order moment of the marginal distribution $\mathbb{P}(\mathbf{x})$.[1] Using a Gaussian mixture model with the priors $\{\pi_j\}_{j=1}^K$, $\mathbb{P}(\mathbf{x})$ is given by:

$$\mathbb{P}(\mathbf{x}) = \sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}; \mathbf{z}_j, \Sigma), \quad \mathcal{N}(\mathbf{x}; \mathbf{z}_j, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\{-\frac{1}{2}(\mathbf{x} - \mathbf{z}_j)^\top \Sigma^{-1} (\mathbf{x} - \mathbf{z}_j)\} \quad (3)$$

Denote the second moment of $\mathbf{x}$ as M, we have (proof in Section A.1):

$$M = \Sigma + \sum_j \pi_j \mathbf{z}_j \mathbf{z}_j^\top. \quad (4)$$

Denote $\boldsymbol{\pi} = [\pi_1, .., \pi_K]^\top$, $Z = [\mathbf{z}_1, .., \mathbf{z}_K]^\top$, and the expectation of $\mathbf{x}$ as $\boldsymbol{\mu}$, the prior over the unlabeled data distribution can be estimated by (proof in Section A.2):

$$\boldsymbol{\pi} = Z^{-1} \boldsymbol{\mu} \quad (5)$$

We estimate the expectation and the second order moment as $\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i$ and $\hat{M} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^\top$, which are unbiased and consistent. In practice, given that test benchmarks are generally class-balanced, we use a uniform prior over the data distribution, *i.e.*, $\pi_j = \frac{1}{K}$. Combining with Eq. (4), the estimated shared covariance $\hat{\Sigma}$ can be written as:

$$\hat{\Sigma} = \hat{M} - \frac{1}{K} \sum_j \mathbf{z}_j \mathbf{z}_j^\top. \quad (6)$$

Let $\mathbf{w}_j = \hat{\Sigma}^{-1} \mathbf{z}_j$ and $b_j = -\frac{1}{2} \mathbf{z}_j^\top \mathbf{w}_j$, the Gaussian discriminant analysis predicts the label for an image $\mathbf{x}$ as follows (proof in Section A.3):

$$y = \operatorname*{argmax}_j f_\mathbf{g}(\mathbf{x})_j = \operatorname*{argmax}_j \mathbf{w}_j^\top \mathbf{x} + b_j \quad (7)$$

where $f_\mathbf{g}(\mathbf{x})_j = \mathbf{w}_j^\top \mathbf{x} + b_j$ is the score for class j.

3.3 Prediction Fusion via Adaptive Calibration.

As a rule of thumb, combining the zero-shot predictions with the ones from the learned model can further improve performance for CLIP adaptations [44, 35, 40, 47, 38, 50]. Previous studies commonly employ a mixing coefficient, α, to balance the contributions of two models, *e.g.*, $f(\mathbf{x}) = f_\mathbf{c}(\mathbf{x}) + \alpha f_\mathbf{g}(\mathbf{x})$. Typically, this hyper-parameter α is optimized on labeled data to maximize accuracy. However, in our context, labels are unavailable, it is not possible to search for the optimal value of α. It is imperative to develop a mechanism that balances the prediction fusion without relying on the label.
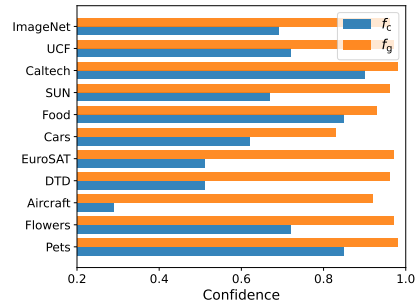
Figure 2: Comparison of confidence.

---

[1] Despite that the modality gap exists between the text and vision space of CLIP models, we can use the unsupervised method from InMaP [28] to effectively align the two modalities.

4

Algorithm 1 Pipeline of our Frolic

1: **Given**: Unlabeled data $\{\mathbf{x}_i\}_{i=1}^N$,
   prototypes $\{\mathbf{z}_j\}_{j=1}^K$ and τ_{c}
2: Build $f_{\mathsf{c}}(\mathbf{x})_y = \mathbf{z}_y^\top \mathbf{x}$
3: Compute $\hat{\Sigma} = \hat{M} - \frac{1}{K}\sum_j \mathbf{z}_j \mathbf{z}_j^\top$
   where $\hat{M} = \frac{1}{N}\sum_i \mathbf{x}_i \mathbf{x}_i^\top$
4: Compute $\mathbf{w}_j = \hat{\Sigma}^{-1}\mathbf{z}_j$, $b_j = -\frac{1}{2}\mathbf{z}_j^\top \mathbf{w}_j$
5: Build $f_{\mathsf{g}}(\mathbf{x})_y = \mathbf{w}_y^\top \mathbf{x} + b_y$
6: Search τ_{g} by Eq. (9)
7: Build $f_{\mathsf{f}}(\mathbf{x}) = f_{\mathsf{g}}(\mathbf{x})/\tau_{\mathsf{g}} + f_{\mathsf{c}}(\mathbf{x})/\tau_{\mathsf{c}}$
8: Compute $\hat{\boldsymbol{\beta}}$ by Algorithm 2
9: **return** $f_{\mathsf{d}}(\mathbf{x}) = f_{\mathsf{f}}(\mathbf{x}) - \ln\hat{\boldsymbol{\beta}}$

Algorithm 2 Estimation of $\boldsymbol{\beta}$

1: **Given**: Unlabeled data $\{\mathbf{x}_i\}_{i=1}^N$,
   predictor $f_{\mathsf{f}}(\cdot)$ and tolerance ϵ.
2: Initialize $\boldsymbol{\beta}^0$, f_{d}^0 and S^0 by Eq. (13)
3: $t = 0$
4: **repeat**
5:     $t = t + 1$
6:     Update $\boldsymbol{\beta}^t$ by solving $(S^{t-1} - I)\boldsymbol{\beta}^t = \mathbf{0}$
7:     Update $f_{\mathsf{d}}^t = f_{\mathsf{f}} - \boldsymbol{\beta}^t$
8:     Update S^t from $\mathbf{s}_j^t = \frac{1}{|\mathcal{C}_j^t|}\sum_{\mathbf{x}\in\mathcal{C}_j^t} s(\mathbf{x})$,
       where $\mathcal{C}_j^t$ is assigned by f_{d}^t
9: **until** $\|\boldsymbol{\beta}^t - \boldsymbol{\beta}^{t-1}\|_1 < \epsilon$
10: **return** $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}^t$

The key in our prediction fusion lies in aligning the average confidence of the two models. Formally, the average confidence over the dataset $\{\mathbf{x}_i\}_{i=1}^N$ scaled by a temperature τ is given by the average of the model's probability for its prediction:

$$\text{conf}(f, \tau) = \frac{1}{N}\sum_{i=1}^N \max_j \text{softmax}(f(\mathbf{x}_i)/\tau)_j. \tag{8}$$

Ideally, a model's average confidence should reflect the predicted accuracy, which is called a well-calibrated model. Suppose we have the oracle well-calibrated models, denoted by $f_{\mathsf{c}}'(\cdot)$ and $f_{\mathsf{g}}'(\cdot)$, Kumart *et al.* [17] prove that the optimal strategy is to fuse the two predictions equally, *i.e.*, $f_{\mathsf{f}}(\mathbf{x}) = f_{\mathsf{c}}'(\mathbf{x}) + f_{\mathsf{g}}'(\mathbf{x})$. However, as shown in Figure 2, f_{g} is much overconfident than f_{c}. Let $f_{\mathsf{g}}(\mathbf{x}) = Cf_{\mathsf{g}}'(\mathbf{x})$ for large $C \in \mathbb{R}^+$ (an overconfident model magnifies its logits) and suppose $f_{\mathsf{c}}(\mathbf{x}) \approx f_{\mathsf{c}}'(\mathbf{x})$. The fused predictions are given by $f_{\mathsf{f}}(\mathbf{x}) = Cf_{\mathsf{g}}'(\mathbf{x}) + f_{\mathsf{c}}'(\mathbf{x})$. For very large C, $f_{\mathsf{f}}(\mathbf{x})$ and $f_{\mathsf{g}}(\mathbf{x})$ have the same predictions, *i.e.*, $f_{\mathsf{f}}(\mathbf{x})$ is biased towards the $f_{\mathsf{g}}(\mathbf{x})$. As we do not have the label to compute accuracy, we cannot apply classical calibration methods [19, 10] to calibrate $f_{\mathsf{g}}(\mathbf{x})$ and $f_{\mathsf{c}}(\mathbf{x})$. As our desideratum is to automatically balance the contribution of the two models, we can optimize τ_{g} to make the confidence of f_{g} to match up the one of f_{c}, which circumvent the need of labels:

$$\tau_{\mathsf{g}} = \underset{\tau_{\mathsf{g}}}{\text{argmin}} \, |\text{conf}(f_{\mathsf{g}}, \tau_{\mathsf{g}}) - \text{conf}(f_{\mathsf{c}}, \tau_{\mathsf{c}})| \tag{9}$$

