

# Bi-directional Distribution Alignment for Transductive Zero-Shot Learning

Zhikai Wang<sup>1</sup>, Yanbin Hao<sup>1\*</sup>, Tingting Mu<sup>2</sup>, Ouxiang Li<sup>1</sup>, Shuo Wang<sup>1</sup>, Xiangnan He<sup>1\*</sup>

<sup>1</sup>University of Science and Technology of China, <sup>2</sup>The University of Manchester

wangzhic@mail.ustc.edu.cn, haoyanbin@hotmail.com, tingtingmu@manchester.ac.uk,

lioox@mail.ustc.edu.cn, {shuowang.hfut, xiangnanhe}@gmail.com

## Abstract

It is well-known that zero-shot learning (ZSL) can suffer severely from the problem of domain shift, where the true and learned data distributions for the unseen classes do not match. Although transductive ZSL (TZSL) attempts to improve this by allowing the use of unlabelled examples from the unseen classes, there is still a high level of distribution shift. We propose a novel TZSL model (named as Bi-VAEGAN), which largely improves the shift by a strengthened distribution alignment between the visual and auxiliary spaces. The key proposal of the model design includes (1) a bi-directional distribution alignment, (2) a simple but effective  $L_2$ -norm based feature normalization approach, and (3) a more sophisticated unseen class prior estimation approach. In benchmark evaluation using four datasets, Bi-VAEGAN achieves the new state of the arts under both the standard and generalized TZSL settings. Code could be found at <https://github.com/Zhikaiwww/Bi-VAEGAN>.

## 1. INTRODUCTION

Zero-shot learning (ZSL) was originally proposed as zero-data learning in computer vision [21], tackling challenging training setups that largely restrict the example and (or) label availability [6]. For instance, in conventional ZSL, no training example is provided for the targeted classes, and they are therefore referred to as the unseen classes. Instead, a large number of training examples paired with their class labels are provided for a different set of classes, which are referred to as the seen classes. This setup is called the inductive ZSL. Its core challenge is to enable the classifier to extract knowledge from the seen classes and transfer it to the unseen classes, assuming the existence of such relevant knowledge [30, 42, 50]. For instance, a ZSL classifier can be constructed to recognize leopard images after feeding it the Felinae images like wildcat, knowing leopard is relevant to Felinae. Information on class relevance is typically provided as auxiliary data, bridg-

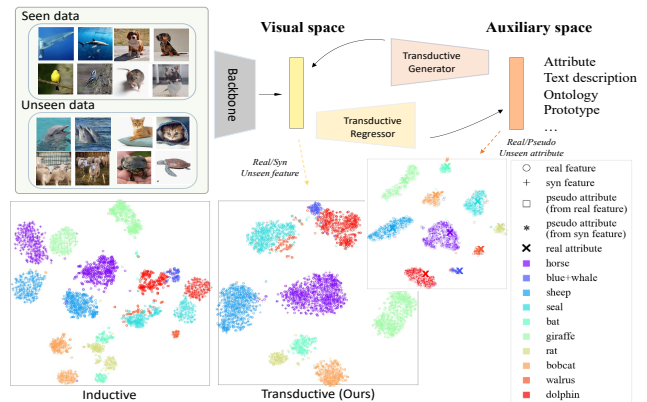


Figure 1. The top figure illustrates the proposed bi-directional generation between the visual and auxiliary spaces. The bottom figure compares the aligned visual space obtained by our method with the unaligned one obtained by inductive ZSL for the unseen classes using the AWA2 data. The bottom right figure shows our approximately aligned attributes in auxiliary space.

ing knowledge transfer from the seen to unseen classes. The auxiliary data can be human-annotated attribute information [38], text description [34], knowledge graph [22] or a formal description of knowledge (e.g., ontology) [15], etc., which are encoded as (set of) embedding vectors. Learning solely from auxiliary data to capture class relevance is challenging, resulting in a discrepancy between the true and modeled distributions for the unseen classes, known as the domain shift problem. To ease the learning, another ZSL setup called transductive ZSL (TZSL) is proposed. It allows to additionally include in the training unlabelled examples collected for the target classes. Since it does not require any extra annotation effort to pair the examples from the unseen classes with their class labels, this setup is still practical in real-world applications.

Benefiting from good example availability, generative models have become popular for synthesizing examples to enhance classifier training [11, 24, 43] and for learning the unseen data distribution [32, 41, 44] under the TZSL setup. Depending on the data availability, they can be formulated

\*Yanbin Hao and Xiangnan He are both the corresponding authors.

as unconditional generation i.e.,  $p(v)$ , or conditional generation i.e.,  $p(v|y)$ . When being conditioned on the auxiliary information, which is a more informative form of class labels, a data-auxiliary (data-label) joint distribution could be learned. If proper supervision is added, intra-class data distribution alignment (with the real data) could be realized, e.g. using a conditional discriminator to discriminate whether the generation is a realistic *wildcat* image. The data-auxiliary joint distribution modeling nature bridges the knowledge between two spaces and enables the generator a proper knowledge transfer tool. As for TZSL, the challenging problem is how to transfer the joint distribution knowledge of seen classes to the unlabelled unseen data and have a realistic generation for the unseen classes. f-VAEGAN [44], a representative generative approach, which enhances the unseen generation using an unconditional discriminator and learns the overall unseen data distribution. This simple strategy turns out to be effective in approximating the conditional distribution of unseen classes. However, the majority of existing works [3,41] only use auxiliary data in the forward generation process, i.e., to generate images from the auxiliary data as by  $p(v|y)$ . It results in weakly guided conditional generation for the unseen classes and the alignment is extremely sensitive to the auxiliary information quality. To bridge better between visual and auxiliary data, particularly for the unseen classes, which is equivalent to enhancing the alignment with the true unseen conditional distribution, we propose a novel bi-directional generation process. It couples the forward generation process with a backward one, i.e., to generate auxiliary data from the images as by  $p(y|x)$ . We build our bi-directional generation based on the feature-generating framework as shown in Figure 1, and name the proposed model Bi-VAEGAN.

Overall, the proposed design covers three aspects. (1) A transductive regressor is added to form the backward generation, synthesizing pseudo auxiliary features conditioned on visual features of an image. This, together with the forward generation as used in f-VAEGAN, provides more constraints to learn the unseen conditional distribution, expecting to achieve better alignments between the visual and auxiliary spaces. (2) We introduce  $L_2$ -feature normalization, a free-lunch data pre-processing method, to further support the conditional alignment. (3) Besides, we note that the (unseen) class prior plays a crucial role in distribution alignment, particularly for those datasets that have extremely unbalanced label distribution. A poor choice of the class prior can easily lead to a poor alignment. To address this issue, we propose a simple but effective class prior estimation approach based on a cluster structure contained by the examples from the unseen classes. The proposed Bi-VAEGAN is compared against various advanced ZSL techniques using four benchmark datasets and achieves a significant TZSL classification accuracy boost compared with the other gen-

erative benchmark models.

