Zero-Shot Logit Adjustment

Dubing Chen[1*] , **Yuming Shen**[2*] , **Haofeng Zhang**[1✉] , **Philip H.S. Torr**[2]
[1]Nanjing University of Science and Technology
[2]University of Oxford
{db.chen, zhanghf}@njust.edu.cn, ymcidence@gmail.com, philip.torr@eng.ox.ac.uk

Abstract

Semantic-descriptor-based Generalized Zero-Shot Learning (GZSL) poses challenges in recognizing novel classes in the test phase. The development of generative models enables current GZSL techniques to probe further into the semantic-visual link, culminating in a two-stage form that includes a generator and a classifier. However, existing generation-based methods focus on enhancing the generator's effect while neglecting the improvement of the classifier. In this paper, we first analyze of two properties of the generated pseudo unseen samples: bias and homogeneity. Then, we perform variational Bayesian inference to back-derive the evaluation metrics, which reflects the balance of the seen and unseen classes. As a consequence of our derivation, the aforementioned two properties are incorporated into the classifier training as seen-unseen priors via logit adjustment. The Zero-Shot Logit Adjustment further puts semantic-based classifiers into effect in generation-based GZSL. Our experiments demonstrate that the proposed technique achieves state-of-the-art when combined with the basic generator, and it can improve various generative Zero-Shot Learning frameworks. Our codes are available on https://github.com/cdb342/IJCAI-2022-ZLA.

1 Introduction

In recognition tasks, it is challenging when classes for training and test are different, known as Zero-Shot Learning (ZSL) problems. The goal of ZSL is to correctly recognize unseen samples with a classifier trained on seen classes. Bridging between training and test domains counts on the semantic class priors, i.e., the human-annotated attribute [Lampert *et al.*, 2009] or the word embedding [Reed *et al.*, 2016]. Through similarities between the class semantic descriptors, ZSL can transfer knowledge from seen to unseen classes without accessing the unseen class data.

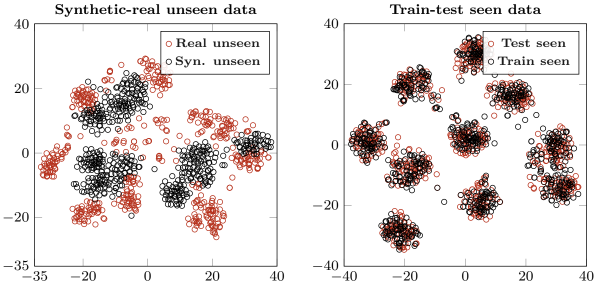

Figure 1: t-SNE visualization of the synthetic-real unseen data (left) and the real train-test seen data (right) in AWA2, we sample the same number for each class pair. The synthetic unseen data shows obvious domain bias from real test unseen data, while the training domain of the seen classes is consistent with the test domain.

Early ZSL works focus on end-to-end classifier learning, particularly the embedding models [Lampert *et al.*, 2013; Li *et al.*, 2019] which match visual features and semantic descriptors in a shared embedding space. These methods appear effective in ZSL but perform poorly in the more challenging Generalized Zero-Shot Learning (GZSL) setting [Chao *et al.*, 2016; Xian *et al.*, 2017]. More recent efforts [Xian *et al.*, 2018; Shen *et al.*, 2020] aim to improve GZSL performance by decomposing the learning process into generator learning and classifier learning. Such a two-stage strategy compensates for the feature expression of unseen classes during the classifier learning by the generated samples. A typical family of studies has focused on the improvement of the generator, exploring alternative architectures, or introducing various inductive biases. On the contrary, there is a scarcity of research on classifiers under the generative paradigm.

A principled classifier design is required to uncap the performance of the two-stage approach. The intuitive notion is introducing semantic information to classifier learning. However, both semantic and visual embedding strategies learn the same knowledge as generative models, i.e., semantic-visual links. It implies that, by directly replacing the classifier, the generated unseen samples can only serve to deliver the generator's learned links to the classifier, with little effect. As a result, classic ZSL classifiers perform worse than vanilla softmax classifiers in the generative setting [Xian *et al.*, 2018].

To determine design direcions for the classifier, we first investigate the distribution of synthetic unseen data. As shown in Figure 1, the sythetic unseen samples deviate severely from

* Equal contribution.

✉ Corresponding author. This work was supported by the National Natural Science Foundation of China (NSFC) under Grants No. 61872187, No. 62077023 and No. 62072246.

the real distribution. This bias is theoretically inevitable since the domain shift problem [Fu *et al.*, 2014], which leads to the misalignment of training and test domains. Nevertheless, we further explore the working mechanism of generated samples in Table 1, finding that the information contained in various synthetic unseen samples is quite homogenous for the classifier. Existing methods employ a large sampling size from generated unseen distribution to magnify the influence of modest valid information included in synthetic unseen samples on feature expression. However, such a resampling strategy has been proven to result in overfitting of the classifier on certain feature patterns (the biased generated unseen distribution in this setting) in other fields [Buda *et al.*, 2018; Menon *et al.*, 2020], which harms the recognition of both seen and unseen classes.