Specifically, we implement this by binary search, as the confidence monotonically decreases as the temperature increases. $\tau_{\mathsf{c}} = 0.01$ is fixed and learned by CLIP. The fused logits are given by:

$$f_{\mathsf{f}}(\mathbf{x}) = f_{\mathsf{g}}(\mathbf{x})/\tau_{\mathsf{g}} + f_{\mathsf{c}}(\mathbf{x})/\tau_{\mathsf{c}} \tag{10}$$

3.4 Correcting Pre-training Label Bias via Label-Free Logit Adjustment

Pre-training datasets typically exhibit a long-tailed concept distribution, leading to biased performance in zero-shot models [49, 25, 5, 1]. This bias occurs because zero-shot models reflect the posterior probability $\mathbb{P}(y|\mathbf{x})$ derived from the pre-training distribution. According to Bayes' rule, this posterior probability is influenced by the pre-training label distribution $\mathbb{P}(y)$, as $\mathbb{P}(y|\mathbf{x}) \propto \mathbb{P}(\mathbf{x}|y)\mathbb{P}(y)$. If the prior probability of class j is significantly larger than that of other classes (*e.g.*, $\mathbb{P}(j) \gg \mathbb{P}(i)$, $\forall i \in [K], i \neq j$), the predictions will be biased toward class j.

Prior research [21, 14] has identified a theoretical optimal solution to address this label bias: let β_y denote the prior probability of class y, *i.e.*, $\beta_y = \mathbb{P}(y)$. The debiased logit of $f_{\mathsf{f}}(\mathbf{x})$ for class y should be (proof in Section A.4):

$$f_{\mathsf{d}}(\mathbf{x})_y = f_{\mathsf{f}}(\mathbf{x})_y - \ln\beta_y. \tag{11}$$

Previous methods estimate β either by accessing the pre-training data [25, 1] or counteract the influence of the prior by optimizing on labeled downstream data [49]. However, these approaches are often impractical due to inaccessible pre-training labels due to privacy or copyright concerns or the

5

necessity for labeled downstream data. In this work, we address label bias using only the unlabeled downstream data $\{\mathbf{x}_i\}_{i=1}^N$.

Let $s(\mathbf{x}) = \text{softmax}(f_\mathsf{f}(\mathbf{x}))$ represent the softmax outputs of $f_\mathsf{f}(\mathbf{x})$, where $s(\mathbf{x})_y = \hat{\mathbb{P}}(y|\mathbf{x})$ is the predicted probability for class y. Define $\mathbf{s}_j = \mathbb{E}_\mathbf{x}[s(\mathbf{x})|Y = j]$ as the expected posterior probability over the image distribution of class j, and let $S = [\mathbf{s}_1, ..., \mathbf{s}_K] \in \mathbb{R}^{K \times K}$. We prove that the pre-training label prior $\boldsymbol{\beta} = [\beta_1, ..., \beta_K]^\top \in \mathbb{R}^K$ must satisfy the following linear equation system:

$$(S - I)\boldsymbol{\beta} = \mathbf{0}. \tag{12}$$

Remark. The key point in Eq. (12) is that we avoid sampling from the pre-training data distribution; instead, we sample from $\mathbb{P}(\mathbf{x}|y)$, which is available from the downstream data. We provide the proof in Section A.5 and the numerical power solver for $\boldsymbol{\beta}$ in Section A.6.

We iteratively refine the estimation of S and solve for $\boldsymbol{\beta}$ using updated pseudo-labels generated by $f_\mathsf{d}(\mathbf{x})$. As $f_\mathsf{d}(\mathbf{x})$ becomes more precise, it yields more accurate pseudo-labels for $\mathbf{x}$, which in turn enhances the accuracy of our estimation of $\boldsymbol{\beta}$. Specifically, we initialize

$$\boldsymbol{\beta}^0 = [1/K, ..., 1/K]^\top, \ f_\mathsf{d}^0 = f_\mathsf{f}, \ \mathbf{s}_j^0 = \frac{1}{|\mathcal{C}_j^0|}\sum_{\mathbf{x}\in\mathcal{C}_j^0} s(\mathbf{x}), \text{ and } S^0 = [\mathbf{s}_1^0, ..., \mathbf{s}_K^0] \tag{13}$$

where $\mathbf{x} \in \mathcal{C}_j^0$ if $\mathbf{x}$ is classified as j by $f_\mathsf{d}^0(\mathbf{x})$. We proceed by solving for $\boldsymbol{\beta}^1$ using Eq. (12), refining $f_\mathsf{d}^1(\mathbf{x})$ using Eq 11 and reassign the pseudo label using $f_\mathsf{d}^1(\mathbf{x})$ to estimate the updated $\mathbf{s}_j^1$. This process is repeated t times until the relative change in $\boldsymbol{\beta}$ satisfies the convergence criterion:

$$\frac{\|\boldsymbol{\beta}^t - \boldsymbol{\beta}^{t-1}\|_1}{\|\boldsymbol{\beta}^{t-1}\|_1} = \|\boldsymbol{\beta}^t - \boldsymbol{\beta}^{t-1}\|_1 < \epsilon, \quad \|\boldsymbol{\beta}^{t-1}\| = 1 \text{ by definition} \tag{14}$$

where ϵ is a predefined threshold for relative error tolerance. We summarize the algorithm for solving $\boldsymbol{\beta}$ in Algorithm 2 and provide the overall pipeline in Algorithm 1.

Discussion: Comparison with Other Prior Estimation Methods. We compare existing methods for estimating pre-training label priors and demonstrate their in-applicability or flaws in our setting.

(1) *Explicit method*: the explicit method directly measures the frequency of each class in pre-training data, *e.g.*, $\beta_y = \frac{N_y}{N}$, where N_y is the sample size for class y and N is the total sample size. Most long-tail learning algorithms, *e.g.*, LA and PC [21, 14], are based on this method because they can access the training data. However, estimating such frequency is complex due to the free-form texts, as opposed to a pre-defined label set. In addition, the pre-training dataset is often inaccessible, making the method inapplicable in our case.

(2) *Implicit method*: [1, 25] allow access to a portion of the pre-training data $\mathcal{D}_\mathsf{pt}$ and use the law of total probability to estimate the prior:

$$\beta_y = \mathbb{P}(y) = \int_\mathbf{x} \mathbb{P}_\mathsf{pt}(\mathbf{x})\mathbb{P}(y|\mathbf{x})d\mathbf{x} = \mathbb{E}_{\mathbf{x}\sim\mathbb{P}_\mathsf{pt}(\mathbf{x})}[\mathbb{P}_\mathsf{pt}(y|\mathbf{x})] \approx \frac{1}{|\mathcal{D}_\mathsf{pt}|}\sum_{\mathbf{x}\in\mathcal{D}_\mathsf{pt}} \hat{\mathbb{P}}_\mathsf{pt}(y|\mathbf{x}) \tag{15}$$

where $\hat{\mathbb{P}}_\mathsf{pt}(y|\mathbf{x})$ denotes the zero-shot model. However, in our setting, we do have access to the pre-training data or a portion of it. Wang *et al.* [37] replace the pre-training data $\mathcal{D}_\mathsf{pt}$ with the downstream data $\mathcal{D}_\mathsf{ds}$ in Eq. (15) to debias. However, this method neglects the distribution discrepancies between the pre-training and downstream data. In Section 4.3, we show that our debiasing significantly outperforms this implicit method.

(3) *TDE* [34]: Tang *et al.* [34] debias by removing features along a global direction, retaining only those orthogonal to it. Specifically, the global feature is estimated by $\bar{\mathbf{x}} = \frac{1}{|\mathcal{D}_\mathsf{pt}|}\sum_{\mathbf{x}\in\mathcal{D}_\mathsf{pt}} \mathbf{x}$. Given a test sample $\mathbf{x}$, TDE decomposes it into parallel and orthogonal directions to $\bar{\mathbf{x}}$: $\mathbf{x} = \mathbf{x}_\| + \mathbf{x}_\perp$. Then, only the orthogonal component is used for classification: $\hat{\mathbb{P}}_\mathsf{pt}(y|\mathbf{x}_\perp)$. While TDE does not require labels for the samples, we cannot apply it because it requires sampling from the pre-training data. In Section 4.3, we replace $\mathcal{D}_\mathsf{pt}$ with downstream data $\mathcal{D}_\mathsf{ds}$ and demonstrate its inferior performance.