## 2. RELATED WORKS

**Inductive Zero-Shot Learning** Previous works on inductive ZSL learn simple projections from the auxiliary (e.g. semantic) to visual spaces to enable knowledge transfer from the seen to the unseen classes [4,7,48,49]. They suffer from the well-known domain shift problem due to the distribution gap between the seen and unseen data. Relation-Net [36] utilizes two embedding modules to align visual and semantic information respectively and compute the relation scores in the embedded space. Generative approaches, e.g., variational auto-encoder (VAE) [10] and generative adversarial network (GAN) [8], synthesize unseen examples and train an additional classifier using the generated examples. This can enhance alignment between synthesized and true unseen domains, although it still suffers from domain shift. Some works, on the other hand, aim to improve by introducing auxiliary modules. For instance, f-CLSWGAN [43] uses a Wasserstein GAN (WGAN) to model the conditional distribution of seen data, while also introducing a classification loss to improve the generation. Cycle-WGAN [12] employs a semantic regressor with a cycle-consistency loss [51] to reconstruct the original semantic features, which provides stronger generation constraints and shares some similar spirit with our work. However, because there is no knowledge of the unseen data, the inductive ZSL heavily relies on the quality of the auxiliary information, making it difficult to solve the performance bottleneck.

**Transductive Zero-Shot Learning** As a concession of inductive ZSL, TZSL allows to use of the test-time unseen data in the training [14,41,45]. One of the most representative approaches is visual structure constraint (VSC) [39] which exploits the cluster structure of the unseen data and proposes to align the projection centers with the cluster centers. Recently, generative models have been actively explored and demonstrate their superiority in TZSL. For instance, f-VAEGAN combines the VAE and GAN, and additionally includes an unconditional discriminator to capture the unseen distribution. SDGN [41] introduces a self-supervised objective for the discriminability mining between seen and unseen domains. STHS-WGAN [3] iteratively adds easily distinguishable unseen classes to the training seen examples to improve the unseen generation. However, previous approaches mostly work with a uni-directional generation from the auxiliary space to the visual one, which could potentially result in limited constraints when learning the unseen distribution. TF-VAEGAN [29] enhances the generated visual features by utilizing an inductive regressor that trained with seen data and a feedback module. Our work is different from TF-VAEGAN as we additionally consider the information exploration for unseen data in the regressor.

### 3. METHOD

#### 3.1. Notations

We use  $V^s = \{v_i^s\}_{i=1}^{n_s}$  and  $V^u = \{v_i^u\}_{i=1}^{n_u}$  to denote the collections of examples from the seen and unseen classes, where each example is characterized by visual feature extracted by a pre-trained network. For examples from the seen classes, their class labels are provided and denoted by  $Y^s = \{y_i\}_{i=1}^{n_s}$ . Attribute (we set as the default choice of auxiliary information) is provided to describe both the seen and unseen classes, represented by vector sets  $A^s = \{a_i^s\}_{i=1}^{N_s}$  and  $A^u = \{a_i^u\}_{i=1}^{N_u}$  where  $N_s$  and  $N_u$  are the number of seen and unseen classes. Under the TZSL setting, a classifier  $f(v) : \mathcal{V}^u \rightarrow \mathcal{Y}^u$  is trained to conduct inference on unseen data, where we use  $\mathcal{V}$  to denote the encoded visual space, and  $\mathcal{Y}$  the label space. The pipeline in the training stage uses the information provided by  $D^{tr} = \{\langle V^s, Y^s \rangle, V^u, \{A^s, A^u\}\}$ , where  $\langle \cdot, \cdot \rangle$  means paired data.

#### 3.2. $L_2$ -Feature Normalization

Feature normalization is an important data preprocessing method, which guarantees the model training and accelerates the model convergence [23]. A common practice in TZSL is to normalize the visual features by the Min-Max approach i.e.,  $v' = \frac{v - \min(v)}{\max(v) - \min(v)}$  [29, 41, 44]. However, we find that it suffers from the distribution skew of synthesized feature value and it is more beneficial to normalize the visual features by their  $L_2$ -norm. For a visual feature vector  $v \in V^s \cup V^u$ , it has

$$v' = L_2(v, r) = \frac{rv}{\|v\|_2}, \quad (1)$$

where the hyperparameter  $r > 0$  controls the norm of a normalized visual feature vector. In the generator, the key modification is to replace the last *sigmoid* layer accompanying the Min-Max approach with an  $L_2$ -normalization layer. We discuss further its effect in Section 4.3.

#### 3.3. Bi-directional Alignment Model

We propose a modular model architecture composed of six modules: (1) a conditional VAE encoder  $E : \mathcal{V} \times \mathcal{A} \rightarrow \mathbb{R}^k$  mapping the visual features to a  $k$ -dimensional hidden representation vector conditioned on the class attributes, (2) a conditional visual generator  $G : \mathcal{A} \times \mathbb{R}^k \rightarrow \mathcal{V}$  synthesizing visual features from a  $k$ -dimensional random vector that is usually sampled from a normal distribution  $\mathcal{N}(\mathbf{0}, \mathbf{1})$ , conditioned on the class attributes, (3) a conditional visual Wasserstein GAN (WGAN) critic  $D : \mathcal{V}^s \times \mathcal{A} \rightarrow \mathbb{R}$  for the seen classes, (4) a visual WGAN critic  $D^u : \mathcal{V}^u \rightarrow \mathbb{R}$  for the unseen classes, (5) a regressor mapping the visual space to the attribute space  $R : \mathcal{V} \rightarrow \mathcal{A}$ , and (6) an attribute WGAN critic  $D^a : \mathcal{A} \rightarrow \mathbb{R}$ .

The proposed workflow consists of two levels. In **Level-1**, the regressor  $R$  is adversarially trained using the critic  $D^a$  so that the pseudo attributes converted from the visual features align with the true attributes. In **Level-2**, the visual generator  $G$  is adversarially trained using the two critics  $D$  and  $D^u$  so that the generated visual features align with the true visual features. Additionally, the training of  $G$  depends on the regressor  $R$ . This encourages the pseudo attributes converted from the synthesized visual features align better with the true attributes. To highlight the core innovation of aligning true and fake data in both the visual and attribute spaces, we name the proposal as bi-directional alignment.

##### 3.3.1 Level 1: Regressor Training

The regressor  $R$  is trained transductively and adversarially. It is constructed by performing supervised learning using the labeled examples from the seen classes, additionally enhanced by unsupervised learning from the visual features and class attributes of the unseen classes. By “unsupervised”, we mean that the features and classes are unpaired for examples from the unseen classes. For each example from the seen classes,  $R$  learns to map its visual features close to its corresponding class attributes via minimizing

$$L_R^s(\mathcal{A}^s, \mathcal{V}^s) = \mathbb{E}[\|R(v^s) - a^s\|_1]. \quad (2)$$

Simultaneously, for examples from the unseen classes, it learns to distinguish their true attributes from the pseudo ones computed from the real unseen visual features by maximizing the adversary objective

$$L_{D^a\text{-WGAN}}^u(\mathcal{A}^u, \mathcal{V}^u) = \mathbb{E}[D^a(a^u)] - \mathbb{E}[D^a(\hat{a}^u)] + \mathbb{E}[(\|\nabla_{\hat{a}^u} D^a(\hat{a}^u)\|_2 - 1)^2], \quad (3)$$

where  $\hat{a}^u = R(v^u)$ ,  $a^u \sim p_G^u(y)$  and  $\bar{a}^u \sim \mathcal{P}_t(a^u, \hat{a}^u)$ <sup>1</sup>. Note that the original attribute vectors are sampled from the unseen class prior  $p_G^u(y)$  explained in Section 3.3.3, and we refer to this as the **prior sample** process. The third term in Eq. (3) is known as the gradient penalty [16], which enables the Lipschitz restriction in the original WGAN [2].