These findings imply that there is a need to restrict the generation number of unseen samples. Can we directly transmit class imbalance information to the classifier during training rather than inefficiently increasing the expression of unseen classes by resampling? In this work, we regard the **bias** and **homogeneity** of generated unseen samples as special prior knowledge. In light of the success of loss modification in class imbalance research [Lin *et al.*, 2017; Menon *et al.*, 2020], we incorporate this seen-unseen prior into the classifier training process in the form of logit adjustment. Specifically, we establish the lower bound of the seen-unseen balanced accuracy [Xian *et al.*, 2017] with variational Bayesian inference and obtain an adjusted posterior. Then the prior takes part in training via adjusting the logits of vanilla cross-entropy loss. This approach, termed Zero-Shot Logit Adjustment (ZLA), allows for a lower number of generated unseen samples while producing more balanced results. By establishing a semantic-prototype mapping, we further introduce the semantic information to the classifier. Notably, our proposed ZLA allows the generated unseen samples to play an adjustment role rather than supplying unseen class information, which overcomes the embedding-based classifier's previous ineffectiveness in the generative scenario (see Sec. 4.5). Our contributions are summarized as follows:

- We mathematically derive the lower bound of the seen-unseen balanced accuracy, allowing us to include generated unseen samples's bias and homogeneity as a seen-unseen prior in cross-entropy via logit adjustment.
- Based on ZLA, we break the previous classifier's inconsistency in training objectives and test metrics and the inability to exploit semantic priors in classifier learning.
- Our proposed classifier enables greatly reducing the generation number of unseen samples. It outperforms SoTAs when combined with the base generator, and can be a plug-in to augment various generation-based methods.

2 Related Work

2.1 Zero-Shot Learning

Zero-Shot Learning (ZSL) [Lampert *et al.*, 2009; Farhadi *et al.*, 2009] has become a popular research area in recent years. Classic ZSL excludes seen classes during the test, while Generalized Zero-Shot Learning (GZSL) [Chao *et al.*, 2016; Xian *et al.*, 2017] considers both seen and unseen classes, attracting more current interest. The typical embedding-based ZSL methods [Li *et al.*, 2019; Skorokhodov and Elhoseiny, 2021] learn the semantic-visual links for classification, but with little effectiveness in the GZSL scenario. The progress of GZSL was once driven by the development of generative models [Kingma and Welling, 2013; Goodfellow *et al.*, 2014], which allowed converting the GZSL problem to common supervised classification using a generator-classifier architecture. Until recently, research on generators [Shen *et al.*, 2020; Han *et al.*, 2021] gradually saturated, whereas classifier design is rarely examined in such a two-stage framework. To break the bottleneck of generation-based approaches, the principle design of a classifier is required.

2.2 Posterior Modification

Posterior modification, which has been deeply studied in class imbalance learning [Lin *et al.*, 2017; Menon *et al.*, 2020], aims at producing a class-balanced prediction. Post-hoc correction [Collell *et al.*, 2016], loss re-weighting[Menon *et al.*, 2013], and logit adjustment [Menon *et al.*, 2020] are its representative strategies. A similar procedure has been adopted in certain ZSL research. DCN [Liu *et al.*, 2018] utilizes entropy regularization to calibrate the predictions of seen and unseen classes. The post-hoc correction, known as calibrated stacking [Chao *et al.*, 2016] in ZSL, is also employed. However, a more general strategy in generation-based GZSL is to sample a large number of unseen class samples from the generated distribution. Although the re-sampling technique [Chawla *et al.*, 2002] has been proven to produce overfitting in long-tail learning [Collell *et al.*, 2016], its shortcoming in generation-based GZSL is a lack of exploration. In this paper, we mathematically introduce the more advanced logit adjustment strategy into GZSL for a better balance between seen and unseen predictions.

3 Methodology

Considering two disjoint label sets, $\mathcal{Y}^s$ and $\mathcal{Y}^u$, GZSL aims at recognizing instances that belong to $\mathcal{Y}^s \cup \mathcal{Y}^u$, while only accessing samples with labels in $\mathcal{Y}^s$ during training. Define the visual space $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ and the semantic set $\mathcal{A} = \{\mathbf{a}_y | y \in \mathcal{Y}^s \cup \mathcal{Y}^u\} \subseteq \mathbb{R}^{d_a}$, where d_x and d_a are dimensions of these two spaces. Then the goal of GZSL is to learn such a classifier, i.e., $f_{gzsl} : \mathcal{X} \rightarrow \mathcal{Y}^s \cup \mathcal{Y}^u$ given the training set $\mathcal{D}^{tr} = \{\mathbf{x}, y | \mathbf{x} \in \mathcal{X}, y \in \mathcal{Y}^s\}$ and the global semantic set $\mathcal{A}$.

The two-stage framework typically processes this problem with two main components: the generator G and the classifier C. The generator G, defined as an arbitrary generative model [Kingma and Welling, 2013; Goodfellow *et al.*, 2014], is first trained with the seen visual features and their corresponding semantics for conditionally mapping the Gaussian noise to fit the real visual distribution. The instances generated by unseen class semantics and the real seen instances are then fed into the classifier C together to fit the posterior probability:

$$\begin{gathered}\hat{\mathbf{x}} = G(\mathbf{z}, \mathbf{a}), \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \\ p(y|\widetilde{\mathbf{x}}) := \mathrm{softmax}[C(\widetilde{\mathbf{x}})], \widetilde{\mathbf{x}} \in \mathcal{X} \cup \{\hat{\mathbf{x}}\},\end{gathered} \tag{1}$$

where $\hat{\mathbf{x}}$ denotes the synthetic instances, $\mathbf{z}$ represents random Gaussian noises, and $\widetilde{\mathbf{x}}$ is either real or synthetic instances. $p(y|\widetilde{\mathbf{x}})$ denotes the posterior probability derived from

Method	G.N.	T1	$\mathcal{A}^U$	$\mathcal{A}^S$	$\mathcal{A}^H$
MSE	588	67.9	17.1	71.6	27.6
	4000	67.5	57.1	59.7	58.4
VAE	588	68.0	25.8	67.6	37.3
	4000	68.6	57.2	68.8	62.9
WGAN	588	68.7	20.9	83.2	33.5
	4000	68.3	57.7	71.0	63.7

Table 1: ZSL (T1) and GZSL ($\mathcal{A}^H$) results of the simple semantic-visual mapping net (denoted as MSE) and two different generative models, VAE and WGAN on AWA2. **G.N.** denotes the generation number per unseen class (588 is the class averaged number of real seen samples).

the classifier. Then the model predicts the class label by taking $\mathcal{C}_{out} = \mathrm{argmax}_y(p(y|\widetilde{\mathbf{x}}))$. In this work, we focus on the design of the classifier under the GZSL setting.