(4) *GLA* [49]: Zhu *et al.* [49] propose to estimate the pre-training prior from the downstream data using the Bayes optimal criterion. The pre-training prior $\boldsymbol{\beta}$ is solved by optimizing:

$$\boldsymbol{\beta} = \arg\min_{\boldsymbol{\beta}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_\mathsf{ds}}[\ell_\mathsf{ce}(f_\mathsf{pt}(\mathbf{x}) - \ln\boldsymbol{\beta}, y)], \tag{16}$$

6

Table 1: Comparison of accuracy (%) on 10 datasets for CLIP ViT-B/16 and ViT-L/14.

	Method	Pets	Flowers	Aircraft	DTD	EuroSAT	Cars	Food	SUN	Caltech	UCF	Average
ViT-B/16	CLIP [29]	88.9	70.4	24.8	44.3	47.7	65.2	86.1	62.5	92.9	66.7	64.9
	TPT [32]	87.7	68.9	24.7	47.7	42.4	66.8	84.6	65.5	94.1	68.0	65.0
	PromptAlign [31]	90.7	72.3	24.8	47.2	47.8	68.5	86.6	67.5	94.0	69.4	66.8
	SuS-X-SD [35]	90.5	73.8	28.6	54.5	57.4	66.1	86.0	67.7	93.6	66.5	68.4
	TDA [15]	88.6	71.4	23.9	47.4	58.0	67.2	86.1	67.6	94.2	70.6	67.5
	GPT4-Prompt [41]	91.0	74.5	28.0	48.5	48.8	66.8	86.3	65.5	94.6	72.0	67.6
	CuPL-CLIP [27]	92.0	73.2	27.7	54.3	52.7	66.4	86.2	68.5	94.6	70.7	68.6
	Frolic	**92.9**	**74.8**	**31.5**	**56.1**	**58.5**	**69.1**	**87.2**	**70.8**	**95.2**	**75.2**	**71.1**
	InMaP [28]	92.9	71.8	28.4	48.0	64.1	70.6	87.7	70.5	93.1	74.0	70.1
	+ Frolic	**93.6**	**74.3**	**31.8**	**58.0**	**65.3**	**71.7**	**88.2**	**72.8**	**95.4**	**75.9**	**72.7**
ViT-L/14	CLIP [29]	93.5	79.3	32.4	53.0	58.0	76.8	91.0	67.5	94.8	74.2	72.0
	TPT [32]	93.6	76.2	31.9	55.2	51.8	77.7	88.9	70.2	95.5	74.9	71.5
	TDA [15]	93.5	80.5	34.7	56.7	64.1	78.3	90.9	71.5	95.9	76.6	74.2
	GPT4-Prompt [41]	94.1	81.5	36.3	54.8	54.1	77.9	91.4	70.3	96.2	80.6	73.7
	CuPL-CLIP [27]	94.3	79.8	35.5	62.7	61.2	78.0	91.3	72.4	96.7	75.9	74.7
	Frolic	**94.9**	**82.4**	**40.0**	**64.1**	**66.2**	**80.8**	**91.8**	**74.5**	**97.2**	**80.0**	**77.1**
	InMaP [28]	95.2	80.7	37.6	60.2	70.6	82.5	92.2	75.0	94.9	80.4	76.9
	+ Frolic	**95.4**	**81.8**	**42.1**	**66.9**	**71.0**	**83.5**	**92.4**	**77.3**	**97.3**	**82.2**	**78.9**

where ℓ_{ce} is the cross-entropy loss and $f_{\text{pt}}(\mathbf{x})$ is the logit of the zero-shot model. While this method circumvents the need for pre-training data access, it is inapplicable because it requires labels for each downstream sample.

4 Experiments

4.1 Setup

Datasets. We conduct experiments on 16 image classification benchmarks, covering diverse range categories including generic objects (ImageNet [8], Caltech [9]), scenes (SUN [42]), textures (DTD [7]), satellite images (EuroSAT [11]), actions (UCF [33]) and fine-grained categories (Pets [26], Cars [16], Flowers [23], Food [4], Aircraft [20]). Additionally, we evaluate on five ImageNet distribution shifted datasets [8]: ImageNetV2 (IN-V2) [30], ImageNet-Sketch (IN-Sketch) [36], ImageNet-A (IN-A) [13], ImageNet-R (IN-R) [12] and ObjectNet [2].

Implementation details. We adopt CLIP [29] ViT-B/16 and ViT-L/14 as our pre-trained models. The default model for ablation studies is CLIP ViT-B/16. We use the same text descriptions as SuS-X [35] and CuPL [27], and adhere to the InMaP [28] settings to include all test images. $\tau_{\text{c}} = 0.01$ is provided by CLIP. ϵ in Algorithm 2 is set to 0.01. All experiments are conducted on a single NVIDIA 3090 GPU if not specified. Note that our algorithm *does not require* any hyper-parameter searching.

4.2 Main Results

We compare our method with several state-of-art methods, including CLIP [29], TPT [32], PromptAlign [31], SuS-X-DS [35], TDA [15], GPT4-Prompt [41], CuPL-CLIP [27], and InMaP [28]. Both TPT and TDA utilize a stream of unlabeled test images. For TPT, TDA and InMaP, we produce the results of ViT-L/14 by executing the official released code and maintaining the same hyper-parameters.

Results on 10 datasets. In Table 1, we summarize the accuracy across all datasets, excluding ImageNet and its shifts (denoted as 10-datasets). Our method consistently shows superior performance across the datasets and backbones, significantly surpassing GPT4-Prompt, which is known for generating high-quality prompts. By integrating our method with InMaP, our Frolic achieves the highest performance, with an average improvement of 2.6% with ViT-B/16 and 2.0% with ViT-L/14.

7

Table 2: Comparison of accuracy (%) on ImageNet and its variants for CLIP ViT-B/16 and ViT-L/14.

	Method	IN	IN-V2	IN-Sketch	IN-A	IN-R	ObjectNet	Average
ViT-B/16	CLIP [29]	68.7	62.2	48.3	50.6	77.7	53.5	60.1
	TPT [32]	68.9	63.4	47.9	54.7	77.0	55.1	61.1
	TDA[15]	69.5	64.6	50.5	60.1	80.2	55.1	63.3
	GPT4-Prompt [41]	68.7	62.3	48.2	50.6	77.8	53.7	60.2
	CuPL-CLIP [27]	69.9	64.4	49.4	59.7	79.5	53.7	62.7
	Frolic	**70.9**	**64.7**	**53.3**	**60.4**	**80.7**	**56.6**	**64.4**
	InMaP [28]	72.5	62.3	49.4	52.2	79.2	54.5	61.6
	+ Frolic	**73.3**	**63.8**	**52.9**	**52.8**	**79.6**	**56.4**	**63.1**
ViT-L/14	CLIP [29]	75.9	70.2	59.7	70.9	87.9	65.5	71.6
	TPT [32]	75.5	70.0	59.8	74.7	87.9	68.0	72.6
	TDA[15]	76.3	71.5	61.3	77.9	89.8	67.0	73.9
	GPT4-Prompt [41]	75.3	70.3	59.9	71.2	87.8	65.7	71.7
	CuPL-CLIP [27]	76.2	71.9	60.7	77.9	89.6	65.7	73.6
	Frolic	**77.4**	**72.5**	**63.1**	**78.9**	**90.3**	**68.7**	**75.1**
	InMaP [28]	79.3	72.1	65.1	62.5	84.8	71.0	72.4
	+ Frolic	**79.7**	**73.1**	**65.7**	**64.0**	**85.9**	**71.7**	**73.3**

Table 3: Accuracy (%) of different models on 10-datasets, ImageNet and its five variant datasets.