The regressor  $R$  aims at learning a mapping from the visual to the attribute features for the seen classes in a supervised manner, and meanwhile learning the distribution of the overall feature domain for the unseen classes in an unsupervised manner. The level-1 training is formulated by

$$\min_R \max_{D^a} L_R^s + \lambda L_{D^a\text{-WGAN}}^u, \quad (4)$$

where  $\lambda$  is a hyper-parameter. It enables knowledge transfer from the seen to unseen classes in the attribute space, where

<sup>1</sup> $\mathcal{P}_t(a, b)$  is an interpolated distribution in the  $L_2$  hypersphere. An example sampled from this distribution is computed by  $c = L_2(ta + (1-t)b, r)$  with  $t \sim \mathcal{U}(0, 1)$  where  $\|a\|_2 = \|b\|_2 = r$ .

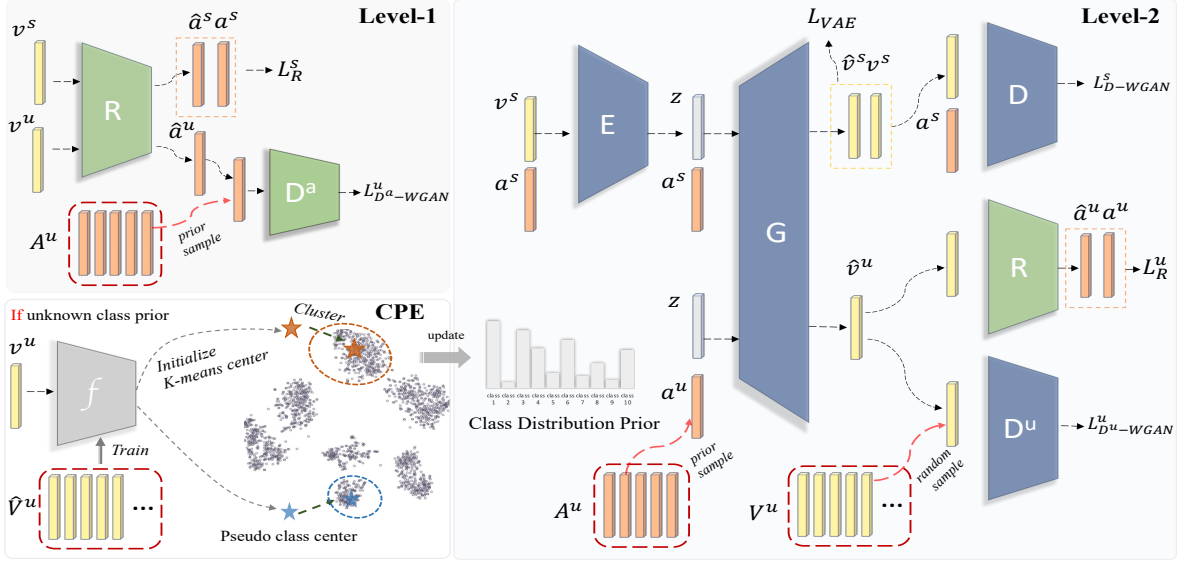


Figure 2. The proposed Bi-VAEGAN model architecture under the TZSL setup.

the feature discriminability is however limited by the hubness problem [49]. But it serves as a good auxiliary module to provide “approximate supervision” for the unseen distribution alignment later in the visual space.

### 3.3.2 Level 2: Generator and Encoder Training

The generator  $G$  is also trained transductively and adversarially. It aims at aligning the synthesized and true features, using the visual critics  $D$  and  $D^u$  in the visual space, while using the frozen regressor  $R$  in the attribute space.

The two visual critics are trained to get better at distinguishing the true visual features from the synthesized ones computed by the conditional generator, i.e.  $\hat{v} \sim G(a, z)$  where  $z \sim \mathcal{N}(\mathbf{0}, 1)$  and  $a \sim p_G(y)$ . For the seen classes, their class prior, denoted by  $p_G^s(y)$ <sup>2</sup>, is simply estimated from the number of examples collected for each class. For the unseen classes, the estimated class prior  $p_G^u(y)$  is used. The synthesized visual feature  $\hat{v}$  is already  $L_2$ -normalized. For the seen classes, the critic is conditioned on their class attributes, i.e.,  $D(\hat{v}^s, a^s)$ , while for the unseen classes, the critic is unconditional, i.e.,  $D^u(\hat{v}^u)$ . The two critics  $D$  and  $D^u$  are trained based on the adversarial objectives:

$$L_{D\text{-WGAN}}^s(A^s, V^s) = \mathbb{E}[D(v^s, a^s)] - \mathbb{E}[D(\hat{v}^s, a^s)] + \mathbb{E}[(\|\nabla_{\bar{v}^s} D(\bar{v}^s, a^s)\|_2 - 1)^2], \quad (5)$$

and

$$L_{D^u\text{-WGAN}}^u(A^u, V^u) = \mathbb{E}[D^u(v^u)] - \mathbb{E}[D^u(\hat{v}^u)] + \mathbb{E}[(\|\nabla_{\bar{v}^u} D^u(\bar{v}^u)\|_2 - 1)^2], \quad (6)$$

<sup>2</sup>For the intra-class alignment, i.e. we know the paired seen labels, the choice of  $p_G^s(y)$  will not affect much of the training.

where  $\bar{v}^s$  and  $\bar{v}^u$  are sampled from the interpolated distribution as explained in footnote 1. Here,  $\hat{v}^u$  is computed from the unseen attributes sampled by  $a^u \sim p_G^u(y)$ . The critic  $D^u$  in Eq. (6) captures the Earth-Mover distance over the unseen data distribution.

Eqs (5) and (6) weakly align the conditional distribution of the unseen classes. It suffers from the absence of any supervised constraints. To further strengthen the alignment, we introduce another training loss, as

$$L_R^u(A^u) = \mathbb{E}[\|\mathbf{R}(G(a^u, z)) - a^u\|_1]. \quad (7)$$

It employs the regressor  $R$  trained in level-1 to enforce supervised constraints. As shown in f-VAEGAN, feature-VAE has the potential of preventing model collapse, and it could serve as a suitable complement to GAN training. Similarly, we adopt the VAE objective function to enhance the adversarial training over the seen classes:

$$L_{\text{VAE}}^s(A^s, V^s) = \mathbb{E}[\text{KL}(\mathbf{E}(v^s, a^s) \|\mathcal{N}(\mathbf{0}, 1)))] + \mathbb{E}_{z^s \sim \mathbf{E}(v^s, a^s)}[(\|\mathbf{G}(a^s, z^s) - v^s\|_2^2)]. \quad (8)$$

The first term is the Kullback-Leibler divergence and the second is the mean-squared-error (MSE) reconstruction loss using the  $L_2$  normalized feature. Finally, the level-2 training is formulated by,

$$\min_{E, G} \max_{D, D^u} L_{\text{VAE}}^s + \alpha L_{D\text{-WGAN}}^s + \beta L_R^u + \gamma L_{D^u\text{-WGAN}}^u, \quad (9)$$

where  $\alpha$ ,  $\beta$  and  $\gamma$  are hyper-parameters. It transfers the knowledge of *paired visual features and attributes* of the seen classes and the estimated *class prior* of the unseen classes, and is enhanced by the attribute regressor  $R$  to

constrain further the visual feature generation for unseen classes. The proposed Bi-VAEGAN architecture can be easily modified to accommodate the inductive ZSL setup, by removing all the loss functions using the unseen data  $V^u$ :

$$\begin{aligned} \text{For level-1: } & \min_{\mathbf{R}} L_R^s \\ \text{For level-2: } & \min_{E, \mathbf{G}} \max_D L_{\text{VAE}}^s + \alpha L_{D\text{-WGAN}}^s + \beta L_R^u. \end{aligned} \quad (10)$$