3.1 Preliminary: Logit Adjustment

Logit adjustment strategy is commonly employed in class imbalance tasks [Lin *et al.*, 2017; Menon *et al.*, 2020], which weights the logit in softmax cross-entropy, i.e.,

$$\mathcal{L}_{LA} = \log[1 + \sum_{y' \neq y} \delta(y, y')\exp(\mathcal{C}_{y'}(x) - \mathcal{C}_y(x))], \quad (2)$$

where $\mathcal{C}_{y'}(\cdot)$ is the logit corresponding to class y', and $\delta(y, y')$ represents the adjustment weight. The larger $\delta(y, y')$ results in the network focusing more on optimizing the logit of y', allowing control of the prediction probabilities of different categories. Existing class imbalance works typically associate the weights with the class prior of y or y' [Cao *et al.*, 2019; Tan *et al.*, 2020; Menon *et al.*, 2020].

3.2 Empirical Analysis on Generated Samples

Regardless the bias problem (Figure 1) of generated unseen samples, generation-based methods achieve a certain success in GZSL. Thus, we empirically investigate the working mechanism of the generated unseen samples. As shown in Table 1, we compare the single class center point (semantic to visual-center mapping trained with MSE loss) resampling technique with two generative models, i.e., VAE [Kingma and Welling, 2013] and WGAN [Gulrajani *et al.*, 2017] (detailed in supplementary material). Two phenomena can be intuitively observed by comparing the results in Table 1: (1) the key success of generation-based models in GZSL relies on unseen class feature expression enhancement by a large number of generated unseen samples; and (2) the more diversified samples (generated from generative models) produce a limited performance improvement compared to replicating a single point. We forego a deeper study due to its orthogonal nature to our work, but these modest findings imply that the synthetic unseen samples are rather homogenous than real ones.

Despite the difficulty of eliminating bias and homogeneity, can we include them in classifier training as the seen-unseen prior? Below, we'll illustrate how, by changing the classifier's optimization target, we can integrate this prior into the learning process in the form of the logit adjustment.

3.3 From the Statistical View

The nature of GZSL is an extreme case of class imbalance, as measured by a class balanced metric $\mathcal{A}^H$ (detailed in Sec. 4.1). Assuming the class space has been completed by a set of pseudo unseen class samples generated by the trained generator, existing classifiers optimize the global accuracy $\mathcal{A}^G$ by modeling the base posterior probability (Eq. 1):

$$\mathcal{A}^G = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} q(\mathcal{C}_{out} = y_{\mathbf{x}} | \mathbf{x}), \quad (3)$$

where $p(\mathbf{x})$ is defined as a uniform distribution over all data, $y_{\mathbf{x}}$ is the true label of $\mathbf{x}$, and $q(\mathcal{C}_{out} = y|\mathbf{x})$ is the probability to predict class y with the classifier $\mathcal{C}$. However, Eq. 3 neglects the imbalance between seen and unseen domains, which is inconsistent with the test metric $\mathcal{A}^H$. To find balanced results across classes, we in turn employ evaluation indicators to guide the design of the classifier. Indeed, let $\mathcal{A}(y)$ denote the accuracy in class y, we have [Collell *et al.*, 2016]

$$\mathcal{A}(y) = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \frac{q(\mathcal{C}_{out} = y|\mathbf{x}) p(y|\mathbf{x})}{p(y)}, \quad (4)$$

where $p(y)$ represents the statistics frequency of class y, and $p(y|\mathbf{x})$ denotes the real posterior probability. Then the average accuracy of seen classes is

$$\begin{aligned} \mathcal{A}^S &= \frac{1}{|\mathcal{Y}^s|} \sum_{y \in \mathcal{Y}^s} \mathcal{A}(y) \\ &= \frac{1}{|\mathcal{Y}^s|} \sum_{y \in \mathcal{Y}^s} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \frac{q(\mathcal{C}_{out} = y|\mathbf{x}) p(y|\mathbf{x})}{p(y)} \\ &= \mathbb{E}_{y \in \mathcal{Y}^s} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \frac{q(\mathcal{C}_{out} = y|\mathbf{x}) p(y|\mathbf{x})}{p(\mathcal{Y}^s) p(y|y \in \mathcal{Y}^s)}, \end{aligned} \quad (5)$$

where $p(y|y \in \mathcal{Y}^s)$ denotes the frequency of class y in $\mathcal{Y}^s$, and $p(\mathcal{Y}^s)$ is a theoretically derived data-independent probability which will be served as a hyperparameter. Analogously, the average accuracy of unseen classes is

$$\mathcal{A}^U = \mathbb{E}_{y \in \mathcal{Y}^u} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \frac{q(\mathcal{C}_{out} = y|\mathbf{x}) p(y|\mathbf{x})}{p(\mathcal{Y}^u) p(y|y \in \mathcal{Y}^u)}. \quad (6)$$

Then we consider the harmonic mean, $\mathcal{A}^H$, which serves the target of attaining high accuracy for both seen and unseen classes, and empirically reaches its maximum when the accuracy of seen and unseen classes is balanced [Xian *et al.*, 2017]. We have

$$\mathcal{A}^H = 2/(\frac{1}{\mathcal{A}^S} + \frac{1}{\mathcal{A}^U}). \quad (7)$$