	Model	ViT-B/16			ViT-L/14		
		10-datasets	ImageNet	IN-Variants	10-datasets	ImageNet	IN-Variants
(1)	f_c	65.1	68.7	58.5	72.0	75.9	72.3
(2)	$f_c - \ln\boldsymbol{\beta}$	68.4	69.7	61.2	75.1	76.2	73.4
(3)	f_g	68.8	69.8	61.3	74.7	76.0	73.1
(4)	$f_c + f_g$	66.3	68.9	59.1	72.5	76.1	72.4
(5)	$f_f = f_c/\tau_c + f_g/\tau_g$	70.4	69.8	61.9	75.5	76.9	73.9
(6)	$f_d = f_f - \ln\boldsymbol{\beta}$	**71.1**	**70.9**	**63.1**	**77.2**	**77.4**	**77.4**

Results on ImageNet and associated five shifts. In Table 2, our Frolic again surpasses the comparison methods, achieving the average accuracy of 64.4% and 75.1% with ViT-B/16 and ViT-L/14, respectively. Additionally, we observe improvements on the distribution shift datasets: IN-V2, IN-Sketch, IN-A, and IN-R with ViT-B/16, and on IN-A and IN-R with ViT-L/14, when our Frolic is combined with InMaP. However, these results still lag behind the original performance of our Frolic. This discrepancy may stem from the hyper-parameters in InMaP being optimized specifically for ImageNet; applying them unchanged to its shifted datasets could lead to over-fitting.

4.3 Ablation Studies and Further Analysis

Effectiveness of the prompt distribution learning. In Table 3 (Row (1) & (3)), we compare the performance of the original CLIP model f_c with our prompt distribution learning model f_g. We observe that modeling the underlying distribution of the text prototypes results in notable performance gains. For example, 3.7% accuracy improvement on 10-datasets using ViT-B/16.

Effectiveness of the prediction fusion. As described in Eq.(10), our Frolic fuses the original CLIP f_c and the prompt distribution learning model f_g via confidence matching. We compare the simple fusion $f_c + f_g$ and our adaptive fusion $f_f = f_c/\tau_c + f_g/\tau_g$ in Table 3 (Row (4) & (5)). We show that our fusion technique outperforms the simple fusion by a large margin. Recall that our adaptive fusion method addresses situations where f_g is more overconfident than f_c. In Figure 3, we illustrate the relationship between performance gains over simple fusion—*i.e.*, $\mathsf{Acc}(f_c/\tau_c + f_g/\tau_g) - \mathsf{Acc}(f_c + f_g)$—and the confidence difference—*i.e.*, $|\mathsf{conf}(f_g, 1) - \mathsf{conf}(f_c, \tau_c)|$. We present this as a scatter plot where each point represents a dataset, and we have fitted these points with a line. As expected, larger confidence differences correlate with more significant improvements.

Effectiveness of the bias correction. Row (2) and (6) in Table 3 demonstrate the effectiveness of our debiasing method, which can further improve the base CLIP model f_c and the fusion model f_d

8

Table 4: Comparison of accuracy (%) between our Frolic and other label bias correcting methods for CLIP ViT-B/16.

Model	Pets	Flowers	Aircraft	DTD	EuroSAT	Cars	Food	SUN	Caltech	UCF	ImageNet	Avg.
CLIP [29]	89.1	71.4	24.8	44.3	47.7	65.2	86.1	62.5	92.9	66.7	68.7	65.4
TDE [34]	84.1	65.8	27.4	49.8	55.3	60.3	84.6	65.5	91.6	68.2	65.9	65.3
Implicit	91.4	71.4	30.6	54.2	56.8	66.0	86.6	69.5	93.5	72.6	69.8	69.3
Frolic	**92.9**	**74.8**	**31.4**	**56.1**	**58.5**	**69.1**	**87.1**	**70.8**	**95.1**	**75.2**	**70.9**	**70.9**
Oracle Frolic	93.1	77.5	32.2	57.3	59.8	69.8	87.4	71.2	95.7	76.3	71.5	71.9
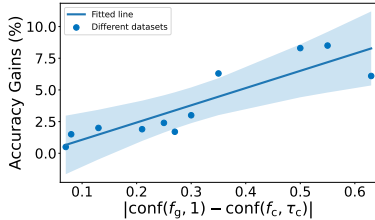
Figure 3: Relation between gains and confidence differences.
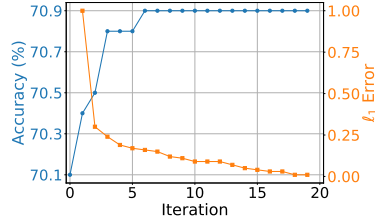
Figure 4: Convergence of accuracy and ℓ_1 error of on ImageNet.

across various backbones and datasets. We also compare our debiasing method with other label bias correction methods in Table 4. The descriptions of TDE [34] and the Implicit method can be found in Section 3.4. The results reveal that TDE [34] does not consistently perform well across all datasets. In contrast, while the implicit method using downstream data enhances zero-shot performance, it underperforms compared to our debiasing method, which shows an average gain of 1.6% over the implicit method. To further assess our method's potential, we replaced pseudo-labeling with ground truth labels. The results reveal that the maximum achievable accuracy surpasses our method by 1.0%, highlighting the importance of our iterative approach for more accurate pseudo-labeling.

Convergence of Algorithm 1. Our method Frolic, as described in Algorithm 2, iteratively solves for the prior $\boldsymbol{\beta}$. In Figure 4, we examine the convergence by displaying the errors $\ell_1 = \|\boldsymbol{\beta}^t - \boldsymbol{\beta}^{t-1}\|_1$ and the accuracy across iterations. We find that the resultant accuracy saturates after only 6 steps, and the relative ℓ_1 error decreases to less than $\epsilon = 0.01$ after 10 steps.

Comparison with other prompt-based methods. The popular prompt-based methods, such as CoOp [46] and CoCoOp [45], require a training procedure with labeled samples while our method does not involve any training. To ensure a fair comparison, we compare our Frolic with CoOp and CoCoOp on across-dataset results, where the CoOp and CoCoOp are trained only with the labeled samples from the ImageNet dataset and then directly tested on the remaining datasets. The results shown in Table 5 demonstrate that our Frolic not only avoids the complexities of training but also exhibits superior generalization performance compared to these methods.

Table 5: Comparison of accuracy (%) between our Frolic and prompt-based methods for CLIP ViT-B/16. $*$ denotes our method built upon InMaP [28]

Model	ImageNet	Pets	Flowers	Aircraft	DTD	EuroSAT	Cars	Food	SUN	Caltech	UCF
CoOp [46]	71.5	93.7	89.1	64.5	68.7	85.3	18.4	64.1	41.9	46.3	66.5
CoCoOp [45]	71.0	94.4	90.1	65.3	71.8	86.0	22.9	67.3	45.7	45.3	68.2
Frolic*	**73.3**	**95.4**	**93.6**	**71.7**	**74.3**	**88.2**	**31.8**	**72.8**	**58.0**	**65.3**	**75.9**

Comparison with adapter-based methods. The adapter-based methods, *e.g.*, LFA [24] and Tip-Adapter [44] boost the CLIP's generalization using labeled training samples. In contrast, our Frolic doesn't require any labeled samples. We evaluate our method with LFA and Tip-Adapert on the

9

Table 6: Comparison of accuracy (%) between our Frolic and adapter-based distribution methods for CLIP ViT-B/16. * denotes our method built upon InMaP [28]

Model	IN	IN-A	IN-V2	IN-R	IN-Sketch	Average
LFA [24]	72.6	51.5	64.7	76.1	48.0	62.5
Tip-Adapter [44]	70.5	49.8	63.1	76.9	48.1	61.6
Frolic*	**73.3**	**52.8**	**63.8**	**79.6**	**52.9**	**64.4**

ImageNet and its variants dataset, where the LFA and Tip-Adapter only utilize the labeled samples from the ImageNet dataset. The results in Table 6 show that our method achieves the best performance across all datasets with nearly 3% improvements in averaged accuracy over LFA.

Running time. Our method Frolic is completely training-free, unlike prompt tuning approaches such as TPT [32] and TDA [15], which involve back-propagating through an expensive encoder during optimization. We assess the wall-clock time of Frolic, TPT, and TDA in Table 7, using the CLIP ViT-B/16 model on ImageNet. These evaluations are conducted on a single NVIDIA A100 GPU. The results indicate that our method not only requires less time but also delivers superior performance.

Table 7: Comparison of running time on ImageNet with ViT-B/16.

Model	Running Time	Accuracy
CLIP [29]	6min	68.7
TPT [32]	6h	68.9
TDA [15]	15min	69.5
Frolic	6.5min	71.1

5 Societal Impact, Limitation and Conclusion

Societal impact and limitation. Models pre-trained on large-scale web-crawled datasets may incorporate knowledge from noisy or malicious samples.