### 3.3.3 Unseen Class Prior Estimation

When training based on the objective functions in Eqs. (3) and (6), the attributes for the unseen classes are sampled from the class prior:  $\mathbf{a}^u \sim p_{\mathbf{G}}^u(y)$ . Since there is no label information provided for the unseen classes, it is not possible to sample from the real class prior  $p^u(y)$ . An alternative way to estimate  $p_{\mathbf{G}}^u(y)$  is needed. We have observed that examples from the unseen classes possess fairly separable cluster structures in the visual space, and benefited from the strong backbone network. Therefore, we propose to estimate the unseen class prior based on such cluster structure. We employ the K-means clustering algorithm and carefully design the initialization of its cluster centers since the prior estimation is extremely sensitive to the initialization. The estimated prior  $p_{\mathbf{G}}^u(y)$  is iteratively updated and in each epoch cluster centers are re-initialized by pseudo class centers calculated from an extra classifier  $f$ . For the very first estimation of  $p_{\mathbf{G}}^u(y)$ , rather than use the naive but sometimes (if it differs greatly from the real prior) harmful uniform prior assumption, we use the inductively trained generator to transfer the seen paired knowledge to have a mild estimation for the unseen. We refer to the estimation approach as the cluster prior estimation (CPE), and its implementation is shown in Algorithm 1 (lines 1-12).

**Discussion:** We attempt to explain the importance of estimating  $p_{\mathbf{G}}^u(y)$  based on the following corollary, which is a natural result of Theorem 3.4 of [9].

**Corollary 3.1.** *Under the generative TZSL setup, for the unseen classes, the total variation distance between the true conditional visual feature distribution  $p^u(\mathbf{v}|y)$  and the estimated one by the generator  $p_{\mathbf{G}}^u(\hat{\mathbf{v}}|y)$  is upper bounded by*

$$\begin{aligned} & \max_{y \in Y^u} d_{\text{TV}}(p_{\mathbf{G}}^u(\hat{\mathbf{v}}|y), p^u(\mathbf{v}|y)) \\ & \leq \frac{1}{\min_{y \in Y^u} p^u(y)} \left( \max_{y \in Y^u} \left( \frac{p^u(y)}{p_{\mathbf{G}}^u(y)} \right) e_f^u(\hat{\mathbf{v}}) + e_f^u(\mathbf{v}) \right. \\ & \quad \left. + \sqrt{8d_{\text{JS}} \left( \sum_{y \in Y^u} p^u(y) p_{\mathbf{G}}^u(\hat{\mathbf{v}}|y), p^u(\mathbf{v}) \right)} \right), \end{aligned} \quad (11)$$

where  $d_{\text{JS}}(\cdot, \cdot)$  is the Jensen-Shanon divergence between two distributions, and  $e_f^u(\mathbf{x})$  denotes the error probability

---

#### Algorithm 1: Bi-VAEGAN (CPE)

---

**Input** :  $\langle V^s, Y^s \rangle, V^u, \{A^s, A^u\}$ , unseen class number  $N_u$ , epoch numbers  $T_1$  and  $T_2$ .

**Output** :  $E, \mathbf{G}, \mathbf{R}, D, D^u, D^a$ .

```

1 for  $i = 1$  to  $T_1$  do
2   | Inductive training with  $\langle V^s, Y^s \rangle, \{A^s, A^u\}$  by
3   | Eq. (10);
4 end
5 for  $i = 1$  to  $T_2$  do
6   | Define uniform distribution label set  $Y_{\mathbf{G}}^u$ ;
7   | Synthesize paired unseen set  $\langle \hat{V}_{\mathbf{G}}^u, Y_{\mathbf{G}}^u \rangle$  using  $\mathbf{G}$ ;
8   | Train a classifier  $f$  using  $\langle \hat{V}_{\mathbf{G}}^u, Y_{\mathbf{G}}^u \rangle$ ;
9   | Assign pseudo class labels by  $\hat{Y}^u = f(V^u)$ ;
10  | Pseudo class centers  $C^u \leftarrow \langle V^u, \hat{Y}^u \rangle$ ;
11  |  $\hat{Y}_{kmeans}^u = Kmeans(V^u, N_u, InitCenter = C^u)$ ;
12  |  $p_{\mathbf{G}}^u(y) \leftarrow \hat{Y}_{kmeans}^u$ ; /* Update prior */
13  | Transductive training with  $\langle V^s, Y^s \rangle, V^u$ ,
    |  $\{A^s, A^u\}$  and  $p_{\mathbf{G}}^u(y)$  by Eqs (4) and (9);
14 end

```

---

that the classification of the input feature vector disagrees with its ground truth using hypothesis  $f$ .

The above result is a straightforward application of the domain adaptation result in Theorem 3.4 of [9], obtained by treating  $p_{\mathbf{G}}^u(\hat{\mathbf{v}}|y)$  as the source domain distribution while  $p^u(\mathbf{v}|y)$  as the target domain distribution. When the estimated and ground truth class priors are equal, i.e.,  $p^u(y) = p_{\mathbf{G}}^u(y)$ , Eq. (11) reduces to

$$\begin{aligned} & \max_{y \in Y^u} d_{\text{TV}}(p_{\mathbf{G}}^u(\hat{\mathbf{v}}|y), p^u(\mathbf{v}|y)) \\ & \leq \frac{1}{\gamma} \left( e_f^u(\hat{\mathbf{v}}) + e_f^u(\mathbf{v}) + \sqrt{8d_{\text{JS}}(p_{\mathbf{G}}^u(\hat{\mathbf{v}}), p^u(\mathbf{v}))} \right), \end{aligned} \quad (12)$$

where  $\gamma = \min_{y \in Y^u} p^u(y)$ . The effect of the class information is completely removed from the bound. As a result, the success of the conditional distribution alignment is dominated by the unconditional distribution matching ( $D^u$ ). This is important for our model because of the unsupervised learning nature for the unseen classes.

