A Boundary Based Out-of-Distribution Classifier for Generalized Zero-Shot Learning

Xingyu Chen[1][0000−0002−5226−963X], Xuguang Lan[1][0000−0002−3422−944X], Fuchun Sun[2][0000−0003−3546−6305], and Nanning Zheng[1][0000−0003−1608−8257]

[1] Xi'an Jiaotong University, Xi'an, China,
[2] Tsinghua University, Beijing, China
xingyuchen1990@gmail.com, {xglan,nnzheng}@mail.xjtu.edu.cn,
fcsun@tsinghua.edu.cn

Abstract. Generalized Zero-Shot Learning (GZSL) is a challenging topic that has promising prospects in many realistic scenarios. Using a gating mechanism that discriminates the unseen samples from the seen samples can decompose the GZSL problem to a conventional Zero-Shot Learning (ZSL) problem and a supervised classification problem. However, training the gate is usually challenging due to the lack of data in the unseen domain. To resolve this problem, in this paper, we propose a boundary based Out-of-Distribution (OOD) classifier which classifies the unseen and seen domains by only using seen samples for training. First, we learn a shared latent space on a unit hyper-sphere where the latent distributions of visual features and semantic attributes are aligned class-wisely. Then we find the boundary and the center of the manifold for each class. By leveraging the class centers and boundaries, the unseen samples can be separated from the seen samples. After that, we use two experts to classify the seen and unseen samples separately. We extensively validate our approach on five popular benchmark datasets including AWA1, AWA2, CUB, FLO and SUN. The experimental results show that our approach surpasses state-of-the-art approaches by a significant margin.

Keywords: Generalized Zero-Shot Learning, boundary based Out-of-Distribution classifier.

1 Introduction

Zero-Shot Learning (ZSL) is an important topic in the computer vision community which has been widely adopted to solve challenges in real-world tasks. In the conventional setting, ZSL aims at recognizing the instances drawn from the unseen domain, for which the training data are lacked and only the semantic auxiliary information is available. However, in real-world scenarios, the instances are drawn from either unseen or seen domains, which is a more challenging task called Generalized Zero-Shot Learning (GZSL).

Previous GZSL algorithms can be grouped into three lines: (1) Embedding methods [2, 1, 3, 14, 25, 18, 28, 27, 5, 10, 33] which aim at learning embeddings that unify the visual features and semantic attributes for similarity measurement. However, due to the bias problem [33], the projected feature anchors

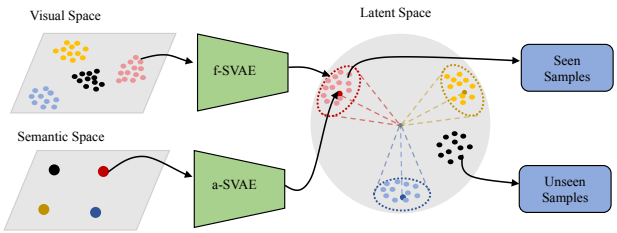

Fig. 1. The boundary based OOD classifier learns a bounded manifold for each seen class on a unit hyper-sphere (latent space). By using the manifold boundaries (dotted circles) and the centers (dark-colored dots), the unseen samples (black dots) can be separated from the seen samples (colored dots).

of unseen classes may be distributed too near to that of seen classes in the embedding space. Consequently, the unseen samples are easily classified into nearby seen classes. (2) Generative methods [21, 6, 31, 9, 15, 26] which focus on generating synthetic features for unseen classes by using generative models such as GAN [11] or VAE [13]. By leveraging the synthetic data, the GZSL problem can be converted to a supervised problem. Although the generative methods substantially improve the GZSL performance, they are still bothered by the feature confusion problem [17]. Specifically, the synthetic unseen features may be tangled with the seen features. As a result, a classifier will be confused by the features which have strong similarities but different labels. An intuitive phenomenon is that previous methods usually make trade-offs between the accuracy of seen classes and unseen classes to get higher Harmonic Mean values. (3) Gating methods [4, 27] which usually incorporates a gating mechanism with two experts to handle the unseen and seen domains separately. Ideally, if the binary classifier is reliable enough, the GZSL can be decomposed to a ZSL problem and a supervised classification problem, which does not suffer from the feature confusion or bias problem. Unfortunately, it is usually difficult to learn such a classifier because unseen samples are not available during training.

To resolve the main challenge in the gating methods, we propose a boundary based Out-of-Distribution (OOD) classifier for GZSL in this paper. As illustrated in Fig. 1, the key idea of our approach is to learn a bounded manifold for each seen class in the latent space. A datum that can be projected into the bounded manifold will be regarded as a seen sample. Otherwise, we believe it is an unseen sample. In this way, we can easily separate unseen classes from seen classes even we do not use any unseen samples for training.

To learn a bounded manifold for each seen class, the proposed OOD classifier learns a shared latent space for both visual features and the semantic attributes. In the latent space, the distributions of visual features and semantic attributes are aligned class-wisely. Different from previous latent distribution aligning approach [26], we build the latent space on a unit hyper-sphere by using

Hyper-Spherical Variational Auto-Encoders (SVAE) [8]. Specifically, the approximated posterior of each visual feature is encouraged to be aligned with a von Mises-Fisher (vMF) distribution, where the mean direction and concentration are associated with the corresponding semantic attribute. Therefore, each class can be represented by a vMF distribution on the unit hyper-sphere, which is easy to find the manifold boundary. In addition, the mean direction predicted by semantic attribute can be regarded as the class center. By leveraging the boundary and the class center, we can determine if a datum is projected into the manifold. In this way, the unseen features can be separated from the seen features. After that, we apply two experts to classify the seen features and unseen features separately.