With the convexity of the inversely proportional function, we have the upper bound of $1/\mathcal{A}^S$ with Jensen Inequality:

$$\begin{aligned} \frac{1}{\mathcal{A}^S} &= 1/\mathbb{E}_{y \in \mathcal{Y}^s} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \frac{q(\mathcal{C}_{out} = y|\mathbf{x}) p(y|\mathbf{x})}{p(\mathcal{Y}^s) p(y|y \in \mathcal{Y}^s)} \\ &\leq \mathbb{E}_{y \in \mathcal{Y}^s} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \frac{p(\mathcal{Y}^s) p(y|y \in \mathcal{Y}^s)}{q(\mathcal{C}_{out} = y|\mathbf{x}) p(y|\mathbf{x})}. \end{aligned} \quad (8)$$

Analogously deriving $\mathcal{A}^U$, we get a lower bound of $\mathcal{A}^H$:

$$\mathcal{A}^H \geq 2/\mathbb{E}_{y \in \mathcal{Y}^s \cup \mathcal{Y}^u} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \frac{|\mathcal{Y}^s \cup \mathcal{Y}^u| p(\mathcal{Y}) p(y|y \in \mathcal{Y})}{|\mathcal{Y}| q(\mathcal{C}_{out} = y|\mathbf{x}) p(y|\mathbf{x})}, \quad (9)$$

where $\mathcal{Y}$ is $\mathcal{Y}^s$ ($\mathcal{Y}^u$) when y belongs to seen (unseen) classes. To simplify the symbols, we designate $|\mathcal{Y}^s \cup \mathcal{Y}^u| p(\mathcal{Y})/|\mathcal{Y}|$ as $p_0(\mathcal{Y})$ in the following.

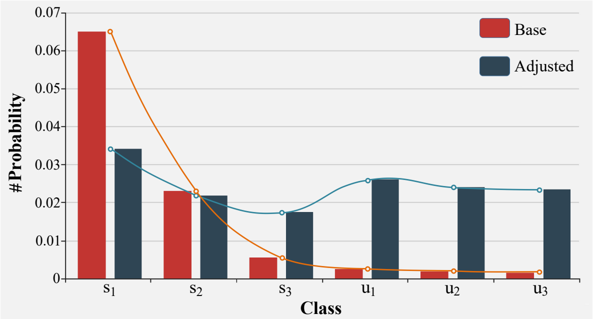

Figure 2: A seen (s) and unseen (u) class-prediction-probability example of modeling the base and the adjusted posteriors. The unseen class probabilities are suppressed when modeling the base posterior $p(y|\mathbf{x})$, while the adjusted posterior $p(y|\mathbf{x})/[p_0(\mathcal{Y})p(y|y\in\mathcal{Y})]$ provides a more balanced distribution.

Despite the difficulty in determining the Bayesian optimal of $\mathcal{A}^H$, maximizing its lower bound achieves an approximate effect, which is equivalent to minimizing its reciprocal. Intuitively, the denominator term of Eq. 9 is minimized if

$$q(\mathcal{C}_{out}=y|\mathbf{x})=\begin{cases}1, \text{ if } y=\text{argmax}_i \frac{p(i|\mathbf{x})}{p_0(\mathcal{Y})p(i|i\in\mathcal{Y})}\\ 0, \text{ otherwise}\end{cases} \quad (10)$$

for each $\mathbf{x}$ in $p(\mathbf{x})$. So, given a datum $(\mathbf{x}, y_\mathbf{x})$, we change the modeling objective in Eq. 1 to the adjusted posterior probability, i.e.,

$$\frac{p(y_\mathbf{x}|\mathbf{x})}{p_0(\mathcal{Y})p(y_\mathbf{x}|y_\mathbf{x}\in\mathcal{Y})}, \quad (11)$$

where $p_0(\mathcal{Y})$ refers to the seen-unseen prior (Sec. 3.2) which reflects the bias and homogeneity of pseudo unseen samples. Eq. 11 theoretically gives a more balanced predicted probability distribution than the base posterior, as shown in Figure 2. Next, we will show the estimation of the adjusted posterior.

3.4 ZLA-Based Classifier

Adjusted Cross-Entropy. The base posterior probability in Eq. 1 is typically estimated with the cross-entropy loss, i.e.,

$$\mathcal{L}_{\text{CE}}=\log\sum_{y'\neq y}[1+\exp(\mathcal{C}_{y'}(x)-\mathcal{C}_y(x))]. \quad (12)$$

Referring to researches on class imbalance [Tan *et al.*, 2020; Menon *et al.*, 2020], we directly model Eq. 11 with $\mathcal{C}(\cdot)$ and the posterior becomes

$$p(y|\mathbf{x}):=p_0(\mathcal{Y})p(y|y\in\mathcal{Y})\cdot\text{softmax}[\mathcal{C}(\mathbf{x})]. \quad (13)$$

This enables integrating the conditional class prior $p(y|y\in\mathcal{Y})$ and the seen-unseen prior $p_0(\mathcal{Y})$ into the softmax cross-entropy in a logit adjustment manner. Then the weights in Eq. 2 are replaced with

$$\delta(y,y'):=\frac{p_0(\mathcal{Y}')p(y'|y'\in\mathcal{Y}')}{p_0(\mathcal{Y})p(y|y\in\mathcal{Y})}. \quad (14)$$

The final adjusted cross-entropy is

$$\mathcal{L}_{ZLA}=\log[1+\sum_{y'\neq y}\frac{p_0(\mathcal{Y}')p(y'|y'\in\mathcal{Y}')}{p_0(\mathcal{Y})p(y|y\in\mathcal{Y})}\exp(\mathcal{C}_{y'}(\mathbf{x})-\mathcal{C}_y(\mathbf{x}))]. \quad (15)$$