### 3.3.4 Predictive Model and Feature Augmentation

After completing the training of the six modules, a predictive model for classifying unseen examples is trained. This results in a classifier  $f: \mathcal{V}^u(\hat{\mathcal{V}}^u) \times \mathcal{H}^u \times \hat{\mathcal{A}}^u \rightarrow \mathcal{Y}^u$  working in an augmented multi-modal feature space [29]. Specifically, the used feature vector  $\mathbf{x}^u$  concatenates the visual features  $\mathbf{v}^u$  (or  $\hat{\mathbf{v}}^u$ ), the pseudo attribute vector computed by the regressor  $\hat{\mathbf{a}}^u = \mathbf{R}(\mathbf{v}^u)$  and the hidden representation vector  $\mathbf{h}^u$  returned by the first fully-connected layer of regressor, which gives  $\mathbf{x}^u = [\mathbf{v}^u, \mathbf{h}^u, \hat{\mathbf{a}}^u]$ . It integrates knowledge of the generator and regressor that is both transductive, and presents stronger discriminability.

| Method |                             | Zero-Shot Learning            |             |             |                   | Generalized Zero-Shot Learning |             |             |             |             |             |             |             |             |             |             |             |             |     |  |  |
|--------|-----------------------------|-------------------------------|-------------|-------------|-------------------|--------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-----|--|--|
|        |                             | AWA1                          |             | AWA2        |                   | CUB                            |             | SUN         |             | AWA1        |             |             | AWA2        |             |             | CUB         |             |             | SUN |  |  |
|        |                             | T1                            | U           | T1          | U                 | S                              | U           | H           | S           | U           | H           | S           | U           | H           | S           | U           | H           |             |     |  |  |
| I      | F-CLSWGAN [43]              | 59.9                          | 62.5        | 58.1        | 54.9              | 76.1                           | 16.8        | 27.5        | 81.8        | 14.0        | 23.9        | 33.1        | 21.8        | 26.3        | 63.8        | 23.7        | 34.4        |             |     |  |  |
|        | SP-AEN [5]                  | -                             | 58.5        | 59.2        | 55.4              | -                              | -           | -           | <u>90.9</u> | 23.3        | 37.1        | 38.6        | 24.9        | 30.3        | <b>70.6</b> | 34.7        | 46.6        |             |     |  |  |
|        | DEM [49]                    | 68.4                          | 67.2        | 61.9        | 51.7              | 32.8                           | 84.7        | 47.3        | 86.4        | 30.5        | 45.1        | 25.6        | 34.3        | 20.5        | 54.0        | 19.6        | 13.6        |             |     |  |  |
|        | ALE [1]                     | 68.2                          | -           | 60.8        | 57.3              | 61.4                           | 57.9        | 59.6        | 68.9        | 52.1        | 59.4        | 36.6        | 42.6        | 39.4        | 57.7        | 43.7        | 49.7        |             |     |  |  |
|        | LisGAN [24]                 | 70.6                          | -           | 61.7        | 58.8              | 76.3                           | 52.6        | 62.3        | -           | -           | -           | 37.8        | 42.9        | 40.2        | 57.9        | 46.5        | 51.6        |             |     |  |  |
| T      | GMN [35]                    | 82.5                          | -           | 64.6        | 64.3              | 79.2                           | 70.8        | 74.8        | -           | -           | -           | 70.6        | 60.2        | 65.0        | 40.7        | 57.1        | 47.5        |             |     |  |  |
|        | DSRL [46]                   | 74.7                          | 72.8        | 56.8        | 48.7              | 74.7                           | 20.8        | 32.6        | -           | -           | -           | 25.0        | 17.7        | 20.7        | 39.0        | 17.3        | 24.0        |             |     |  |  |
|        | GFZSL [37]                  | 48.1                          | 78.6        | 50.0        | 64.0              | 67.2                           | 31.7        | 43.1        | -           | -           | -           | 45.8        | 24.9        | 32.2        | -           | -           | -           |             |     |  |  |
|        | ALE.trans [1]               | -                             | 70.7        | 54.5        | 55.7              | -                              | -           | -           | 73.0        | 12.6        | 21.5        | 45.1        | 23.5        | 30.9        | 22.6        | 19.9        | 21.2        |             |     |  |  |
|        | PREN [47]                   | -                             | 78.6        | 66.4        | 62.8              | -                              | -           | -           | 88.6        | 32.4        | 47.4        | 55.8        | 35.2        | 43.1        | 27.2        | 35.4        | 30.8        |             |     |  |  |
|        | f-VAEGAN <sup>†</sup> [44]  | -                             | 89.8        | 71.1/74.2*  | 70.1              | -                              | -           | -           | 88.6        | 84.8        | 86.7        | 65.1        | 61.4        | 63.2        | 41.9        | 60.6        | 49.6        |             |     |  |  |
|        | SABR-T <sup>†</sup> [32]    | -                             | 88.9        | 74.0        | 67.5              | -                              | -           | -           | <b>91.0</b> | 79.7        | 85.0        | <b>73.7</b> | 67.2        | 70.3        | 41.5        | 58.8        | 48.6        |             |     |  |  |
|        | TF-VAEGAN <sup>†</sup> [29] | -                             | 92.6        | 74.7/77.2*  | <u>70.9</u>       | -                              | -           | -           | 89.6        | 87.3        | 88.4        | <u>72.1</u> | <u>69.9</u> | <u>71.0</u> | 47.1        | <u>62.4</u> | <u>53.7</u> |             |     |  |  |
|        | GXE [25]                    | 89.8                          | 83.2        | 61.3        | 63.5              | <b>89.0</b>                    | <u>87.7</u> | <u>88.4</u> | 90.0        | 80.2        | 84.8        | 68.7        | 57.0        | 62.3        | 58.1        | 45.4        | 51.0        |             |     |  |  |
|        | LSA <sup>†</sup> [17]       | -                             | 92.8        | - /80.6*    | 71.7              | -                              | -           | -           | 86.7        | 88.5        | 87.6        | -           | -           | -           | <u>59.5</u> | 46.0        | 51.8        |             |     |  |  |
|        | SDGN <sup>†</sup> [41]      | <u>92.3</u>                   | 93.4        | 74.9        | 68.4              | 88.1                           | 87.3        | 87.7        | 89.3        | <u>88.8</u> | <u>89.1</u> | 70.2        | <u>69.9</u> | 70.1        | 46.0        | 62.0        | 52.8        |             |     |  |  |
|        | STHS-WGAN <sup>†</sup> [3]  | -                             | <u>94.9</u> | <u>77.4</u> | 67.5              | -                              | -           | -           | -           | -           | -           | -           | -           | -           | -           | -           | -           | -           |     |  |  |
|        | T                           | Bi-VAEGAN <sup>†</sup> (ours) | <b>93.9</b> | <b>95.8</b> | <u>76.8/82.8*</u> | <b>74.2</b>                    | <u>88.3</u> | <b>89.8</b> | <b>89.1</b> | <b>91.0</b> | <b>90.0</b> | <b>90.4</b> | 71.7        | <b>71.2</b> | <b>71.5</b> | 45.4        | <b>66.8</b> | <b>54.1</b> |     |  |  |

Table 1. TZSL performance comparison where the unseen class prior is provided when needed. “†” denotes the method that adopts the known unseen class prior assumption. “\*” denotes the result is obtained using fine-grained visual descriptions (AK2 in Section 4.3) for CUB and the competing results marked by “\*” are cited from [17].

## 4. EXPERIMENT

We conduct experiments using four datasets, including AWA1 [20], AWA2 [42], CUB [40] and SUN [31]. Visual features are extracted by the pretrained ResNet-101 [18]. Details on the datasets and implementation details are provided in the supplementary material. We conduct experiments at the feature level following [29, 42]. Under the TZSL setup, testing performance of the unseen classes is of interest, for which the average per class top-1 (T1) accuracy is used, denoted by  $ACC^u$  (U). The ZSL community is also interested in the generalized TZSL performance, i.e., testing performance for both the seen and unseen classes [13, 19, 27, 33]. We use the harmonic mean of the average per class top-1 accuracies of the seen and unseen classes to assess it, as  $H = \frac{2ACC^u \times ACC^s}{ACC^u + ACC^s}$ . Also, all existing results reported in the tables come from their published papers, and “-” means such result is missing.