The proposed classifier can incorporate with any state-of-the-art ZSL method. The core idea is very straightforward and easy to implement. We evaluate our approach on five popular benchmark datasets, i.e. AWA1, AWA2, CUB, FLO and SUN for generalized zero-shot learning. The experimental results show that our approach surpasses the state-of-the-art approaches by a significant margin.

2 Related Work

Embedding Methods To solve GZSL, the embedding methods [1–3, 14, 25, 18, 28, 27, 5, 10, 33, 35, 19, 32, 34] usually learn a mapping to unify the visual features and semantic attributes for similarity measurement. For example, Zhang et al. [35] embed features and attributes into a common space where each point denotes a mixture of seen class proportions. Other than introducing common space, Kodirov et al. [14] propose a semantic auto-encoder which aims to embed visual feature vector into the semantic space while constrain the projection must be able to reconstruct the original visual feature. On the contrary, Long et al. [19] learn embedding from semantic space into visual space. However, due to the bias problem, previous embedding methods usually misclassify the unseen classes into seen classes. To alleviate the bias problem, Zhang et al. [33] propose a co-representation network which adopts a single-layer cooperation module with parallel structure to learn a more uniform embedding space with better representation.

Generative Methods The generative methods [21, 6, 31, 9, 15, 26] treat GZSL as a case of missing data and try to generate synthetic samples of unseen classes from semantic information. By leveraging the synthetic data, the GZSL problem can be converted to a supervised classification problem. Therefore, These methods usually rely on generative models such as GAN [11] and VAE [13]. For example, Xian et al. [31] directly generate image features by pairing a conditional WGAN with a classification loss. Mishara et al [21] utilize a VAE to generate image features conditional on the class embedding vector. Felix et al. [9] propose a multi-modal cycle-consistent GAN to improve the quality of the synthetic features. Compared to the embedding methods, the generative methods significantly improve the GZSL performance. However, Li et al. [17] find

that the generative methods are bothered by the feature confusion problem. To alleviate this problem, they present a boundary loss which maximizes the decision boundary of seen categories and unseen ones while training the generative model.

Gating Methods There are a few works using a gating based mechanism to separate the unseen samples from the seen samples for GZSL. The gate usually incorporates two experts to handle seen and unseen domains separately. For example, Socher et al. [27] propose a hard gating model to assign test samples to each expert. Only the selected expert is used for prediction, ignoring the other expert. Recently, Atzmon et al. [4] propose a soft gating model which makes soft decisions if a sample is from a seen class. The key to the soft gating is to pass information between three classifiers to improve each one's accuracy. Different from the embedding methods and the generative methods, the gating methods do not suffer from the bias problem or the feature confusion problem. However, a key difficulty in gating methods is to train a binary classifier by only using seen samples. In this work, we propose a boundary based OOD classifier by only using seen samples for training. The proposed classifier is a hard gating model. Compared to previous gating methods, it provides much more accurate classification results.

3 Revisit Spherical Variational Auto-Encoders

The training objective of a general variational auto-encoder is to maximize $\log \int p_\phi(x, z)dz$, the log-likelihood of the observed data, where x is the training data, z is the latent variable and $p_\phi(x, z)$ is a parameterized model representing the joint distribution of x and z. However, computing the marginal distribution over the latent variable z is generally intractable. In practice, it is implemented to maximize the Evidence Lower Bound (ELBO).

$$\log \int p_\phi(x, z)dz \geq \mathbb{E}_{q(z))} \left[\log p_\phi(x|z)\right] - KL(q(z)||p(z)), \tag{1}$$

where $q(z)$ approximates the true posterior distribution and $p(z)$ is the prior distribution. $p_\phi(x|z)$ is to map a latent variable to a data point x which is parameterized by a decoder network. $KL(q(z)||p(z))$ is the Kullback-Leibler divergence which encourages $q(z)$ to match the prior distribution. The main difference for various variational auto-encoders is in the adopted distributions.

For SVAE [8], both of the prior and posterior distributions are based on von Mises-Fisher(vMF) distributions. A vMF distribution can be regarded as a Normal distribution on a hyper-sphere, which is defined as:

$$q(z|\mu, \kappa) = C_m(\kappa) \exp(\kappa \mu^T z) \tag{2}$$

$$C_m(\kappa) = \frac{\kappa^m/2 - 1}{(2\pi)^{m/2} I_{m/2-1}(\kappa)} \tag{3}$$

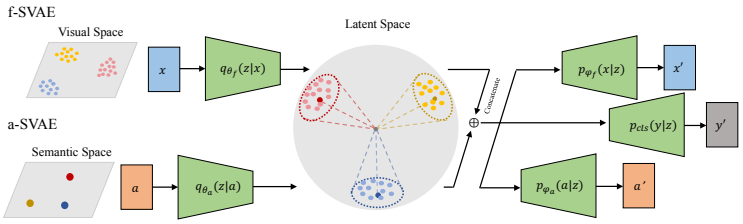

Fig. 2. Our model consists of two SVAEs, one for visual features and another for semantic attributes. By combining the objective functions of the two SVAEs with a cross-reconstruction loss and a classification loss, we train our model to align the latent distributions of visual features and semantic attributes class-wisely. In this way, each class can be represented by a vMF distribution whose boundary is easy to find.

where $\mu \in \mathbb{R}^m, ||\mu||_2 = 1$ represents the direction on the sphere and $\kappa \in \mathbb{R}_{\geq 0}$ represents the concentration around μ. $C_m(\kappa)$ is the normalizing constant, I_v is the modified Bessel function of the first kind at order v.