In contrast to the standard cross-entropy form, we consider the specific prior information in the generation-based GZSL setting, which contributes to the balancing results across classes. In practice, we make $p_0(\mathcal{Y}^s)$ much bigger than $p_0(\mathcal{Y}^u)$. This is intuitively explainable from two perspectives: first, small $p_0(\mathcal{Y}^u)/p_0(\mathcal{Y}^s)$ promotes seen samples to focus on learning decision boundaries between seen classes; and second, large $p_0(\mathcal{Y}^s)/p_0(\mathcal{Y}^u)$ encourages large prediction probabilities for unseen classes, which serves the same purpose as a large generation number of unseen samples [Xian *et al.*, 2018; Han *et al.*, 2021].

Semantic Prior Inclusion. Embedding-based methods [Li *et al.*, 2019; Skorokhodov and Elhoseiny, 2021] work by learning a semantic-visual direct link. In this case, an extra generator is hard to aid in embedder learning (see Sec. 4.5 for experimental results) because the overlap between the knowledge (i.e., semantic-visual link) learned by the generator and the embedder results in semantic information crucial for knowledge transfer not being fully exploited. The proposed ZLA, in contrast, allows for supporting the learning of the semantic-based classifier through an adjustment mechanism. No longer teaching the net the semantic-visual link of unseen classes, the pseudo unseen samples adjust the decision boundaries between seen and unseen classes by weighting the logits, avoiding knowledge overlapping to an extent. Specifically, we adopt a prototype learner $\mathcal{P}$ [Li *et al.*, 2019; Skorokhodov and Elhoseiny, 2021] to directly map semantics to visual prototypes, and then the adjusted posterior probability of a datum $\mathbf{x}$ is estimated through cosine similarity, i.e.,

$$\mathcal{C}(\mathbf{x}):=\cos(\mathbf{x},\mathcal{P}(\mathbf{a}))/\tau, \quad (16)$$

where τ is the temperature [Hinton *et al.*, 2015]. In the test phase, the prediction class y^* corresponds to the prototype that achieves the maximum similarity.

$$y^*=\underset{i}{\text{argmax}}\frac{p(i|\mathbf{x})}{p_0(\mathcal{Y})p(i|i\in\mathcal{Y})}:=\underset{i}{\text{argmax}}\cos(\mathbf{x},\mathcal{P}(\mathbf{a}_i)). \quad (17)$$

4 Experiments

4.1 Datasets and Metrics

Benchmark Datasets. We study GZSL performed in Animals with Attributes 2 (AWA2) [Lampert *et al.*, 2013], Attribute Pascal and Yahoo (APY) [Farhadi *et al.*, 2009], Caltech-UCSD Birds-200-2011 (CUB) [Wah *et al.*, 2011], and SUN Attribute (SUN) [Patterson and Hays, 2012], following the common split (version 2) proposed in [Xian *et al.*, 2017]. AWA2 includes 50 animal species and 85 attribute annotations, accounting 37,322 samples. APY contains 32 classes of 15,339 samples and 64 attributes. CUB consists of 11,788 samples with 200 bird species, annotated by 312 attributes. SUN carries 14,340 images from 717 different scenario-style with 102 attributes.

Visual Representations and Semantic Descriptors. We follow [Xian *et al.*, 2017] to represent images as the 2048-D ResNet-101 [He *et al.*, 2016] features. Moreover, we regard the artificial attribute annotations that come with the datasets as the semantic descriptors of AWA2, APY, and SUN, and the

	Method	Reference		AWA2			CUB			SUN			APY	
			$\mathcal{A}^U$	$\mathcal{A}^S$	$\mathcal{A}^H$	$\mathcal{A}^U$	$\mathcal{A}^S$	$\mathcal{A}^H$	$\mathcal{A}^U$	$\mathcal{A}^S$	$\mathcal{A}^H$	$\mathcal{A}^U$	$\mathcal{A}^S$	$\mathcal{A}^H$
†	Li et al.	ICCV [Li *et al.*, 2019]	56.4	<u>81.4</u>	66.7	47.4	47.6	47.5	36.3	2.8	39.3	26.5	**74.0**	39.0
	DVBE	CVPR [Xu *et al.*, 2020]	63.6	70.8	67.0	53.2	60.2	56.5	45.0	37.2	40.7	32.6	58.3	41.8
	RGEN	ECCV[Xie *et al.*, 2020]	**67.1**	76.5	71.5	60.0	**73.5**	66.1	44.0	31.7	36.8	30.4	48.1	37.2
	APN	NeurIPS [Min *et al.*, 2020]	56.5	78.0	65.5	65.3	<u>69.3</u>	67.2	41.9	34.0	37.6	-	-	-
‡	Li et al.	AAAI [Li *et al.*, 2021]	56.9	80.2	66.6	51.1	58.2	54.4	47.6	36.6	41.4	-	-	-
	GCM-CF	CVPR [Yue *et al.*, 2021]	60.4	75.1	67.0	61.0	59.7	60.3	47.9	37.8	<u>42.2</u>	37.1	56.8	44.9
	AGZSL	ICLR [Chou *et al.*, 2021]	65.1	78.9	71.3	41.4	49.7	45.2	29.9	**40.2**	34.3	35.1	65.5	45.7
	FREE	ICCV [Chen *et al.*, 2021a]	60.4	75.4	67.1	55.7	59.9	57.7	47.4	37.2	41.7	-	-	-
	SDGZSL	ICCV [Chen *et al.*, 2021b]	64.6	73.6	68.8	59.9	66.4	63.0	-	-	-	38.0	57.4	45.7
	f-CLSWGAN	CVPR [Xian *et al.*, 2018]	57.7	71.0	63.7	59.4	63.3	61.3	46.2	35.2	40.0	32.5	57.2	41.5
	WGAN+**ZLAP**	**Proposed**	<u>65.4</u>	**82.2**	**72.8**	**73.0**	64.8	**68.7**	50.1	<u>38.0</u>	**43.2**	**40.2**	53.8	<u>46.0</u>
	CE-GZSL	CVPR [Han *et al.*, 2021]	65.3	75.0	69.9	66.9	65.9	66.4	**52.4**	34.3	41.5	28.3	<u>65.8</u>	39.6
	CE-GZSL+**ZLAP**	**Proposed**	64.8	80.9	<u>72.0</u>	<u>71.2</u>	66.2	<u>68.6</u>	<u>50.9</u>	35.7	42.0	<u>38.3</u>	60.9	**47.0**