### 4.1. Comparison and Result Analysis

#### 4.1.1 Known Unseen Class Prior

We compare performance with both inductive (I) and transductive (T) state of the arts under the same setting for fair comparison. For TZSL approaches, when class prior of the unseen classes is required, the compared existing techniques assume such information is provided. Therefore, we first apply the same setting for the proposed Bi-VAEGAN. Table 1 reports the results. To distinguish from our later results obtained by the proposed prior estimation approach, methods using the provided unseen prior are marked by “†”. Note that we do not report the generalized TZSL performance for STHS-WGAN here. This is because it uses a harder evaluation setting different from the other ap-

proaches by assuming that the unseen and seen data are indistinguishable during training.

It can be observed from Table 1 that in general, the transductive approaches outperform the inductive approaches with a large gap. The proposed Bi-VAEGAN outperforms the transductive state of the arts, particularly the two baseline frameworks F-VAEGAN and TF-VAEGAN that Bi-VAEGAN is built on, on almost all the datasets. The new state-of-the-art performance that we have achieved is 93.9% (AWA1), 95.8% (AWA2), and 74.2% (SUN) for TZSL, while 89.1% (AWA1), 90.4% (AWA2), and 54.1% (SUN) for generalized TZSL. Note that for the CUB dataset that has less intra-class clustering property, we find a simple feature pre-tuning network will further boost the performance from 76.8% to 78.0% and we put the discussion in the supplementary. It is worth mentioning that for the SUN dataset that is intra-class sample scarce, Bi-VAEGAN achieves satisfactory performance. Learning from the SUN dataset is challenging. This is because the low sample number of each class inherently makes the conditional generation less discriminative. Bi-VAEGAN provides more discriminative features benefitting from its bidirectional alignment generation and feature augmentation in Section 3.3.4.

#### 4.1.2 Unknown Unseen Class Prior

In this experiment, the unseen class prior is not provided. We compare the proposed prior estimation approach for our method with a naive assumption of uniform class prior and a different approach that treats the prior estimation as a label shift problem and solves it by the black box shift estimation (BBSE) method [28]. BBSE builds upon the strong assumption that  $p_G^u(\hat{v} | y) = p^u(v | y)$ , while our CPE assumes the cluster structure plays an important role in class prior

| Method                                  | AWA1        | AWA2        | CUB         | SUN         |
|---|-------------|-------------|-------------|-------------|
| <i>Non-generative</i>                   |             |             |             |             |
| DSRL [46]                               | 74.7        | 72.8        | 56.8        | 48.7        |
| GXE [25]                                | 89.8        | 83.2        | 61.3        | 63.5        |
| VSC [39]                                | -           | 81.7        | 71.0        | 62.2        |
| <i>Generative with uniform prior</i>    |             |             |             |             |
| f-VAEGAN <sup>‡</sup> [44]              | 62.1        | 56.5        | 72.1        | 69.8        |
| TF-VAEGAN <sup>‡</sup> [29]             | 63.0        | 58.6        | 74.5        | 71.1        |
| Bi-VAEGAN                               | 66.3        | 60.3        | <b>76.8</b> | <b>74.2</b> |
| <i>Generative with prior estimation</i> |             |             |             |             |
| Bi-VAEGAN (BBSE)                        | 90.9        | 83.1        | 72.5        | 68.4        |
| Bi-VAEGAN (CPE)                         | <b>91.5</b> | <b>85.6</b> | 74.0        | 71.3        |

Table 2. Performance comparison in ACC<sup>u</sup> for both generative and non-generative techniques using different unseen class priors. ‘<sup>‡</sup>’ means our reproduced result.

estimation. Details on BBSE estimation are provided in the supplementary material. In Table 2, we compare our results with the existing ones, where, for methods that need unseen class prior, a uniform prior is used. By comparing Table 2 with Table 1 for the generative methods, it can be seen that, when the used unseen class prior differs significantly from the one computed from the real class sizes, there are significant performance drops, e.g., over 30% for the extremely unbalanced AWA2 dataset. Both BBSE and CPE could provide a satisfactory prior estimation, while our CPE demonstrates consistently better performance. It can be seen from Figure 3 that the CPE prior and the one computed from the real class sizes match pretty well for most classes, and for both the unbalanced and balanced datasets.

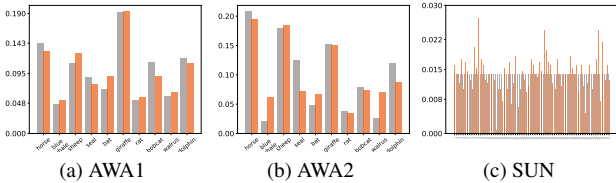


Figure 3. Comparison between the estimated unseen class distribution prior by CPE (orange) and the provided prior computed from the class sizes (gray).

## 4.2. Ablation Study

We perform an ablation study to examine the effect of the proposed  $L_2$ -feature normalization (FN), the use of an inductive regressor (R) that is trained only with the paired seen data, and the use of the adversarial module ( $D^a$ ) in the level-1 transductive regressor training. We implement a Min-Max normalized f-VAEGAN as the baseline. Results are reported in Table 3. One observation is that FN and  $D^a$  consistently improve performance over four datasets, respectively. We conclude the following from Table 3: (1) The  $L_2$ -feature normalization is a free-lunch setting, requiring minimal effort but resulting in a satisfactory performance gain. (2) A naive implementation of the inductive

| Method |   |       | AWA2              | CUB               | SUN               |
|--------|---|-------|-------------------|-------------------|-------------------|
| FN     | R | $D^a$ |                   |                   |                   |
| ✗      | ✗ | ✗     | 91.6              | 72.1              | 69.8              |
| ✗      | ✓ | ✗     | 92.3(+0.7)        | 74.3(+2.1)        | 70.8(+1.0)        |
| ✗      | ✓ | ✓     | 95.5(+3.2)        | 75.8(+1.5)        | 72.2(+1.4)        |
| ✓      | ✓ | ✓     | <b>95.8(+0.3)</b> | <b>76.8(+1.0)</b> | <b>74.2(+2.0)</b> |

Table 3. Ablation study of transductive ZSL results.

regressor is beneficial but somehow limited. The regressor trained solely with the seen attributes provides weak constraints to the unseen generation. (3) The adversarially ( $D^a$ ) trained regressor integrates the unseen attribute information and exhibits superiority in the bi-directional synthesis.

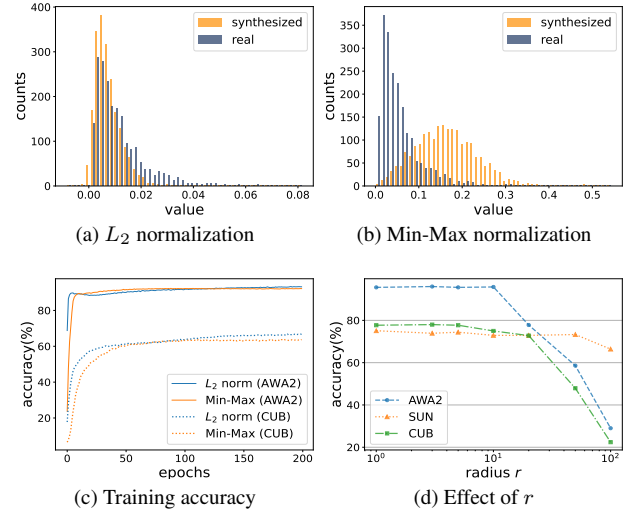


Figure 4. (a) and (b) compare the real and synthesized feature value distributions of AWA2 after  $L_2$  and Min-Max normalizations, respectively. (c) compares the TZSL performance observed during the training for the two normalization approaches using the simplified model on AWA2 and CUB. (d) displays the TZSL performance for different radius values used in  $L_2$ -normalization.