Theoretically, $q(z)$ should be optimized over all data points, which is not tractable for large dataset. Therefore it uses $q_\theta(z|x) = q(z|\mu(x), \kappa(x))$ which is parameterized by an encoder network to do stochastic gradient descent over the dataset. The final training objective is defined as:

$$L_{\text{SVAE}}(\theta, \phi; x) = \mathbb{E}_{q_\theta(z|x)}\left[\log p_\phi(x|z)\right] - KL(q_\theta(z|x)||p(z)). \tag{4}$$

4 Proposed Approach

4.1 Problem Formulation

We first introduce the definitions of OOD classification and GZSL. We are given a set of training samples of seen classes $\mathcal{S} = \{(x, y, a)|x \in \mathcal{X}, y \in \mathcal{Y}_s, a \in \mathcal{A}_s\}$ where x represents the feature of an image extracted by a CNN, y represents the class label in $\mathcal{Y}_s = \left\{y_s^1, y_s^2, ..., y_s^N\right\}$ consisting of N seen classes and a represents corresponding class-level semantic attribute which is usually hand-annotated or a Word2Vec feature [20]. We are also given a set $\mathcal{U} = \{(y, a)|y \in \mathcal{Y}_u, a \in \mathcal{A}_u\}$ of unseen classes $\mathcal{Y}_u = \left\{y_u^1, y_u^2, ..., y_u^M\right\}$. The zero shot recognition states that $\mathcal{Y}_s \cap \mathcal{Y}_u = \varnothing$. Given $\mathcal{S}$ and $\mathcal{U}$, the OOD classifier aims at learning a binary classifier $f_{OOD} : \mathcal{X} \rightarrow \{0, 1\}$ that distinguishes if a datum belongs to $\mathcal{S}$ or $\mathcal{U}$. The task of GZSL aims at learning a classifier $f_{gzsl} : \mathcal{X} \rightarrow \mathcal{Y}_s \cup \mathcal{Y}_u$.

4.2 Boundary Based Out-of-Distribution Classifier

The proposed OOD classifier aims to classify the unseen and seen domains by only using seen samples for training. The core idea of our approach is quite

straightforward. First, we build a latent space for visual features and semantic attributes. Then we learn a bounded manifold for each seen class. Next we find the boundaries of the learned manifolds. By leveraging the boundaries, we can determine if a test sample is projected into the manifolds. For the samples which can be projected into the manifolds, we believe they belong to the seen domain and assign them to a seen expert. Otherwise, we assign them to an unseen expert.

Build the Latent Space on a Unit Hyper-sphere Different from previous works, we build the latent space on a unit hyper-sphere by using hyper-spherical variational auto-encoders. In the latent space, each class is approximately represented by a vMF distribution of which the mean direction can be regarded as the class center. Using the spherical representation has two advantages. First, we can naturally use cosine similarity as the distance metric since all latent variables and class centers are located on the unit hyper-sphere. Second, for each seen class, it is easy to find the manifold boundary. Specifically, we can find a threshold based on the cosine similarities between the latent variables and the class center. According to the class center and the corresponding boundary, we can determine if a visual feature is projected into the manifold.

Learn A Bounded Manifold for Each Class To learn a bounded manifold for each class, inspired by [26], we encourage the latent distributions of visual features and the corresponding semantic attribute to be aligned with each other in the latent space. As illustrated in Fig. 2, our model consists of two SVAEs correspond to two data modalities, one for visual features and another for semantic attributes, denoted as f-SVAE and a-SVAE, respectively. Given an attribute $a \in \mathcal{A}_s$, the encoder of a-SVAE predicts a vMF distribution $q_{\theta_a}(z|a) = q(z|\mu(a), \kappa(a))$. Meanwhile, given the corresponding visual feature x, the encoder of f-SVAE predicts a vMF distribution $q_{\theta_f}(z|x) = q(z|\mu(x), \kappa(x))$. Each SVAE regards the distribution predicted by another SVAE as the prior distribution. Therefore, we can align the two distributions by optimizing the objective functions of f-SVAE and a-SVAE simultaneously. We further adopt a cross-reconstruction loss and a classification loss to ensure the latent representations capture the modality invariant information while preserving discrimination. Therefore, the training objective consists four parts.

f-SVAE: For the f-SVAE, we expect to maximize the log-likelihood and minimize the discrepancy between the approximated posterior $q_{\theta_f}(z|x)$ and the prior distribution $q_{\theta_a}(z|a)$. Therefore, the training objective is defined as:

$$L_{f-SVAE} = \mathbb{E}_{p(x,a)}[\mathbb{E}_{q_{\theta_f}(z|x)}[\log p_{\phi_f}(x|z)] - \lambda_f D_z(q_{\theta_f}(z|x) \parallel q_{\theta_a}(z|a))], \quad (5)$$

where $\mathbb{E}_{q_{\theta_f}(z|x)}[\log p_{\phi_f}(x|z)]$ represents the expectation of log-likelihood over latent variable z. In practice, we use the negative reconstruction error of visual feature x instead. $p_{\phi_f}(x|z)$ is the decoder network of f-SVAE. $D_z(q_{\theta_f}(z|x) \parallel q_{\theta_a}(z|a))$ represents the discrepancy between the two vMF distributions. λ_f is a hyper-parameter to weight the discrepancy term. It worth noting that

$D_z(q_{\theta_f}(z|x) \parallel q_{\theta_a}(z|a))$ is the Earth Mover's Distance (EMD) between the two distributions which is defined as:

$$D_z(q_{\theta_f}(z|x) \parallel q_{\theta_a}(z|a)) = \inf_{\Omega \in \prod(q_{\theta_f}, q_{\theta_a})} \mathbb{E}_{(z_1,z_2)\sim\Omega}[\parallel z_1 - z_2 \parallel]. \tag{6}$$

The reason we use EMD instead of the KL-divergence is that the KL-divergence may fail when the support regions of the two distributions $q_{\theta_f}(z|x)$ and $q_{\theta_a}(z|a)$ do not completely coincide. To calculate the EMD, we utilize the Sinkhorn iteration algorithm in [7].