Limitation. Our approach assumes that the feature representations follow a mixture of Gaussian; however, this assumption may not always hold. On the other hand, the quality and distribution of data used in pre-training can significantly impact the performance of pre-trained models. Our method relies on the capabilities of pre-trained models for downstream tasks, if the pre-trained knowledge differs from the downstream tasks, the efficacy of our method may be limited.

Conclusion. In this work, we propose label-**F**ree p**ro**mpt distribution **l**earning and b**i**as **c**orrection, dubbed as **Frolic**, framework to boost the performance of zero-shot models. Our Frolic models each class prototype via a Gaussian distribution and fuses the learned model with the original CLIP [29] via confidence matching. The proposed framework further effectively removes the label bias without accessing to the pre-training data. Extensive experiments across various datasets demonstrate the effectiveness of our approach.

10

Acknowledgments and Disclosure of Funding

The work is supported by the National Natural Science Foundation of China (Grants No. 62202439), and the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG2-PhD-2021-01-002). This work is also supported by the advanced computing resources provided by the Supercomputing Center of the USTC.

References

[1] James Urquhart Allingham, Jie Ren, Michael W. Dusenberry, Xiuye Gu, Yin Cui, Dustin Tran, Jeremiah Zhe Liu, and Balaji Lakshminarayanan. A simple zero-shot prompt weighting technique to improve prompt ensembling in text-image models. In *ICML*, 2023.

[2] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. In *NeurIPS*, 2019.

[3] Christopher M Bishop. Pattern recognition and machine learning. *Springer*, 2:200–202.

[4] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 - mining discriminative components with random forests. In *ECCV*, 2014.

[5] Hao Chen, Bhiksha Raj, Xing Xie, and Jindong Wang. On catastrophic inheritance of large foundation models. *arXiv preprint arXiv:2402.01909*, 2024.

[6] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *CVPR*, 2023.

[7] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, 2014.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[9] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR Workshops*, 2004.

[10] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *ICML*, 2017.

[11] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, 12(7):2217–2226, 2019.

[12] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, 2021.

[13] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. 2021.

[14] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. In *CVPR*, 2021.

[15] Adilbek Karmanov, Dayan Guan, Shijian Lu, Abdulmotaleb El Saddik, and Eric Xing. Efficient test-time adaptation of vision-language models. In *CVPR*, 2024.

[16] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV Workshops*, 2013.

11

[17] Ananya Kumar, Tengyu Ma, Percy Liang, and Aditi Raghunathan. Calibrated ensembles can mitigate accuracy tradeoffs under distribution shift. In *UAI*, 2022.

[18] Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. In *CVPR*, 2022.

[19] Rachel Luo, Shengjia Zhao, Jiaming Song, Jonathan Kuck, Stefano Ermon, and Silvio Savarese. Privacy preserving recalibration under domain shift. *CoRR*, abs/2008.09643, 2020.

[20] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew B. Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *CoRR*, abs/1306.5151, 2013.

[21] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *ICLR*, 2021.

[22] RV Mises and Hilda Pollaczek-Geiringer. Praktische verfahren der gleichungsauflösung. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 9(1):58–77, 1929.

[23] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008.

[24] Yassine Ouali, Adrian Bulat, Brais Martínez, and Georgios Tzimiropoulos. Black box few-shot adaptation for vision-language models. *CoRR*, abs/2304.01752, 2023.

[25] Shubham Parashar, Zhiqiu Lin, Tian Liu, Xiangjue Dong, Yanan Li, Deva Ramanan, James Caverlee, and Shu Kong. The neglected tails of vision-language models. *arXiv preprint arXiv:2401.12425*, 2024.

[26] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *CVPR*, 2012.

[27] Sarah M. Pratt, Ian Covert, Rosanne Liu, and Ali Farhadi. What does a platypus look like? generating customized prompts for zero-shot image classification. In *ICCV*, 2023.

[28] Qi Qian, Yuanhong Xu, and Juhua Hu. Intra-modal proxy learning for zero-shot visual categorization with CLIP. In *NeurIPS*, 2023.

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[30] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *ICML*, 2019.

[31] Jameel Abdul Samadh, Hanan Gani, Noor Hussein, Muhammad Uzair Khattak, Muzammal Naseer, Fahad Shahbaz Khan, and Salman H. Khan. Align your prompts: Test-time prompting with distribution alignment for zero-shot generalization. In *NeurIPS*, 2023.

[32] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. In *NeurIPS*, 2022.

[33] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.

[34] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. In *NeurIPS*, 2020.

[35] Vishaal Udandarao, Ankush Gupta, and Samuel Albanie. Sus-x: Training-free name-only transfer of vision-language models. *CoRR*, abs/2211.16198, 2022.

12

[36] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019.

[37] Xudong Wang, Zhirong Wu, Long Lian, and Stella X Yu. Debiased learning from naturally imbalanced pseudo-labels. In *CVPR*, 2022.

[38] Zhengbo Wang, Jian Liang, Lijun Sheng, Ran He, Zilei Wang, and Tieniu Tan. A hard-to-beat baseline for training-free CLIP-based adaptation. In *ICLR*, 2024.

[39] Zhicai Wang, Yanbin Hao, Tingting Mu, Ouxiang Li, Shuo Wang, and Xiangnan He. Bi-directional distribution alignment for transductive zero-shot learning. In *CVPR*, 2023.

[40] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. In *CVPR*, 2022.

[41] Wenhao Wu, Huanjin Yao, Mengxi Zhang, Yuxin Song, Wanli Ouyang, and Jingdong Wang. Gpt4vis: What can GPT-4 do for zero-shot visual recognition? *CoRR*, abs/2311.15732, 2023.

[42] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010.

[43] Xuanyu Yi, Jiajun Deng, Qianru Sun, Xian-Sheng Hua, Joo-Hwee Lim, and Hanwang Zhang. Invariant training 2d-3d joint hard samples for few-shot point cloud recognition. In *ICCV*, 2023.

[44] Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free adaption of CLIP for few-shot classification. In *ECCV*, 2022.

[45] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, 2022.

[46] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 130(9):2337–2348, 2022.

[47] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. In *ICCV*, 2023.

[48] Beier Zhu, Yulei Niu, Xian-Sheng Hua, and Hanwang Zhang. Cross-domain empirical risk minimization for unbiased long-tailed classification. In *AAAI*, 2022.

[49] Beier Zhu, Kaihua Tang, Qianru Sun, and Hanwang Zhang. Generalized logit adjustment: Calibrating fine-tuned models by removing label bias in foundation models. In *NeurIPS*, 2023.

[50] Xingyu Zhu, Beier Zhu, Yi Tan, Shuo Wang, Yanbin Hao, and Hanwang Zhang. Selective vision-language subspace projection for few-shot CLIP. In *ACM MM*, 2024.

13

Contents

1   **Introduction**   **1**

2   **Related Works**   **3**

3   **Methods**   **3**

     3.1   Setup 3

     3.2   Label-Free Prompt Distribution Learning 4

     3.3   Prediction Fusion via Adaptive Calibration. 4

     3.4   Correcting Pre-training Label Bias via Label-Free Logit Adjustment 5

4   **Experiments**   **7**

     4.1   Setup 7

     4.2   Main Results 7

     4.3   Ablation Studies and Further Analysis 8

5   **Societal Impact, Limitation and Conclusion**   **10**

A   **Theoretical Analysis**   **15**

     A.1   Proof of Eq. (6): Estimation of Class Covariance from Marginal Second Order Moment   15

     A.2   Proof of Eq. (5): Estimation of the Priors of Gaussian Mixture Models 16

     A.3   Proof of Eq. (7): Parameters of our Learned Model 17

     A.4   Proof of Eq. (11): Debiased Classifier for Downstream Data 17

     A.5   Proof of Eq. (12): Equation to Estimate Pre-training Priors 18

     A.6   Power Method to Estimate Pretraining Priors 18

B   **Details of ImageNet Variant Datasets**   **18**

C   **Licenses**   **19**

14

A Theoretical Analysis

A.1 Proof of Eq. (6): Estimation of Class Covariance from Marginal Second Order Moment

We first derive the second order moments for a multivariate Gaussian and then for a Gaussian mixture, corresponding to the marginal distribution of $\mathbb{P}(\mathbf{x})$.