Table 2: GZSL performance comparisons with state-of-the-art methods. $\mathcal{A}^U$ and $\mathcal{A}^S$ denote the per-class accuracy (%) on unseen and seen classes, respectively, and $\mathcal{A}^H$ is their harmonic mean. The best results are bolded, and the underlines indicate the second-place results. † and ‡ represent whether a generator is employed to obtain the pseudo unseen samples, respectively (‡ indicates yes, and † is the opposite). **ZLAP** is the proposed zero-shot logit adjustment prototype learner.

1024-dimensional character-based CNN-RNN features [Reed *et al.*, 2016] generated from textual descriptions as the semantics of CUB.

Evaluation Protocol. We calculate the average per-class top-1 accuracy for unseen and seen classes, respectively, denoted as $\mathcal{A}^U$ and $\mathcal{A}^S$. Then the metric $\mathcal{A}^H$ for GZSL is represented as their harmonic mean.[Xian *et al.*, 2017].

4.2 Implementation Details

Since our work focuses on studying a plug-in classifier, we test it on WGAN [Gulrajani *et al.*, 2017], with the same generator and discriminator structures as [Xian *et al.*, 2018]. Our prototype learner $\mathcal{P}$ consists of a multi-layer perceptron (MLP) with a single 1024-D hidden layer activated by LeakyReLU and no activation function in the output. The Adam optimizer is employed with a learning rate of 1×10^{-3}, and the batch size is set at 512 for evaluating our design. When plugging into CE-GZSL [Han *et al.*, 2021], we employ a batch size of 256 in SUN and 512 in other datasets instead of the default 4096 (due to device limitations) while maintaining all other settings in the published paper.

4.3 Comparison with State-of-the-Arts

We apply the proposed classifier to the vanilla WGAN (compare to f-CLSWGAN [Xian *et al.*, 2018]) and the more advanced CE-GZSL [Han *et al.*, 2021] to show its effect and compatibility in different generative frameworks. The baseline results are obtained from the official codes. As shown in Table 2, the combination of the proposed classifier and the basic generative framework (WGAN) outperforms SoTAs, demonstrating the excellent class balancing capability of ZLA. We note that the performance gain provided by our module is not as significant in the highly fine-grained SUN dataset as it is in other datasets. It is mainly due to the minor bias problem of generated unseen samples in the case of multiple classes and modest feature variations that it takes the limited strengths of our approach. Moreover, our classifier improves the performance of both f-CLSWGAN and CE-GZSL, even though CE-GZSL has already produced decent results, proving its plug-in ability in two-stage frameworks.

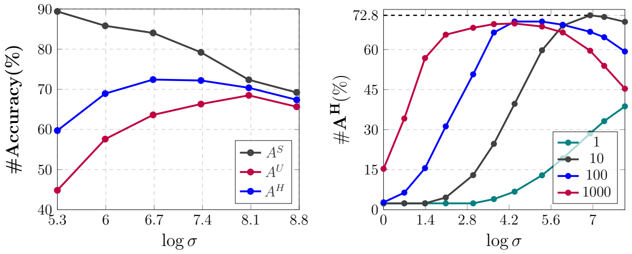

Figure 3: **Left:** hyperparemeter analysis of σ (see 4.4 for its definition) on AWA2. **Right:** $\log \sigma$ varies w.r.t. generation number per unseen class. Large generation number lowers the performance cap (72.8 with 10 generated vs. 69.7 with 1000 generated).

4.4 Hyperparameters

Three key factors are involved in our work, i.e., the generation number (per unseen class) N_g, $p_0(\mathcal{Y})$ in Eq. 11, and the temperature τ in Eq. 16. Following [Skorokhodov and Elhoseiny, 2021], τ is fixed at 0.04 for all experiments, since it has a slight bearing on our study. We first examine $p_0(\mathcal{Y})$ in the form of the ratio of $p_0(\mathcal{Y}^s)$ to $p_0(\mathcal{Y}^u)$, which more intuitively reflects the seen-unseen dichotomy. The ratio is denoted as σ, and its effect is plotted in Figure 3 (left), where a reverse variation of seen and unseen accuracy can be observed, demonstrating ZLA's capacity to moderate the prediction between classes. Figure 3 (right) depicts the influence carried by N_g. Although N_g also posses the ability to affect the accuracy, a large generation number lowers the performance cap (72.8 with 10 generated vs. 69.7 with 1000 generated), indicating the harm of re-sampling the biased samples. In the major experiment (Table 2), we generate 10 samples per unseen class for all datasets to contrast with f-CLSWGAN [Xian *et al.*, 2018], and the best results are obtained when σ is set to 1000, 30, 60, and 300 for AWA2, CUB, SUN, and APY, respectively (better results are possible by trading off between N_g