a-SVAE: Similarly, for the a-SVAE, $q_{\theta_f}(z|x)$ is regarded as the prior distribution. The objective function is defined as:

$$L_{a-SVAE} = \mathbb{E}_{p(x,a)}[\mathbb{E}_{q_{\theta_a}(z|a)}[\log p_{\phi_a}(a|z)] - \lambda_a D_z(q_{\theta_a}(z|a) \parallel q_{\theta_f}(z|x))], \tag{7}$$

where $\mathbb{E}_{q_{\theta_a}(z|a)}[\log p_{\phi_a}(a|z)]$ represents the negative reconstruction error of semantic attribute a. $D_z(q_{\theta_a}(z|a) \parallel q_{\theta_f}(z|x))$ is the discrepancy between the two vMF distributions. As EMD is symmetrical, $D_z(q_{\theta_a}(z|a) \parallel q_{\theta_f}(z|x))$ equals to $D_z(q_{\theta_f}(z|x) \parallel q_{\theta_a}(z|a))$, weighted by hyper-parameter λ_a.

Cross-reconstruction Loss: Since we learn a shared latent space for the two different modalities, the latent representations should capture the modality invariant information. For this purpose, we also adopt a cross-reconstruction regularizer:

$$L_{cr} = \mathbb{E}_{p(x,a)}[\mathbb{E}_{q_{\theta_a}(z|a)}[\log p_{\phi_f}(x|z)] + \mathbb{E}_{q_{\theta_f}(z|x)}[\log p_{\phi_a}(a|z)]], \tag{8}$$

where $\mathbb{E}_{q_{\theta_a}(z|a)}[\log p_{\phi_f}(x|z)]$ and $\mathbb{E}_{q_{\theta_f}(z|x)}[\log p_{\phi_a}(a|z)]$ also represent negative reconstruction errors.

Classification Loss: To make the latent variables more discriminate, we introduce the following classification loss:

$$L_{cls} = \mathbb{E}_{p(x,y,a)}[\mathbb{E}_{q_{\theta_a}(z|a)}[\log p_{\phi_{cls}}(y|z)] + \mathbb{E}_{q_{\theta_f}(z|x)}[\log p_{\phi_{cls}}(y|z)]], \tag{9}$$

where ϕ_{cls} represents the parameters of a linear softmax classifier. Although the classification loss may hurt the inter-class association between seen and unseen classes, it also reduces the risk for unseen features being projected into the manifolds of seen classes, which benefits to the binary classification. The reason is that our OOD classifier only cares about separating unseen features from the seen features, but not cares about which class the unseen features belong to.

Overall Objective: Finally, we train our model by maximizing the following objective:

$$L_{overall} = L_{f-SVAE} + L_{a-SVAE} + \alpha L_{cr} + \beta L_{cls}, \tag{10}$$

where α, β are the hyper-parameters used to weight the two terms.

Find the Boundaries for OOD Classification When the proposed model is trained to convergence, the visual features and the semantic attributes are

aligned class-wisely in the latent space. Each class is represented by a vMF distribution. Therefore, the manifold of each class can be approximately represented by a circle on the unit hyper-sphere. By leveraging the center and the boundary, we can determine whether a latent variable locates in the manifold.

For class $y^i \in \mathcal{Y}_s$, the class center can be found by using its semantic attribute. Given $a^i \in \mathcal{A}_s$, a-SVAE predicts a vMF distribution $q(z|\mu(a^i), \kappa(a^i))$ of which $\mu(a^i)$ is regarded as the class center.

There could be many ways to find the boundaries. In this paper, we present a simple yet effective one. We first encode all training samples of seen classes to latent variables. After that we calculate the cosine similarity $S(z^i, \mu(a^i))$ between each latent variable z^i and the corresponding class center $\mu(a^i)$. Then we search a threshold η which is smaller than $\gamma \in (0, 100\%)$ and larger than $1-\gamma$ of the cosine similarities. We adopt η for all seen classes to represent the boundaries. Here, γ can be viewed as the OOD classification accuracy on training samples. Given a γ, we can find the corresponding threshold η.

Given a test sample x which may come from a seen class or an unseen class, we first encode it to latent variable z. Then we compute the cosine similarities between it to all seen class centers and find the maximum. By leveraging the threshold η, we determine the test sample belongs to unseen class or seen class using Eq.(11),

$$y^{OOD} = \begin{cases} 0, & \text{if} \quad \max\{S(z, \mu(a^i))|\forall a^i \in \mathcal{A}_s\} < \eta \\ 1, & \text{if} \quad \max\{S(z, \mu(a^i))|\forall a^i \in \mathcal{A}_s\} \geq \eta \end{cases} \tag{11}$$

where 0 stands for unseen class and 1 for seen class.

Generalized Zero-Shot Classification For the GZSL task, we incorporate the proposed OOD classifier with two domain experts. Given a test sample, the OOD classifier determines if it comes from a seen class. Then, according to the predicted label, the test sample is assigned to a seen expert or an unseen expert for classification.

4.3 Implementation Details

OOD Classifier For the f-SVAE, we use two 2-layer Fully Connected (FC) network for the encoder and decoder networks. The first FC layer in the encoder has 512 neurons with ReLU followed. The output is then fed to two FC layers to produce the mean direction and the concentration for the reparameterize trick. The mean direction layer has 64 neurons and the concentration layer only has 1 neuron. The output of mean direction layer is normalized by its norm such that it lies on the unit hyper-sphere. The concentration layer is followed by a Softplus activation to ensure its output larger than 0. The decoder consists of two FC layers. The first layer has 512 neurons with ReLU followed. Then second layer has 2048 neurons.

The structure of a-SVAE is similar to f-SVAE except for the input dimension and the neuron number of the last FC layer equal to the dimension of the

semantic attributes. We use a Linear Softmax classifier which takes the latent variables as input for calculating the classification loss. The structure is same as in [31].