For a class j with parameters $\mathbf{z}_j$ and Σ, the conditional probability density function is given by:

$$\mathcal{N}(\mathbf{x}; \mathbf{z}_j, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\{-\frac{1}{2}(\mathbf{x} - \mathbf{z}_j)^\top \Sigma^{-1}(\mathbf{x} - \mathbf{z}_j)\} \tag{17}$$

The second order moment generating function for class j is:

$$M_j = \mathbb{E}_{\mathbf{x} \in \mathcal{C}_j}[\mathbf{x}\mathbf{x}^\top] = \int_{\mathbf{x}} \mathcal{N}(\mathbf{x}; \mathbf{z}_j, \Sigma)\mathbf{x}\mathbf{x}^\top d\mathbf{x} \tag{18}$$

$$= \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \int_{\mathbf{x}} \exp\{-\frac{1}{2}(\mathbf{x} - \mathbf{z}_j)^\top \Sigma^{-1}(\mathbf{x} - \mathbf{z}_j)\}\mathbf{x}\mathbf{x}^\top d\mathbf{x} \tag{19}$$

$$\overset{(a)}{=} \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \int_{\mathbf{y}} \exp\{-\frac{1}{2}\mathbf{y}^\top \Sigma^{-1}\mathbf{y}\}(\mathbf{y} + \mathbf{z}_j)(\mathbf{y} + \mathbf{z}_j)^\top d\mathbf{y} \tag{20}$$

$$= \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \int_{\mathbf{y}} \exp\{-\frac{1}{2}\mathbf{y}^\top \Sigma^{-1}\mathbf{y}\}(\mathbf{y}\mathbf{y}^\top + \underbrace{\mathbf{y}\mathbf{z}_j^\top + \mathbf{z}_j\mathbf{y}^\top}_{\text{vanish by symmetry}} + \mathbf{z}_j\mathbf{z}_j^\top)d\mathbf{y} \tag{21}$$

$$\overset{(b)}{=} \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \int_{\mathbf{y}} \exp\{-\frac{1}{2}\mathbf{y}^\top \Sigma^{-1}\mathbf{y}\}(\mathbf{y}\mathbf{y}^\top + \mathbf{z}_j\mathbf{z}_j^\top)d\mathbf{y} \tag{22}$$

$$\overset{(c)}{=} \mathbf{z}_j\mathbf{z}_j^\top + \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \int_{\mathbf{y}} \exp\{-\frac{1}{2}\mathbf{y}^\top \Sigma^{-1}\mathbf{y}\}(\mathbf{y}\mathbf{y}^\top)d\mathbf{y}. \tag{23}$$

$\overset{(a)}{=}$ holds as we change the integral variables $\mathbf{y} = \mathbf{x} - \mathbf{z}_j$. We have $\overset{(b)}{=}$ because the $\exp(\cdot)$ function is an even function of $\mathbf{y}$ and the factors $\mathbf{y}\mathbf{z}_j^\top$ and $\mathbf{z}_j\mathbf{y}^\top$ will vanish during integral by symmetry. For $\overset{(c)}{=}$, we take the term $\mathbf{z}_j\mathbf{z}_j^\top$ outside of the integral as they are constant.

The covariance matrix Σ and its inverse matrix Σ^{-1} the can be expressed through an expansion in terms of its eigenvalues $\{\lambda_i\}_{i=1}^d$ and eigenvectors $\{\mathbf{u}_i\}_{i=1}^d$:

$$\Sigma = \sum_{i=1}^d \lambda_i \mathbf{u}_i \mathbf{u}_i^\top, \quad \Sigma^{-1} = \sum_{i=1}^d \frac{1}{\lambda_i}\mathbf{u}_i\mathbf{u}_i^\top \tag{24}$$

Similarly, we can decompose $\mathbf{y}$ using the set of eigenvectors: $\mathbf{y} = \sum_{j=1}^d e_j \mathbf{u}_j$, where $e_j = \mathbf{u}_j^T \mathbf{y}$. (We temporarily abuse the subscript j here. It does *not* represent class j until we reach Eq. (32)) We have the following expression:

$$\mathbf{y}\mathbf{y}^\top = \sum_{i=1}^d \sum_{j=1}^d e_i e_j \mathbf{u}_i \mathbf{u}_j^\top \tag{25}$$

$$\mathbf{y}^\top \Sigma^{-1}\mathbf{y} = \sum_{i=1}^d e_i\mathbf{u}_i^\top \sum_{k=1}^d \frac{1}{\lambda_k}\mathbf{u}_k\mathbf{u}_k^\top \sum_{j=1}^d e_j\mathbf{u}_j \overset{(d)}{=} \sum_{k=1}^d (\frac{e_k}{\sqrt{\lambda_k}})^2 \tag{26}$$

We obtain $\overset{(d)}{=}$ due to the property of eigenvalues, *i.e.*, $\mathbf{u}_i^\top \mathbf{u}_i = 1$ and $\mathbf{u}_i^\top \mathbf{u}_j = 0$, for $i \neq j$. Denote $U = [\mathbf{u}_1, ..., \mathbf{u}_d]^\top$, we have $\mathbf{e} = U\mathbf{y}$. As the determinant $|U| = 1$, the probability density after transformed remains unchanged: $\mathbb{P}(\mathbf{e}) = |U|^{-1}\mathbb{P}(\mathbf{y}) = \mathbb{P}(\mathbf{y})$. Apply Eq. (25) and Eq. (26) into

15

Eq. (23), we have:

$$\frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \int_{\mathbf{y}} \exp\{-\frac{1}{2}\mathbf{y}^\top \Sigma^{-1}\mathbf{y}\}(\mathbf{y}\mathbf{y}^\top)\mathrm{d}\mathbf{y} \tag{27}$$

$$= \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \sum_{i=1}^{d}\sum_{j=1}^{d} \mathbf{u}_i\mathbf{u}_j^\top \int_{\mathbf{e}} \exp\{\sum_{k=1}^{d} -\frac{1}{2}(\frac{e_k}{\sqrt{\lambda_k}})^2\}e_i e_j \mathrm{d}\mathbf{e} \tag{28}$$

$$= \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \sum_{i=1}^{d}\sum_{j=1}^{d} \mathbf{u}_i\mathbf{u}_j^\top \int_{\mathbf{e}} \prod_{k=1}^{d} \exp\{-\frac{1}{2}(\frac{e_k}{\sqrt{\lambda_k}})^2\}e_i e_j \mathrm{d}\mathbf{e} \tag{29}$$

$$\overset{(e)}{=} \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \sum_{i=1}^{d} \mathbf{u}_i\mathbf{u}_i^\top \int_{e_i} \exp\{-\frac{1}{2}(\frac{e_i}{\sqrt{\lambda_i}})^2\}e_i^2 \mathrm{d}e_i \tag{30}$$

$$\overset{(f)}{=} \sum_{i=1}^{d} \mathbf{u}_i\mathbf{u}_i^\top \int_{e_i} \frac{1}{\sqrt{2\pi\lambda_i}} \exp\{-\frac{1}{2}(\frac{e_i}{\sqrt{\lambda_i}})^2\}e_i^2 \mathrm{d}e_i \tag{31}$$

$$\overset{(g)}{=} \sum_{i=1}^{d} \mathbf{u}_i\mathbf{u}_i^\top \lambda_i = \Sigma \tag{32}$$

For $\overset{(e)}{=}$, the terms $i \neq j$ disappear by symmetry similar to $\overset{(b)}{=}$. We make use of $|\Sigma| = \prod_{i=1}^{d}\lambda_i$ for $\overset{(f)}{=}$. We have $\overset{(g)}{=}$ because we regard $e_i \sim \mathcal{N}(0, \sqrt{\lambda_i})$ and note that $\mathbb{E}[e_i^2] = \mathsf{var}[e_i] + \mathbb{E}[e_i]^2 = \lambda_i + 0 = \lambda_i$. Combining Eq. (32) with Eq. (23), we have the second order moment for class j is:

$$M_j = \mathbf{z}_j\mathbf{z}_j^\top + \Sigma. \tag{33}$$

Using a Gaussian mixture model with the priors $\{\pi_j\}_j^K$, $\mathbb{P}(\mathbf{x})$ is given by:

$$\mathbb{P}(\mathbf{x}) = \sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}; \mathbf{z}_j, \Sigma). \tag{34}$$