## 4.3. Further Examinations and Discussion

**On  $L_2$ -Feature Normalization** The difference between using the proposed  $L_2$ -normalization and the standard Min-Max normalization is the use of Eq. (1) or the sigmoid activation in the last normalization layer of the generator  $G$ . To demonstrate the difference between the two approaches, we perform a simple experiment, where the network structure is compressed to contain only three core modules  $G$ ,  $D$ , and  $D^a$ . The value distributions of real and synthesized visual features after two normalizations are compared in Figures 4a and 4b. It can be seen that the  $L_2$ -normalization aligns better with the two feature value distributions, while Min-Max results in a quite significant gap between the two. We investigate further the two approaches by looking into their partial derivatives with respect to each dimension, i.e., for the  $L_2$  norm,  $\frac{dv'_i}{dv_i} = \frac{r}{\|v_i\|_2}$ , and for the Min-Max,

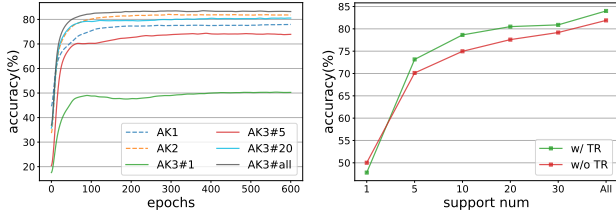


Figure 5. (a) compares the training accuracies for different auxiliary knowledge. (b) compares TZSL performance with and without the transductive regressor when knowledge AK3 is used.

$\frac{d\sigma(v_i)}{dv_i} = \sigma(v_i)(1 - \sigma(v_i))$ . The sigmoid feature has a smaller derivative with a larger input magnitude and vice versa. This causes the sigmoid to skew the activated output towards the middle value e.g., 0.5, and this is not suitable when the feature value distribution is skewed to one side, especially for those features last activated by ReLU. It can be seen in Figure 4c that  $L_2$  normalization performs better than Min-Max in terms of higher accuracy and faster convergence in early training. We examine further the effect of the radius parameter  $r$  on different datasets in Figure 4d. It is observed that a smaller  $r$  could lead to a more stable performance, while a larger  $r$  results in an increased gradient that could potentially cause instability in the training.

**On Auxiliary Knowledge of CUB** Auxiliary knowledge plays an overly important role in the success of ZSL, and the motivation of TZSL is to reduce such dependency by learning from unlabelled examples from unseen classes. To examine the effectiveness of our regressor proposed to improve transductive learning and reduce dependency, we conduct experiments using different types of auxiliary knowledge on the CUB dataset. These include the original 312-dim attribute vectors (AK1), the 1024-dim CNN-RNN embedding from fine-grained visual description [34] (AK2), and the 2048-dim averaged visual prototype features of  $n$ -shot support (unseen) dataset with label information as assumed in few-shot learning (AK3# $n$ ) [26]. A ground-truth prototype (AK3#all) is the strongest auxiliary information where the arrival of a conditional alignment is easier. The effectiveness of the prototypes becomes weaker as the number of support examples decreases, where the outliers tend to dominate the generation. AK2 performs similarly well to the ground-truth prototypes, and this indicates that the embeddings learned from a large-scale language model can serve as a good approximation to the ground-truth visual prototypes. The training accuracies are compared in Figure 5.(a). Figure 5.(b) is produced by removing our transductive regressor (TR) when the model conditions on AK3. It is observed that when TR is not employed, a considerable gap opens up for the less informative prototypes. This demonstrates that TR can improve the model’s robustness to auxiliary quality. It is important to note that when the quality is extremely low (as in the AK3#1 case), alignment

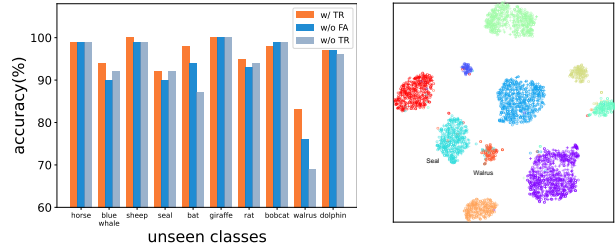


Figure 6. (a) The classification accuracy of different classes in AWA2. ‘FA’ denotes feature augmentation. (b) T-SNE visualization of augmented real/synthesized feature.

in the unseen domain is hard to realize no matter whether the TR module is presented or not.

**On Feature Augmentation** The transductive regressor supports distinguishing difficult examples, while the concatenated features based on cross-modal knowledge (see Section 3.3.4) improve the feature discriminability. Figure 6.(a) shows that the regressor leads to a better alignment for the less discriminative classes, such as ‘bat’ and ‘walrus’, and the concatenated features contribute significantly to hardness-aware alignment. Figure 6.(b) shows that for the resemble (hard) pairs, such as ‘walrus’ and ‘seal’, the concatenated features are better decoupled in this multi-modal space and the synthesis becomes more discriminative (compare with the visualization in Figure 1).

## 5. CONCLUSION

We have presented a novel bi-directional cross-modal generative model for TZSL. By generating domain-aligned features for the unseen classes from both the forward and backward directions, the distribution alignment between the visual and auxiliary space has been significantly enhanced. By conducting extensive experiments, we have discovered that  $L_2$  normalization can result in a more stable training than the commonly used Min-Max normalization. We have also conducted a thorough examination of how the unseen class prior can affect the model performance and proposed a more effective prior estimation approach, which guarantees the generative approach could still be effective under challenging unknown class prior scenarios.

## 6. ACKNOWLEDGEMENT

This work is mainly supported by the National Key Research and Development Program of China (2021ZD0111802). It was also supported in part by the National Key Research and Development Program of China under Grant 2020YFB1406703, and by the National Natural Science Foundation of China (Grants No.62101524 and No.U21B2026).