We train our model by the Adam optimizer with learning rate 0.001. The batch size is set to 128. The hyper-parameter λ_f, λ_a, α, β are set to 0.1, 0.1, 1.0 and 1.0, respectively.

Unseen and Seen Experts For the unseen samples, we use the f-CLSWGAN [31] with the code provided by the authors. For the seen samples, we directly combine the encoder of f-SVAE and the linear softmax classifier for classification.

5 Experiments

The proposed approach is evaluated on five benchmark datasets, where plenty of recent state-of-the-art methods are compared. Moreover, the features and settings used in experiments follow the paper [30] for fair comparison.

5.1 Datasets, Evaluation and Baselines

Datasets The five benchmark datasets include Animals With Attributes 1 (AWA1)[16], Animal With Attributes 2 (AWA2) [30], Caltech-UCSD-Birds (CUB) [29], FLOWER (FLO) [22] and SUN attributes (SUN) [23]. Specifically, AWA1 contains 30,475 images and 85 kinds of properties, where 40 out of 50 classes are obtained for training. In AWA2, 37,322 images in the same classes are re-collected because original images in AWA1 are not publicly available. CUB has 11,788 images from 200 different types of birds annotated with 312 properties, where 150 classes are seen and the others are unseen during training. FLO consists of 8,189 images which come from 102 flower categories, where 82/20 classes are used for training and testing. For this dataset, we use the same semantic descriptions provided by [24]. SUN has 14,340 images of 717 scenes annotated with 102 attributes, where 645 classes are regarded as seen classes and the rest are unseen classes.

Evaluation For OOD classification, the in-distribution samples are regarded as the seen samples and the out-of-distribution samples are regarded as unseen samples. The True-Positive-Rate (**TPR**) indicates the classification accuracy of seen classes and the False-Positive-Rate (**FPR**) indicates the accuracy of unseen classes. We also measure the Area-Under-Curve (**AUC**) by sweeping over classification threshold.

For GZSL, the average of per-class precision (AP) is measured. The "**ts**" and "**tr**" denote the Average Precision (AP) of images from unseen and seen classes, respectively. "**H**" is the harmonic mean which is defined as: $H = 2 * tr * ts/(tr + ts)$. The harmonic mean reflects the ability of method that recognizes seen and unseen images simultaneously.

Table 1. Comparison with various gating models on validation set. **AUC** denotes Area-Under-Curve when sweeping over detection threshold. **FPR** denotes False-Positive-Rate on the threshold that yields 95% True Positive Rate for detecting in-distribution samples. The best results are highlighted with bold numbers.

	AWA1			CUB			SUN		
Method	**H**	**AUC**	**FPR**	**H**	**AUC**	**FPR**	**H**	**AUC**	**FPR**
MAX-SOFTMAX-3 [12]	53.1	88.6	56.8	43.6	73.4	79.6	38.4	61.0	92.3
CB-GATING-3[4]	56.8	92.5	45.5	44.8	82.0	72.0	40.1	77.7	77.5
Ours	**70.1**	**95.0**	**12.5**	**67.7**	**99.4**	**2.5**	**71.0**	**99.5**	**1.6**

Table 2. OOD classification results of our approach by selecting different thresholds using γ.

	AWA1		AWA2		CUB		FLO		SUN	
	TPR	**FPR**	**TPR**	**FPR**	**TPR**	**FPR**	**TPR**	**FPR**	**TPR**	**FPR**
$\gamma = 0.85$	85.0	5.3	85.2	6.8	84.2	0.7	85.3	0.4	85.4	0.2
$\gamma = 0.90$	90.1	6.3	89.8	8.2	89.5	0.9	88.2	0.6	90.6	0.2
$\gamma = 0.95$	95.4	7.9	95.2	10.6	94.9	1.1	94.4	0.8	95.1	0.4

GZSL Baselines We compare our approach with three lines of previous works in the experiments. (1) Embedding methods which focus on learning embeddings that unify the visual features and semantic attributes for similarity measurement. We include the recent competitive baselines: SJE [2], ALE [1], PSR [3], SAE [14], EZSL [25], LESAE [18], ReViSE [28], CMT [27], SYNC [5], DeViSE [10] and CRnet [33]. (2) Generative methods which focus on generating synthetic features or images for unseen classes using GAN or VAE. We also compare our approach with the recent state-of-the-arts such as CVAE [21], SP-AEN [6], f-CLSWGAN [31], CADA-VAE [26], cycle-(U)WGAN [9], SE [15] and AFC-GAN [17]. (3) Gating methods which aim at learning a classifier to distinguish the unseen features from the seen features. We compare our approach with the recent state-of-the-art COSMO [4].

5.2 Out-of-Distribution Classification

In this experiment, We conduct OOD classification experiments on the five benchmark datasets.

We first compare the boundary based OOD classifier with two state-of-the-art gating-based methods: (1) MAX-SOFTMAX-3 is a basline gating model of [12]. (2)CB-GATING-3 is the best confidence-based gating model in [4]. For a fair comparison, we use the same dataset splitting as in [4]. Table 1 shows the classification results of the proposed OOD classifier compared to the two baseline methods. It worth noting that the FPR scores are reported on the threshold that yields 95% TPR for detecting in-distribution samples. It can be seen that the two baseline methods have much higher FPR values. For example, the FPR of CB-GATING-3 is 45.5% on AWA1, 72.0% on CUB and 77.5% on SUN. It indicates that most of the unseen samples are misclassified to the seen samples. However, the FPR of our approach is reduced to 12.5% on AWA1, 2.5% on CUB and 1.6% on SUN, which significantly outperforms the baselines

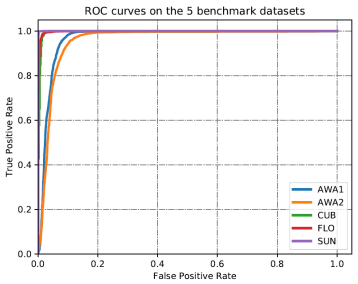

Fig. 3. The ROC curves on the five benchmark datasets.