The second order moment for the marginal distribution $\mathbb{P}(\mathbf{x})$ is:

$$M = \mathbb{E}[\mathbf{x}\mathbf{x}^\top] = \int_{\mathbf{x}} \sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}; \mathbf{z}_j, \Sigma)\mathrm{d}\mathbf{x}\mathbf{x}^\top\mathbf{x} \tag{35}$$

$$= \sum_{j=1}^{K} \pi_j \int_{\mathbf{x}} \mathcal{N}(\mathbf{x}; \mathbf{z}_j, \Sigma)\mathbf{x}\mathbf{x}^\top\mathrm{d}\mathbf{x} \tag{36}$$

$$= \sum_{j=1}^{K} \pi_j M_j = \sum_{j=1}^{K} \pi_j(\mathbf{z}_j\mathbf{z}_j^\top + \Sigma) \tag{37}$$

$$= \sum_{j=1}^{K} \pi_j\mathbf{z}_j\mathbf{z}_j^\top + (\sum_{j=1}^{K} \pi_j)\Sigma = \Sigma + \sum_{j=1}^{K} \pi_j\mathbf{z}_j\mathbf{z}_j^\top \tag{38}$$

$\square$

A.2 Proof of Eq. (5): Estimation of the Priors of Gaussian Mixture Models

The expectation of $\mathbf{x}$ is defined as:

$$\mathbb{E}[\mathbf{x}] = \int_{\mathbf{x}} \sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}; \mathbf{z}_j, \Sigma)\mathbf{x}\mathrm{d}\mathbf{x} = \sum_{j=1}^{K} \pi_j \int_{\mathbf{x}} \mathcal{N}(\mathbf{x}; \mathbf{z}_j, \Sigma)\mathbf{x}\mathrm{d}\mathbf{x} = \sum_{j=1}^{K} \pi_j\mathbf{z}_j \tag{39}$$

Denote $\boldsymbol{\pi} = [\pi_1, .., \pi_K]^\top$, $Z = [\mathbf{z}_1, .., \mathbf{z}_K]^T$, and the expectation of $\mathbf{x}$ as $\boldsymbol{\mu}$, Eq. (39) can be rewrite as:

$$\boldsymbol{\mu} = Z\boldsymbol{\pi}. \tag{40}$$

Therefore, the priors can be solve by $\boldsymbol{\pi} = Z^{-1}\boldsymbol{\mu}$.
$\square$

16

A.3 Proof of Eq. (7): Parameters of our Learned Model

The posterior of classes $\mathbb{P}(y|\mathbf{x})$ can be expression as:

$$\mathbb{P}(y|\mathbf{x}) = \frac{\mathbb{P}(\mathbf{x}|y)\mathbb{P}(y)}{\mathbb{P}(\mathbf{x})} \propto \mathcal{N}(\mathbf{x}; \mathbf{z}_y, \Sigma)\pi_y. \tag{41}$$

To classify $\mathbf{x}$, we seek the class y that maximizes this posterior. Since the term $\mathbb{P}(\mathbf{x})$ does not depend on y, we can simplify our task to maximizing $\mathcal{N}(\mathbf{x}; \mu_y, \Sigma)\pi_y$. Taking natural logarithm gives:

$$\ln \mathcal{N}(\mathbf{x}; \mathbf{z}_y, \Sigma)\pi_y = \ln \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \exp\{-\frac{1}{2}(\mathbf{x} - \mathbf{z}_y)^\top \Sigma^{-1}(\mathbf{x} - \mathbf{z}_y)\}\pi_y \tag{42}$$

$$= \ln \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} - \frac{1}{2}(\mathbf{x} - \mathbf{z}_y)^\top \Sigma^{-1}(\mathbf{x} - \mathbf{z}_y) + \ln \pi_y \tag{43}$$

$$= c_1 - \frac{1}{2}\mathbf{x}^T \Sigma^{-1}\mathbf{x} + \mathbf{z}_y^\top \Sigma^{-1}\mathbf{x} - \frac{1}{2}\mathbf{z}_y^T \Sigma^{-1}\mathbf{z}_y + c_2 \tag{44}$$

$$= \mathbf{z}_y^\top \Sigma^{-1}\mathbf{x} - \frac{1}{2}\mathbf{z}_y^T \Sigma^{-1}\mathbf{z}_y + c \tag{45}$$

$$= \mathbf{w}_y^T \mathbf{x} + b_y + c \tag{46}$$

The first term in Equation (43) is constant; we incorporate it using a constant c_1. Consider that most test benchmarks are generally class-balanced, we use a uniform prior c_2 to incorporate $\ln \pi_y$. In Eq. (45), we use c to absorb all constant terms, including c_1, c_2 and $-\frac{1}{2}\mathbf{x}^T \Sigma^{-1}\mathbf{x}$. Let $\mathbf{w}_j = \hat{\Sigma}^{-1}\mathbf{z}_j$ and $b_j = -\frac{1}{2}\mathbf{z}_j^\top \mathbf{w}_j$, we get Eq. (46). □

A.4 Proof of Eq. (11): Debiased Classifier for Downstream Data

Proposition 1. *(Modified from Theorem 1 in [14]). Let $\mathbb{P}_{\mathsf{pt}}(y|\mathbf{x})$ and $\mathbb{P}_{\mathsf{ds}}(y|\mathbf{x})$ be the distributions of the pre-train and downstream data, respectively. Let $\beta_y = \mathbb{P}_{\mathsf{pt}}(y)$ and $\pi_y = \mathbb{P}_{\mathsf{ds}}(y)$ denote the priors of the pre-train and the downstream data, respectively. Assume the likelihood $\mathbb{P}(\mathbf{x}|y)$ is unchanged between pre-train and downstream data, i.e., $\mathbb{P}(\mathbf{x}|y) = \mathbb{P}_{\mathsf{pt}}(\mathbf{x}|y) = \mathbb{P}_{\mathsf{ds}}(\mathbf{x}|y)$. If $f_{\mathsf{pt}}(\mathbf{x})_y$ is the logit of class y from the softmax model to estimate $\mathbb{P}_{\mathsf{pt}}(y|\mathbf{x})$, then the estimated $\mathbb{P}_{\mathsf{ds}}(y|\mathbf{x})$ is formulated as:*

$$\mathbb{P}_{\mathsf{ds}}(y|\mathbf{x}) = \mathrm{softmax}(f_{\mathsf{pt}}(\mathbf{x}) - \ln \boldsymbol{\beta} + \ln \boldsymbol{\pi})_y, \tag{47}$$

where $\boldsymbol{\beta} = [\beta_1, ..., \beta_K]$ and $\boldsymbol{\pi} = [\pi_1, ..., \pi_K]$.

Proof.

$$\mathbb{P}_{\mathsf{ds}}(y|\mathbf{x}) = \frac{\mathbb{P}_{\mathsf{ds}}(\mathbf{x}|y)\mathbb{P}_{\mathsf{ds}}(y)}{\mathbb{P}_{\mathsf{ds}}(\mathbf{x})} = \frac{\mathbb{P}_{\mathsf{pt}}(\mathbf{x}|y)\mathbb{P}_{\mathsf{ds}}(y)}{\mathbb{P}_{\mathsf{ds}}(\mathbf{x})} \tag{48}$$

$$= \frac{\mathbb{P}_{\mathsf{pt}}(\mathbf{x}|y)\mathbb{P}_{\mathsf{pt}}(y)}{\mathbb{P}_{\mathsf{pt}}(\mathbf{x})} \frac{\mathbb{P}_{\mathsf{ds}}(y)}{\mathbb{P}_{\mathsf{pt}}(y)} \frac{\mathbb{P}_{\mathsf{pt}}(\mathbf{x})}{\mathbb{P}_{\mathsf{ds}}(\mathbf{x})} \tag{49}$$

$$\overset{(a)}{=} \mathbb{P}_{\mathsf{pt}}(y|\mathbf{x})\frac{\pi_y}{\beta_y}\frac{1}{Z} = \mathrm{softmax}(f_{\mathsf{pt}}(\mathbf{x}))_y \frac{\pi_y}{\beta_y}\frac{1}{Z} \tag{50}$$

$$= \frac{\exp(f_{\mathsf{pt}}(\mathbf{x})_y)}{Z\sum_{j=1}^K \exp(f_{\mathsf{pt}}(\mathbf{x})_j)}\frac{\exp(\ln \pi_y)}{\exp(\ln \beta_y)} \tag{51}$$

$$= \frac{\exp(f_{\mathsf{pt}}(\mathbf{x})_y - \ln \beta_y + \ln \pi_y)}{Z\sum_{j=1}^K \exp(f_{\mathsf{pt}}(\mathbf{x})_j)} \tag{52}$$

$$\overset{(b)}{=} \frac{\exp(f_{\mathsf{pt}}(\mathbf{x})_y - \ln \beta_y + \ln \pi_y)}{\sum_{j=1}^K \exp(f_{\mathsf{pt}}(\mathbf{x})_j - \ln \beta_j + \ln \pi_j)} \tag{53}$$

$$= \mathrm{softmax}(f_{\mathsf{pt}}(\mathbf{x}) - \ln \boldsymbol{\beta} + \ln \boldsymbol{\pi})_y. \tag{54}$$