## References

- [1] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for attribute-based classification. In *CVPR*, pages 819–826, 2013. 6
- [2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *ICML*, pages 214–223. PMLR, 2017. 3
- [3] Liu Bo, Qiulei Dong, and Zhanyi Hu. Hardness sampling for self-training based transductive zero-shot learning. In *CVPR*, pages 16499–16508, 2021. 2, 6
- [4] Soravit Changpinyo, Wei-Lun Chao, and Fei Sha. Predicting visual exemplars of unseen classes for zero-shot learning. In *ICCV*, pages 3476–3485, 2017. 2
- [5] Long Chen, Hanwang Zhang, Jun Xiao, Wei Liu, and Shih-Fu Chang. Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In *CVPR*, pages 1043–1052, 2018. 6
- [6] Shiming Chen, Ziming Hong, Guo-Sen Xie, Wenhan Yang, Qinmu Peng, Kai Wang, Jian Zhao, and Xinge You. Msdn: Mutually semantic distillation network for zero-shot learning. In *CVPR*, pages 7612–7621, 2022. 1
- [7] Yu-Ying Chou, Hsuan-Tien Lin, and Tyng-Luh Liu. Adaptive and generative zero-shot learning. In *ICLR*, 2020. 2
- [8] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018. 2
- [9] Remi Tachet des Combes, Han Zhao, Yu-Xiang Wang, and Geoffrey J. Gordon. Domain adaptation with conditional distribution matching and generalized label shift. In *NeurIPS*, 2020. 5
- [10] Carl Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*, 2016. 2
- [11] Mohamed Elhoseiny and Mohamed Elfeki. Creativity inspired zero-shot learning. In *ICCV*, pages 5784–5793, 2019. 1
- [12] Rafael Felix, Ian Reid, Gustavo Carneiro, et al. Multi-modal cycle-consistent generalized zero-shot learning. In *ECCV*, pages 21–37, 2018. 2
- [13] Yaogong Feng, Xiaowen Huang, Pengbo Yang, Jian Yu, and Jitao Sang. Non-generative generalized zero-shot learning via task-correlated disentanglement and controllable samples synthesis. In *CVPR*, pages 9346–9355, 2022. 6
- [14] Yanwei Fu, Timothy M Hospedales, Tao Xiang, and Shao-gang Gong. Transductive multi-view zero-shot learning. *TPAMI*, 37(11):2332–2345, 2015. 2
- [15] Yuxia Geng, Jiaoyan Chen, Zhuo Chen, Jeff Z Pan, Zhi-quan Ye, Zonggang Yuan, Yantao Jia, and Huajun Chen. Ontozsl: Ontology-enhanced zero-shot learning. In *WWW*, pages 3325–3336, 2021. 1
- [16] Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. In *NeurIPS*, pages 5767–5777, 2017. 3
- [17] Celina Hanouti and Hervé Le Borgne. Learning semantic ambiguities for zero-shot learning. *arXiv preprint arXiv:2201.01823*, 2022. 6
- [18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 6
- [19] Xia Kong, Zuodong Gao, Xiaofan Li, Ming Hong, Jun Liu, Chengjie Wang, Yuan Xie, and Yanyun Qu. En-compactness: Self-distillation embedding & contrastive generation for generalized zero-shot learning. In *CVPR*, pages 9306–9315, 2022. 6
- [20] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *TPAMI*, 36(3):453–465, 2013. 6
- [21] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *AAAI*, volume 1, page 3, 2008. 1
- [22] Chung-Wei Lee, Wei Fang, Chih-Kuan Yeh, and Yu-Chiang Frank Wang. Multi-label zero-shot learning with structured knowledge graphs. In *CVPR*, pages 1576–1585, 2018. 1
- [23] Boyi Li, Felix Wu, Ser-Nam Lim, Serge Belongie, and Kilian Q Weinberger. On feature normalization and data augmentation. In *CVPR*, pages 12383–12392, 2021. 3
- [24] Jingjing Li, Mengmeng Jing, Ke Lu, Zhengming Ding, Lei Zhu, and Zi Huang. Leveraging the invariant side of generative zero-shot learning. In *CVPR*, pages 7402–7411, 2019. 1, 6
- [25] Kai Li, Martin Renqiang Min, and Yun Fu. Rethinking zero-shot learning: A conditional visual classification perspective. In *ICCV*, pages 3583–3592, 2019. 6, 7
- [26] Kai Li, Yulun Zhang, Kunpeng Li, and Yun Fu. Adversarial feature hallucination networks for few-shot learning. In *CVPR*, pages 13470–13479, 2020. 8
- [27] Xiangyu Li, Xu Yang, Kun Wei, Cheng Deng, and Muli Yang. Siamese contrastive embedding network for compositional zero-shot learning. In *CVPR*, pages 9326–9335, 2022. 6
- [28] Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In *ICML*, pages 3122–3130. PMLR, 2018. 6
- [29] Sanath Narayan, Akshita Gupta, Fahad Shahbaz Khan, Cees GM Snoek, and Ling Shao. Latent embedding feedback and discriminative features for zero-shot classification. In *ECCV*, pages 479–495. Springer, 2020. 2, 3, 5, 6, 7
- [30] Mohammad Norouzi, Tomáš Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. In *ICLR*, 2014. 1
- [31] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*, pages 2751–2758. IEEE, 2012. 6
- [32] Akanksha Paul, Narayanan C Krishnan, and Prateek Munjal. Semantically aligned bias reducing zero shot learning. In *CVPR*, pages 7056–7065, 2019. 1, 6
- [33] Farhad Pourpanah, Moloud Abdar, Yuxuan Luo, Xinlei Zhou, Ran Wang, Chee Peng Lim, Xi-Zhao Wang, and QM Jonathan Wu. A review of generalized zero-shot learning methods. *TPAMI*, 2022. 6

- [34] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *CVPR*, pages 49–58, 2016. 1, 8
- [35] Mert Bulent Sariyildiz and Ramazan Gokberk Cinbis. Gradient matching generative networks for zero-shot learning. In *CVPR*, pages 2168–2178, 2019. 6
- [36] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, pages 1199–1208, 2018. 2
- [37] Vinay Kumar Verma and Piyush Rai. A simple exponential family framework for zero-shot learning. In *ECML PKDD*, pages 792–808. Springer, 2017. 6
- [38] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 1
- [39] Ziyu Wan, Dongdong Chen, Yan Li, Xingguang Yan, Junge Zhang, Yizhou Yu, and Jing Liao. Transductive zero-shot learning with visual structure constraint. In *NeurIPS*, pages 9972–9982, 2019. 2, 7
- [40] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010. 6
- [41] Jiamin Wu, Tianzhu Zhang, Zheng-Jun Zha, Jiebo Luo, Yongdong Zhang, and Feng Wu. Self-supervised domain-aware generative network for generalized zero-shot learning. In *CVPR*, pages 12767–12776, 2020. 1, 2, 3, 6
- [42] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *TPAMI*, 41(9):2251–2265, 2018. 1, 6
- [43] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *CVPR*, pages 5542–5551, 2018. 1, 2, 6
- [44] Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. f-vaegan-d2: A feature generating framework for any-shot learning. In *CVPR*, pages 10275–10284, 2019. 1, 2, 3, 6, 7
- [45] Hairui Yang, Baoli Sun, Baopu Li, Caifei Yang, Zhihui Wang, Jenhui Chen, Lei Wang, and Haojie Li. Iterative class prototype calibration for transductive zero-shot learning. *TCSVT*, 2022. 2
- [46] Meng Ye and Yuhong Guo. Zero-shot classification with discriminative semantic representation learning. In *CVPR*, pages 7140–7148, 2017. 6, 7
- [47] Meng Ye and Yuhong Guo. Progressive ensemble networks for zero-shot recognition. In *CVPR*, pages 11728–11736, 2019. 6
- [48] Yunlong Yu, Zhong Ji, Yanwei Fu, Jichang Guo, Yanwei Pang, and Zhongfei (Mark) Zhang. Stacked semantics-guided attention model for fine-grained zero-shot learning. In *NeurIPS*, pages 5998–6007, 2018. 2
- [49] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *CVPR*, pages 2021–2030, 2017. 2, 4, 6
- [50] Ziming Zhang and Venkatesh Saligrama. Zero-shot recognition via structured prediction. In *ECCV*, pages 533–548. Springer, 2016. 1
- [51] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, pages 2223–2232, 2017. 2