Table 3. Generalized Zero-Shot Learning results on AWA1, AWA2, CUB, FLO and SUN. We measure the AP of Top-1 accuracy in %. The best results are highlighted with bold numbers.

	AWA1			AWA2			CUB			FLO			SUN		
Method	**ts**	**tr**	**H**	**ts**	**tr**	**H**	**ts**	**tr**	**H**	**ts**	**tr**	**H**	**ts**	**tr**	**H**
SJE [2]	11.3	74.6	19.6	8.0	73.9	14.4	23.5	59.2	33.6	13.9	47.6	21.5	14.7	30.5	19.8
ALE [1]	16.8	76.1	27.5	14.0	81.8	23.9	23.7	62.8	34.4	13.3	61.6	21.9	21.8	33.1	26.3
PSR [3]	-	-	-	20.7	73.8	32.3	24.6	54.3	33.9	-	-	-	20.8	37.2	26.7
SAE [14]	16.7	82.5	27.8	8.0	73.9	14.4	18.8	58.5	29.0	-	-	-	8.8	18.0	11.8
ESZSL [25]	6.6	75.6	12.1	5.9	77.8	11.0	12.6	63.8	21.0	11.4	56.8	19.0	11.0	27.9	15.8
LESAE [18]	19.1	70.2	30.0	21.8	70.6	33.3	24.3	53.0	33.3	-	-	-	21.9	34.7	26.9
ReViSE [28]	46.1	37.1	41.1	46.4	39.7	42.8	37.6	28.3	32.3	-	-	-	24.3	20.1	22.0
CMT [27]	0.9	87.6	1.8	0.5	90.0	1.0	7.2	49.8	12.6	-	-	-	8.1	21.8	11.8
SYNC [5]	8.9	87.3	16.2	10.0	90.5	18.0	11.5	70.9	19.8	-	-	-	7.9	43.3	13.4
DeViSE [10]	13.4	68.7	22.4	17.1	74.7	27.8	23.8	53.0	32.8	9.9	44.2	16.2	16.9	27.4	20.9
CRnet [33]	58.1	74.7	65.4	52.6	78.8	63.1	45.5	56.8	50.5	-	-	-	34.1	36.5	35.3
CVAE [21]	-	-	47.2	-	-	51.2	-	-	34.5	-	-	-	-	-	26.7
SP-AEN [6]	-	-	-	23.3	90.9	37.1	34.7	70.6	46.6	-	-	-	24.9	38.6	30.3
f-CLSWGAN [31]	57.9	61.4	59.6	52.1	68.9	59.4	43.7	57.7	49.7	59.0	73.8	65.6	42.6	36.6	39.4
cycle-(U)WGAN [9]	59.6	63.4	59.8	-	-	-	47.9	59.3	53.0	61.6	69.2	65.2	47.2	33.8	39.4
SE [15]	56.3	67.8	61.5	58.3	68.1	62.8	41.5	53.3	46.7	-	-	-	40.9	30.5	34.9
CADA-VAE [26]	57.3	72.8	64.1	55.8	75.0	63.9	51.6	53.5	52.4	-	-	-	47.2	35.7	40.6
AFC-GAN [17]	-	-	-	58.2	66.8	62.2	53.5	59.7	56.4	60.2	80.0	68.7	49.1	36.1	41.6
COSMO+fCLSWGAN [4]	64.8	51.7	57.5	-	-	-	41.0	60.5	48.9	59.6	81.4	68.8	35.3	40.2	37.6
COSMO+LAGO [4]	52.8	80.0	63.6	-	-	-	44.4	57.8	50.2	-	-	-	44.9	37.7	41.0
Ours($\gamma = 0.95$)	59.0	94.3	**72.6**	55.9	94.9	**70.3**	53.8	94.6	**68.6**	61.9	91.7	**73.9**	57.8	95.1	**71.9**

methods. Therefore, we achieve the best harmonic mean and AUC scores. Our approach can be categorized as a hard-gating approach. Compared to the soft-gating method in [4], our approach is more straightforward and more effective.

We also present the OOD classification results on the test sets of the five benchmark datasets in Table 2. It can be seen that the proposed OOD classifier shows stable performance when we sweep the threshold. The ROC curves are shown in Fig. 3, where the AUC is 96.8% on AWA1, 95.7% on AWA2, 99.6% on CUB, 99.8% on FLO and 99.9% on SUN.

5.3 Comparison with State-of-the-Arts

We further evaluate our approach on the five benchmark datasets under the GZSL setting. We report the top-1 accuracy and harmonic mean of each method in Table 3 where "-" indicates that the result is not reported.

We see that most of the embedding methods suffer from the bias problem. For example, the ts values of baseline methods [2, 1, 3, 14, 25, 18, 28, 27, 5, 10] are much lower than the tr values, which leads to poor harmonic results. Compared to the embedding methods, the generative methods [21, 6, 31, 9, 15, 26] show much higher harmonic mean results. However, due to the feature confusion problem, these methods have to make trade-offs between ts and tr values to get higher harmonic mean results. For example, the ts values of f-CLSWGAN, cycle-(U)WGAN, SE, CADA-VAE and AFC-GAN are higher than the tr values on the SUN dataset, which means the accuracy of seen classes are even worse than the unseen classes. The gating based method [4] is not good enough to classify the unseen and seen domains. Therefore the performance does not show obvious improvement compared to generative methods.