For $\overset{(a)}{=}$, we denote the term that is not related to y as $\frac{1}{Z} = \frac{\mathbb{P}_{\mathsf{pt}}(\mathbf{x})}{\mathbb{P}_{\mathsf{ds}}(\mathbf{x})}$. We derive $\overset{(b)}{=}$ from the requirement that $\mathbb{P}_{\mathsf{ds}}(y|\mathbf{x})$, being a probability, must sum to 1 across all possible classes $y \in [K]$:

$$\sum_{i=1}^K \mathbb{P}_{\mathsf{ds}}(i|\mathbf{x}) = \frac{\sum_{i=1}^K \exp(f_{\mathsf{pt}}(\mathbf{x})_i - \ln \beta_i + \ln \pi_i)}{Z\sum_{j=1}^K \exp(f_{\mathsf{pt}}(\mathbf{x})_j)} = 1. \tag{55}$$

17

Therefore, we have $Z \sum_{j=1}^{K} \exp(f_{\text{pt}}(\mathbf{x})_j) = \sum_{i=1}^{K} \exp(f_{\text{pt}}(\mathbf{x})_i - \ln \beta_i + \ln \pi_i)$. In our context, the pre-trained model f_{pt} is equivalent to our f_{f}. $\qquad \square$

A.5 Proof of Eq. (12): Equation to Estimate Pre-training Priors

Proposition 2. *Let* $s(\mathbf{x}) = [\mathbb{P}(Y = 1|\mathbf{x}), ..., \mathbb{P}(Y = K|\mathbf{x})]^\top \in \mathbb{R}^K$ *be the likelihood vector,* $\mathbf{s}_j = \mathbb{E}_{\mathbf{x}|Y=j}[s(\mathbf{x})]$ *and* $S = [\mathbf{s}_1, ..., \mathbf{s}_K] \in \mathbb{R}^{K \times K}$. *The pretraining prior* $\boldsymbol{\beta} = [\beta_1, ..., \beta_K]^\top \in \mathbb{R}^K$ *must satisfy the linear system:*

$$(S - I)\boldsymbol{\beta} = \mathbf{0}. \tag{56}$$

Proof.

$$\beta_y = \int_{\mathbf{x}} \mathbb{P}_{\text{pt}}(\mathbf{x}) \mathbb{P}_{\text{pt}}(y|\mathbf{x}) d\mathbf{x} = \int_{\mathbf{x}} \sum_{y' \in [K]} \mathbb{P}(\mathbf{x}|y') \beta_{y'} \mathbb{P}_{\text{pt}}(y|\mathbf{x}) d\mathbf{x} \tag{57}$$

$$= \sum_{y' \in [K]} \beta_{y'} \int_{\mathbf{x}} \mathbb{P}(\mathbf{x}|y') \mathbb{P}_{\text{pt}}(y|\mathbf{x}) d\mathbf{x} \tag{58}$$

$$= \sum_{y' \in [K]} \beta_{y'} \mathbb{E}_{\mathbf{x}|Y=y'}[\mathbb{P}_{\text{pt}}(y|\mathbf{x})] \tag{59}$$

$$= \sum_{y' \in [K]} \beta_{y'} \mathbb{E}_{\mathbf{x}|Y=y'}[s(\mathbf{x})]_y, \tag{60}$$

$$= \sum_{y' \in [K]} S_{yy'} \beta_{y'} \tag{61}$$

Note that Equation (61) precisely represents the matrix multiplication given by:

$$\boldsymbol{\beta} = S\boldsymbol{\beta} \tag{62}$$

By moving the RHS term to the LHS, Eq. (56) is obtained. $\qquad \square$

A.6 Power Method to Estimate Pretraining Priors

The solution to Equation (62) involves finding the eigenvector corresponding to the eigenvalue of 1 for the matrix S. We can apply SVD decomposition to find the solution; however, we find that the results might be numerically unstable. Instead, we adopt power iteration from [22]. Like the Jacobi and Gauss-Seidel methods, the power method for approximating eigenvalues is iterative. We first initialize $\boldsymbol{\beta}_0 = [\frac{1}{K}, ..., \frac{1}{K}]$ of a uniform distribution. Then, we perform the sequence:

$$\bar{\boldsymbol{\beta}}_t = S\boldsymbol{\beta}_{t-1} \tag{63}$$

$$\boldsymbol{\beta}_t = \frac{\bar{\boldsymbol{\beta}}_t}{\|\bar{\boldsymbol{\beta}}_t\|_1} \tag{64}$$

We repeat the sequence until the relative change is small: $\|\boldsymbol{\beta}_t - \boldsymbol{\beta}_{t-1}\| < \epsilon$.

B Details of ImageNet Variant Datasets

ImageNet-V2 [30]: sampling from the original ImageNet and including 10,000 images of 1,000 ImageNet categories.

ImageNet Sketch [36]: including 138 50,000 images and covering 1,000 ImageNet categories.

ImageNet-R [12]: containing renditions (*e.g.*, art, cartoons, graffiti) for ImageNet classes, comprising 30,000 images from 200 ImageNet categories.

ImageNet-A [13]: collecting real-world images that are misclassified by ResNet-50, totaling 7,500 images from 200 of ImageNet categories.

ObjectNet: [2] including 50,000 test images with rotation, background, and viewpoint, and overlapping 113 classes with ImageNet.

18

C Licenses

All the datasets we considered are publicly available, we list their licences and URLs as follows:

- **Caltech101** [9]: Non-commercial, `https://data.caltech.edu/records/mzrjq-6wc02`.
- **OxfordPets** [26]: MIT License, `https://www.robots.ox.ac.uk/~vgg/data/pets/`.
- **StanfordCars** [16]: MIT License, `https://www.kaggle.com/datasets/jessicali9530/stanford-cars-dataset`
- **Flowers102** [23]: MIT License, `https://www.robots.ox.ac.uk/~vgg/data/flowers/102/index.html`.
- **Food101** [4]: Non-commercial,`https://data.vision.ee.ethz.ch/cvl/datasets_extra/food-101/`
- **FGVCAircraft** [20]: Non-commercial, `https://www.robots.ox.ac.uk/~vgg/data/fgvc-aircraft/`.
- **SUN397** [42]: Non-commercial, `https://vision.princeton.edu/projects/2010/SUN/`.
- **DTD** [7]: Non-commercial, `https://www.robots.ox.ac.uk/~vgg/data/dtd/`
- **EuroSAT** [11]: MIT License, `https://github.com/phelber/EuroSAT`
- **UCF101** [33]: Non-commercial,`https://www.crcv.ucf.edu/data/UCF101.php`
- **ImageNet** [8]: Non-commercial, `http://image-net.org`.
- **ImageNetV2** [30]: MIT License, `https://github.com/modestyachts/ImageNetV2`.
- **ImageNet-R** [12]: MIT License, `https://github.com/hendrycks/imagenet-r`.
- **ImageNet-Sketch** [36]: MIT License,`https://github.com/HaohanWang/ImageNet-Sketch`.
- **ImageNet-A** [13]: MIT License, `https://github.com/hendrycks/natural-adv-examples`.
- **ObjectNet** [2]: Creative Commons Attribution 4.0, `https://objectnet.dev`.

19

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We point out that existing prompt learning methods can not capture the various image representations and ignore the label bias in CLIP. We propose Frolic to address these problems. Experiments on various datasets demonstrate the effectiveness of our method.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations have been discussed in Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

20

Answer: [Yes]

Justification: We have provided the proof related to our method in Section A.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We have provided the Frolic Algorithm in Algorithm 1 and Algorithm 2, and included the implementation details in Section 4.1.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

21

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have uploaded the codes in supplemental materials.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We have introduced the testing details in Section 4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Given the zero-shot models, the process of our method is deterministic. Running multiple times will not introduce randomness.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

22

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have introduced the GPU type to reproduce results in Section 4.

   Guidelines:
   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics.

   Guidelines:
   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have discussed the societal impact on Section 5.

    Guidelines:
    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

23

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our method aims to improve the generalization ability of zero-shot models, which poses no such risks to the best of our knowledge.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We have provided the licenses of each dataset in Section C.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

24

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: We do not involve such experiments.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

25