Method	ReLU	AWA2 $\mathcal{A}^U$	AWA2 $\mathcal{A}^S$	AWA2 $\mathcal{A}^H$	CUB $\mathcal{A}^U$	CUB $\mathcal{A}^S$	CUB $\mathcal{A}^H$
Non Gen.	✓	55.1	82.0	65.9	68.3	54.8	60.8
	×	24.6	**89.9**	38.6	60.5	**66.6**	63.4
Gen.	✓	56.5	81.3	66.7	64.7	64.7	64.7
	×	**65.4**	82.2	**72.8**	**73.0**	64.8	**68.7**

Table 3: Comparison of the pure prototype learner and the generation-based ZLA prototype learner. **Non Gen.:** a latest proposed pure prototype learner, implemented by the official code (with the post-hoc correction removed for a fair comparison). **Gen.:** WGAN-based zero-shot logit adjustment prototype learner.

and σ). In comparison to CE-GZSL [Han *et al.*, 2021], we keep the published generated number and take the value of σ as 28, 9, 7, and 920 for the above datasets.

4.5 Ablation Study

In this section, we perform ablation studies to validate our design and display the key elements in our implementation.

Function of ZLA. Figure 3 (right) shows the function of ZLA, where $\log \sigma = 0$ means $p_0(\mathcal{Y}^s) = p_0(\mathcal{Y}^u)$, i.e., without major adjustments. The effect of ZLA is reflected in the comparison of different σ values. Intuitively, an adequate σ value produces considerable performance gains in varying generation numbers, especially when the number is minimal. Furthermore, ZLA enables the prototype learner to be effective in generative scenarios, as illustrated in Table 4 (second lines in two baselines reflect previous ineffectiveness), which further reduces the reliance on the generation number and thereby resolving the previously discussed bias problem.

Beyond the Standard Prototype Learner. Existing prototype leaners [Li *et al.*, 2019; Skorokhodov and Elhoseiny, 2021] are simply established on the semantic-visual links of seen classes, generalizing to unseen classes based on semantic similarities. In this paper, we find the last ReLU layer is crucial to these approaches' hitherto unseen class performances. To investigate the effect of the ReLU function, we compare the latest pure prototype learner [Skorokhodov and Elhoseiny, 2021] with the WGAN-based ZLAP in Table 3. When the ReLU layer is removed, as shown in Table 3, the seen accuracy improves in both baselines, whereas the unseen accuracy decreases if pseudo unseen samples are not accessible. Zeroing the negative output layer values intuitively affects (seen class) prototype expression. However, it provides a regularization which narrows the gap between the unseen class prototypes and the instances when (pseudo) unseen instances are unavailable in training, since the visual feature components are likewise larger than or equal to zero [Xian *et al.*, 2017]. In this sense, our proposed ZLA allows the model to remove the ReLU layer by adjusting the unseen class expression using the synthetic unseen instances, resulting in a win-win situation for both seen and unseen class accuracy.

Beyond the Vanilla Softmax Classifier. In Table 4, we compare the prototype-based classifier to the vanilla softmax classifier to validate its necessity. Results reveal that the prototype learner beats the vanilla softmax classifier thoroughly when ZLA is employed. The explanations are as follows:

Classifier	ZLA	L.N.	AWA2 $\mathcal{A}^U$	AWA2 $\mathcal{A}^S$	AWA2 $\mathcal{A}^H$	CUB $\mathcal{A}^U$	CUB $\mathcal{A}^S$	CUB $\mathcal{A}^H$
Vanilla	✓	×	40.2	**82.5**	54.1	61.4	52.3	56.5
	×	✓	57.7	71.0	63.7	59.4	63.3	61.3
	✓	✓	61.2	74.6	67.3	66.8	63.5	65.1
Proto.	✓	×	**65.4**	82.2	**72.8**	**73.0**	**64.8**	**68.7**
	×	✓	50.7	75.8	60.8	60.0	63.4	61.4
	✓	✓	64.1	73.1	68.3	72.4	63.0	67.4

Table 4: **Vanilla softmax classifier** vs. **prototype learner**, based on WGAN, where L.N. represents a large generation number.

(1) the prototype learner enables further regularization on unseen class prototypes by learning semantic-prototype relations with real-world data, whereas the vanilla softmax classifier learns to distinguish unseen classes solely by generated samples; and (2) classifier weights mapped from semantics focus more on category-distinctive information, i.e., semantic information. Specifically, in coarse-grained datasets like AWA2, the generated unseen class samples meet a more serious bias problem, causing the classifier to incorrectly detect unseen classes. The prototype learner, on the other hand, can provide supplemental information to unseen class weights by comprehending the semantic-visual links in seen classes. In fine-grained datasets such as CUB, samples from different classes are relatively close together, making it challenging to separate them correctly. The semantics-based classifier, which contains intrinsically category-distinctive information, aids in increased discrimination.

4.6 Time Complexity Analysis

We note that some recent generation-based methods [Han *et al.*, 2021; Chen *et al.*, 2021b] also mine the semantic discriminant information of samples. However, these methods are typically built on class contrast during generator training, which yields a time complexity of $O(N|\mathcal{Y}^s|)$ in the training phase (N is the data size). In comparison, the proposed classifier combined with the vanilla WGAN achieves a comparable result with $O(N)$ time complexity. Moreover, the proposed method allows for a much smaller (10 vs. 4000 in AWA2) synthetic number in the classifier training phase, resulting in further time savings.