It can be seen that our approach achieves superior performance compared to the previous methods on all datasets, e.g. we achieve 72.6% harmonic mean on AWA1, 70.3% on AWA2, 68.6% on CUB, 73.9% on FLO and 71.9% on SUN, which significantly outperforms the baseline methods. In our experiments, we incorporate the proposed OOD classifier with the ZSL classifier of f-CLSWGAN. By using the proposed classifier, the ts values are improved compared to the original approach. Moreover, in our approach the ts values are significant higher. Compared to the gating based method COSMO+fCLSWGAN which also uses the ZSL classifier of f-CLSWGAN, our approach also has much higher harmonic mean results. It indicates that the proposed OOD classifier is more reliable.

Obviously, the GZSL performance of our approach mainly depends on the OOD classifier and the ZSL classifier. As our OOD classifier is reliable enough to separate the unseen features from the seen features, the GZSL problem can be substantially simplified. Therefore, our approach does not suffer the bias problem or feature confusion problem. In practice, we can replace the ZSL classifier by any state-of-the-art models. Consequently, the Harmonic mean of our approach could be further improved by using more powerful ZSL models.

5.4 Model Analysis

In this section, to give a deep insight into our approach, we analyze our model under different settings.

Latent Space Visualization To demonstrate the learned latent space, we visualize the latent variables of seen features and unseen features to the 2D-plane by using t-SNE. The visualization results of five datasets are shown in Fig. 4 where the blue dots represent the seen variables and the orange dots represent the unseen variables. It can be seen that the features in each seen class are clustered together in the latent space so that they can be easily classified. The unseen features are encoded to the latent variables chaotically scattered across the latent space. We see that most of the unseen variables locate out of the manifolds of seen classes. Although the inter-class association between seen and unseen classes is broken, the unseen variables can be easily separated from the seen variables.

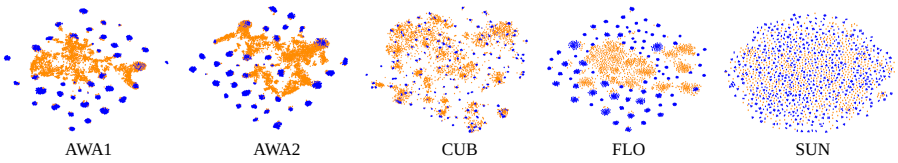

Fig. 4. The t-SNE visualization results for the learned latent space on the test sets of AWA1, AWA2, CUB, FLO and SUN. The blue dots represent the variables encoded from seen classes. The orange dots represent the variables encoded from unseen classes.

Table 4. Binary classification results of different training objective functions. We report the AUC and the FPR corresponding to $\gamma = 0.95$.

	AWA1		CUB	
Objective Function	**AUC**	**FPR**	**AUC**	**FPR**
$L_{f-SVAE} + L_{a-SVAE}$	62.5	93.3	56.1	88.5
$L_{f-SVAE} + L_{a-SVAE} + L_{cr}$	89.3	44.2	60.6	86.7
$L_{f-SVAE} + L_{a-SVAE} + L_{cls}$	94.9	15.7	98.2	9.2
$L_{overall}$	96.8	7.9	99.6	1.1

Ablation Study As defined in Eq.(10), the overall objective of our model consists of L_{f-SVAE}, L_{a-SVAE}, L_{cr} and L_{cls}. In this experiment, we analyze the impact of each term on AWA1 and CUB datasets. We report the AUC and the FPR scores on the threshold corresponding to $\gamma = 0.95$ for four objective functions in Table 4, where "+" stands for the combination of different terms. When there lacks of L_{cr} and L_{cls}, we observe that the first objective function only achieves 62.5% AUC score on AWA1, and 56.1% on CUB. The FPR score are 93.3% and 88.5%, respectively. It can be seen that the unseen samples can hardly be separated from the seem samples. When we further add L_{cr}, the AUC score increases to 89.3% and the FPR decreases to 44.2% on AWA1. However, the results only have small improvements on CUB dataset. We find that learning the modality invariant information helps to improve the OOD classification. But the improvement is influenced by the number of classes. When we add L_{cls} to the first objective function, the AUC score is improved to 94.9% on AWA1 and 98.2% on CUB. It can be seen that the classification loss heavily affects the binary classification. When we combine both L_{cr} and L_{cls}, the overall objective achieves the best OOD classification results.

Parameters Sensitivity The hyper-parameters in our approach are tuned by cross-validation. Fixing λ_f and λ_a to 0.1, we mainly tune α and β for our approach. Fig. 5 shows the AUC scores influenced by each hyper-parameter on AWA1 and CUB datasets, where each hyper-parameter is varied with the others are fixed. It can be seen that our method can work stably with different parameters.

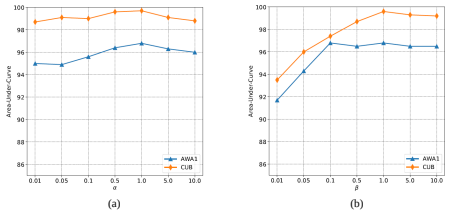

Fig. 5. Parameter sensitivity on AWA1 and CUB datasets.

Table 5. The influence of latent space dimension on the AUC score for AWA1 and CUB datasets.

Dimension	16	32	64	128	256
AWA1	95.2	95.7	96.8	90.5	86.2
CUB	95.8	96.5	99.6	97.7	95.1

Dimension of Latent Space In this analysis, we explore the robustness of our OOD classifier to the dimension of latent space. We report the AUC score in Table 5 with respect to different dimensions on AWA1 and CUB, ranging from 16, 32, 64, 128, and 256. We observe that the AUC score increases while we increase the latent space dimension and reaches the peak for both datasets at 64. When we continue to increase the dimension, the AUC score begins to decline, which indicates that increasing the dimension also may increase the risk of overfitting. For general consideration, we set the dimension to 64 for all datasets.

6 Conclusions

In this paper, we present an Out-of-Distribution classifier for the Generalized Zero-Shot learning problem. The proposed classifier is based on multi-modal hyper-spherical variational auto-encoders which learns a bounded manifold for each seen class in the latent space. By using the boundaries, we can separate the unseen samples from the seen samples. After that, we use two experts to classify the unseen samples and the seen samples separately. In this way, the GZSL problem is simplified to a ZSL problem and a conventional supervised classification problem. We extensively evaluate our approach on five benchmark datasets. The experimental results show that our approach surpasses state-of-the-art approaches by a significant margin.

Acknowledgements

This work was supported in part by Trico-Robot plan of NSFC under grant No.91748208, National Major Project under grant No.2018ZX01028-101, Shaanxi Project under grant No.2018ZDCXLGY0607, NSFC under grant No.61973246, and the program of the Ministry of Education for the university.

References

1. Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C.: Label-embedding for image classification. IEEE transactions on pattern analysis and machine intelligence **38**(7), 1425–1438 (2016)
2. Akata, Z., Reed, S., Walter, D., Lee, H., Schiele, B.: Evaluation of output embeddings for fine-grained image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2927–2936 (2015)
3. Annadani, Y., Biswas, S.: Preserving semantic relations for zero-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7603–7612 (2018)
4. Atzmon, Y., Chechik, G.: Adaptive confidence smoothing for generalized zero-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11671–11680 (2019)
5. Changpinyo, S., Chao, W.L., Gong, B., Sha, F.: Synthesized classifiers for zero-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5327–5336 (2016)
6. Chen, L., Zhang, H., Xiao, J., Liu, W., Chang, S.F.: Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1043–1052 (2018)
7. Cuturi, M.: Sinkhorn distances: Lightspeed computation of optimal transport. In: Advances in neural information processing systems. pp. 2292–2300 (2013)
8. Davidson, T.R., Falorsi, L., De Cao, N., Kipf, T., Tomczak, J.M.: Hyperspherical variational auto-encoders. 34th Conference on Uncertainty in Artificial Intelligence (UAI-18) (2018)
9. Felix, R., Kumar, V.B., Reid, I., Carneiro, G.: Multi-modal cycle-consistent generalized zero-shot learning. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 21–37 (2018)
10. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al.: Devise: A deep visual-semantic embedding model. In: Advances in neural information processing systems. pp. 2121–2129 (2013)
11. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
12. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. Proceedings of International Conference on Learning Representations (2017)
13. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
14. Kodirov, E., Xiang, T., Gong, S.: Semantic autoencoder for zero-shot learning. arXiv preprint arXiv:1704.08345 (2017)
15. Kumar Verma, V., Arora, G., Mishra, A., Rai, P.: Generalized zero-shot learning via synthesized examples. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4281–4289 (2018)
16. Lampert, C.H., Nickisch, H., Harmeling, S.: Attribute-based classification for zero-shot visual object categorization. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(3), 453–465 (2014)
17. Li, J., Jing, M., Lu, K., Zhu, L., Yang, Y., Huang, Z.: Alleviating feature confusion for generative zero-shot learning. In: Proceedings of the 27th ACM International Conference on Multimedia. pp. 1587–1595. ACM (2019)

18. Liu, Y., Gao, Q., Li, J., Han, J., Shao, L.: Zero shot learning via low-rank embedded semantic autoencoder. In: IJCAI. pp. 2490–2496 (2018)
19. Long, Y., Liu, L., Shao, L., Shen, F., Ding, G., Han, J.: From zero-shot learning to conventional supervised classification: Unseen visual data synthesis. In: CVPR. pp. 1627–1636 (2017)
20. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)
21. Mishra, A., Krishna Reddy, S., Mittal, A., Murthy, H.A.: A generative model for zero shot learning using conditional variational autoencoders. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 2188–2196 (2018)
22. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing. pp. 722–729. IEEE (2008)
23. Patterson, G., Xu, C., Su, H., Hays, J.: The sun attribute database: Beyond categories for deeper scene understanding. International Journal of Computer Vision **108**(1-2), 59–81 (2014)
24. Reed, S., Akata, Z., Lee, H., Schiele, B.: Learning deep representations of fine-grained visual descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 49–58 (2016)
25. Romera-Paredes, B., Torr, P.: An embarrassingly simple approach to zero-shot learning. In: International Conference on Machine Learning. pp. 2152–2161 (2015)
26. Schonfeld, E., Ebrahimi, S., Sinha, S., Darrell, T., Akata, Z.: Generalized zero-and few-shot learning via aligned variational autoencoders. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8247–8255 (2019)
27. Socher, R., Ganjoo, M., Manning, C.D., Ng, A.: Zero-shot learning through cross-modal transfer. In: Advances in neural information processing systems. pp. 935–943 (2013)
28. Tsai, Y.H.H., Huang, L.K., Salakhutdinov, R.: Learning robust visual-semantic embeddings. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 3591–3600. IEEE (2017)
29. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011)
30. Xian, Y., Lampert, C.H., Schiele, B., Akata, Z.: Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. IEEE transactions on pattern analysis and machine intelligence (2018)
31. Xian, Y., Lorenz, T., Schiele, B., Akata, Z.: Feature generating networks for zero-shot learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5542–5551 (2018)
32. Xu, X., Shen, F., Yang, Y., Zhang, D., Shen, H.T., Song, J.: Matrix tri-factorization with manifold regularizations for zero-shot learning. In: CVPR. pp. 3798–3807 (2017)
33. Zhang, F., Shi, G.: Co-representation network for generalized zero-shot learning. In: International Conference on Machine Learning. pp. 7434–7443 (2019)
34. Zhang, H., Koniusz, P.: Zero-shot kernel learning. In: CVPR. pp. 7670–7679 (2018)
35. Zhang, Z., Saligrama, V.: Zero-shot learning via semantic similarity embedding. In: Proceedings of the IEEE international conference on computer vision. pp. 4166–4174 (2015)