5 Conclusion

In this work, we theoretically include the logit adjustment tech in the generation-based GZSL. We begin by examining the bias and homogeneity of the generated unseen samples, which build the seen-unseen prior. Then we derive an adjusted posterior from the seen-unseen balanced metric, which enables integrating the seen-unseen prior into the original cross-entropy via logit adjustment. Considering the zero-shot setting, we call our approach Zero-Shot Logit Adjustment. Based on ZLA, we inject the semantic information into the classifier, which always fails in existing two-stage methods.

Our work explores the underutilized potential of the generation-based GZSL by breaking the previous inconsistency between the classifier's training objectives and testing metrics. This approach allows for greatly fewer generated unseen samples, achieving SoTA with little time consumption.

References

[Buda *et al.*, 2018] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

[Cao *et al.*, 2019] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *NeurIPS*, 2019.

[Chao *et al.*, 2016] Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *ECCV*, pages 52–68, 2016.

[Chawla *et al.*, 2002] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *JAIR*, 16:321–357, 2002.

[Chen *et al.*, 2021a] Shiming Chen, Wenjie Wang, Beihao Xia, Qinmu Peng, Xinge You, Feng Zheng, and Ling Shao. Free: Feature refinement for generalized zero-shot learning. *ICCV*, 2021.

[Chen *et al.*, 2021b] Zhi Chen, Yadan Luo, Ruihong Qiu, Zi Huang, Jingjing Li, and Zheng Zhang. Semantics disentangling for generalized zero-shot learning. In *ICCV*, 2021.

[Chou *et al.*, 2021] Yu-Ying Chou, Hsuan-Tien Lin, and Tyng-Luh Liu. Adaptive and generative zero-shot learning. In *ICLR*, 2021.

[Collell *et al.*, 2016] Guillem Collell, Drazen Prelec, and Kaustubh Patil. Reviving threshold-moving: a simple plug-in bagging ensemble for binary and multiclass imbalanced data. *CoRR*, 2016.

[Farhadi *et al.*, 2009] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *CVPR*, pages 1778–1785, 2009.

[Fu *et al.*, 2014] Yanwei Fu, Timothy M Hospedales, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Transductive multi-view embedding for zero-shot recognition and annotation. In *ECCV*, pages 584–599. Springer, 2014.

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, 27, 2014.

[Gulrajani *et al.*, 2017] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017.

[Han *et al.*, 2021] Zongyan Han, Zhenyong Fu, Shuo Chen, and Jian Yang. Contrastive embedding for generalized zero-shot learning. In *CVPR*, pages 2371–2381, 2021.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[Hinton *et al.*, 2015] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *NeurIPS*, 2015.

[Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *ICLR*, 2013.

[Lampert *et al.*, 2009] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, pages 951–958, 2009.

[Lampert *et al.*, 2013] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE TPAMI*, 36(3):453–465, 2013.

[Li *et al.*, 2019] Kai Li, Martin Renqiang Min, and Yun Fu. Rethinking zero-shot learning: A conditional visual classification perspective. In *ICCV*, pages 3583–3592, 2019.

[Li *et al.*, 2021] Xiangyu Li, Zhe Xu, Kun Wei, and Cheng Deng. Generalized zero-shot learning via disentangled representation. In *AAAI*, volume 35, pages 1966–1974, 2021.

[Lin *et al.*, 2017] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017.

[Liu *et al.*, 2018] Shichen Liu, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Generalized zero-shot learning with deep calibration network. In *NeurIPS*, pages 2005–2015, 2018.

[Menon *et al.*, 2013] Aditya Menon, Harikrishna Narasimhan, Shivani Agarwal, and Sanjay Chawla. On the statistical consistency of algorithms for binary classification under class imbalance. In *ICML*, pages 603–611. PMLR, 2013.

[Menon *et al.*, 2020] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. *arXiv preprint arXiv:2007.07314*, 2020.

[Min *et al.*, 2020] Shaobo Min, Hantao Yao, Hongtao Xie, Chaoqun Wang, Zheng-Jun Zha, and Yongdong Zhang. Domain-aware visual bias eliminating for generalized zero-shot learning. In *CVPR*, pages 12664–12673, 2020.

[Patterson and Hays, 2012] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*, pages 2751–2758, 2012.

[Reed *et al.*, 2016] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *CVPR*, pages 49–58, 2016.

[Shen *et al.*, 2020] Yuming Shen, Jie Qin, Lei Huang, Li Liu, Fan Zhu, and Ling Shao. Invertible zero-shot recognition flows. In *ECCV*, pages 614–631, 2020.

[Skorokhodov and Elhoseiny, 2021] Ivan Skorokhodov and Mohamed Elhoseiny. Class normalization for (continual)? generalized zero-shot learning. *ICLR*, 2021.

[Tan *et al.*, 2020] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *CVPR*, pages 11662–11671, 2020.

[Wah *et al.*, 2011] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. *california institute of technology*, 2011.

[Xian *et al.*, 2017] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *CVPR*, pages 4582–4591, 2017.

[Xian *et al.*, 2018] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *CVPR*, pages 5542–5551, 2018.

[Xie *et al.*, 2020] Guo-Sen Xie, Li Liu, Fan Zhu, Fang Zhao, Zheng Zhang, Yazhou Yao, Jie Qin, and Ling Shao. Region graph embedding network for zero-shot learning. In *ECCV*, pages 562–580. Springer, 2020.

[Xu *et al.*, 2020] Wenjia Xu, Yongqin Xian, Jiuniu Wang, Bernt Schiele, and Zeynep Akata. Attribute prototype network for zero-shot learning. *NeurIPS*, 2020.

[Yue *et al.*, 2021] Zhongqi Yue, Tan Wang, Qianru Sun, Xian-Sheng Hua, and Hanwang Zhang. Counterfactual zero-shot and open-set visual recognition. In *CVPR*, pages 15404–15414, 2021.