A Variational Autoencoder with Deep Embedding Model for Generalized Zero-Shot Learning

Peirong Ma, Xiao Hu*

School of mechanical and electrical engineering, Guangzhou University, Guangzhou, China
mpr_666@163.com, huxiao@gzhu.edu.cn

Abstract

Generalized zero-shot learning (GZSL) is a challenging task that aims to recognize not only unseen classes unavailable during training, but also seen classes used at training stage. It is achieved by transferring knowledge from seen classes to unseen classes via a shared semantic space (e.g. attribute space). Most existing GZSL methods usually learn a cross-modal mapping between the visual feature space and the semantic space. However, the mapping model learned only from the seen classes will produce an inherent bias when used in the unseen classes. In order to tackle such a problem, this paper integrates a deep embedding network (DE) and a modified variational autoencoder (VAE) into a novel model (DE-VAE) to learn a latent space shared by both image features and class embeddings. Specifically, the proposed model firstly employs DE to learn the mapping from the semantic space to the visual feature space, and then utilizes VAE to transform both original visual features and the features obtained by the mapping into latent features. Finally, the latent features are used to train a softmax classifier. Extensive experiments on four GZSL benchmark datasets show that the proposed model significantly outperforms the state of the arts.

Introduction

In recent years, deep learning techniques have developed rapidly, partly thanks to the widespread availability of large-scale labeled datasets. For example, in image classification, learning an excellent classification model always requires a sufficiently large number of manually labeled samples per category as a training set. However, the object categories in reality follow a long-tailed distribution. For some rare objects, only a limited number of labeled samples can be provided or even no samples available. In addition, new object categories may emerge dynamically. In this case, the performance of the deep neural network is significantly degraded. Therefore, how to train the classification model with a limited number of labeled samples or even no samples has attracted considerable interest in academia in recent years, which motivates the emergence of few-shot learning (Fei-Fei, Fergus, and Perona 2006) and zero-shot learning (ZSL) (Larochelle, Erhan, and Bengio 2008; Lampert, Nickisch, and Harmeling 2009), respectively.

ZSL aims to identify novel (unseen) classes that do not provide any training samples. In ZSL, the seen classes and the unseen classes are taken to train and test, respectively. However, the seen and unseen classes are completely different categories. The performance of ZSL depends entirely on the classification accuracy on the unseen classes, as shown in Figure 1(a). Since the objects that need to be recognized in real scenario may come from the unseen classes or from the seen classes instead of just from the unseen classes, ZSL is extended to GZSL (Scheirer et al. 2013). Like ZSL, only the seen classes are available while training for GZSL. Unlike ZSL, GZSL classifies not only the unseen classes but also the seen classes, as shown in Figure 1(b). The performance of GZSL depends on the harmonic mean of the seen and unseen classes' classification accuracy. Therefore, GZSL is more realistic and challenging than ZSL and this paper is dedicated to GZSL task. Since the images of the unseen classes cannot be obtained during the training phase, the semantic information, such as attributes (Farhadi et al. 2009; Ferrari and Zisserman 2008), word vectors (Mikolov et al. 2013) and sentence descriptions (Reed et al. 2016), shared between the classes is usually used to transfer knowledge from the seen classes to the unseen classes for realizing the unseen classes recognition. The semantic information is also called semantic embedding, class embedding or class prototype.

Most of the existing GZSL methods focus on building an embedding model that first learns the mapping between visual feature space and semantic space (Akata et al. 2013; Frome et al. 2013; Lampert, Nickisch, and Harmeling 2014; Romera-Paredes and Torr 2015; Reed et al. 2016; Zhang, Xiang, and Gong 2017; Sung et al. 2018; Wei et al.

*Corresponding author

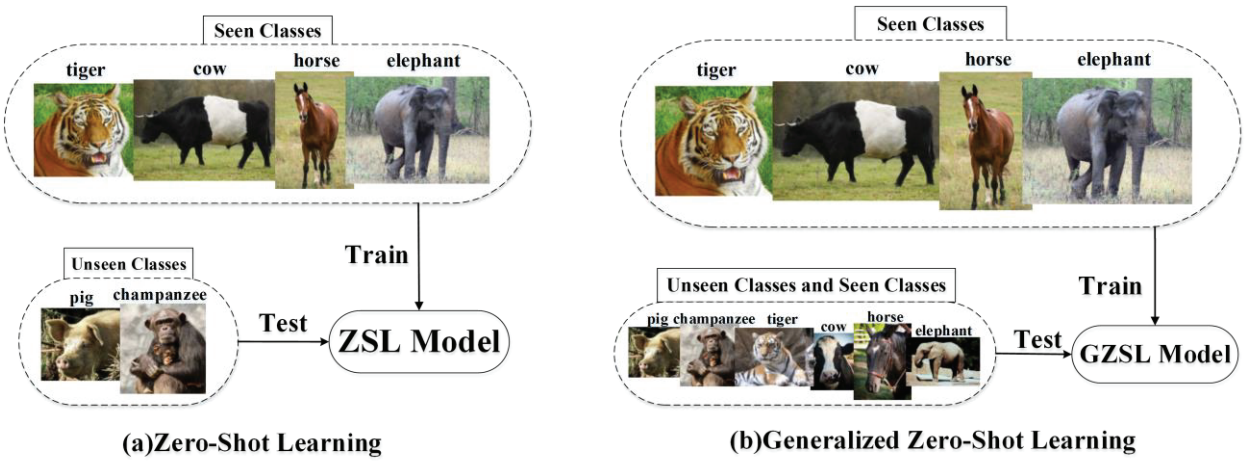


Figure 1: Comparison of zero-shot learning and generalized zero-shot learning.

2019). Then a nearest neighbor (NN) search is performed in the embedding space shared by image features and class embeddings to predict class labels. Since the seen classes and the unseen classes are completely different categories, the embedding model learned only from the seen classes will generate a bias when used in the unseen classes, that is, the projection domain shift (Fu et al. 2015). The essential cause of the bias problem is the lack of unseen samples.

Recently, some works have already regarded GZSL as a missing data problem (Mishra et al. 2018; Verma et al. 2018; Xian et al. 2018b; Li et al. 2019). In order to solve this problem, generative adversarial networks (GAN) (Goodfellow et al. 2014) or VAE (Kingma and Welling 2013) are firstly utilized to generate synthetic features of unseen classes from class embeddings to control the ratio between the seen and unseen samples. This process is called feature generation. Then the synthetic features along with the original features of the seen classes are taken to train a linear classifier. This way transforms GZSL into a traditional classification task, which alleviates data imbalance between the seen and unseen classes and achieves better performance than learning an embedding model. Due to the instability of GAN during training (Gulrajani et al. 2017), VAE become a better choice. Among those feature generation methods, a CADA-VAE model proposed by Schonfeld et al. (2019) transforms the class embeddings and visual features into a latent space of the VAE, which alleviates the projection domain shift and improve GZSL performance. However, it fails to learn the mapping between class embeddings and image features, there is still a large bias between latent features transformed from different embedding space.

In order to solve the problems mentioned above, this paper proposes a new model that combines a deep embedding network (DE) and a modified VAE (Named as DE-VAE), as shown in Figure 3. The proposed DE-VAE model firstly learns a mapping from semantic space to visual feature space via the deep embedding network. Secondly, both the features obtained by the mapping from class embeddings, and the original image features are input into the VAE to carry out cross-modal alignment. Thirdly, the class embeddings and image features are transformed into latent features by the trained deep embedding network and the encoder of VAE. Finally, a simple softmax classifier is trained using these latent features to implement GZSL.

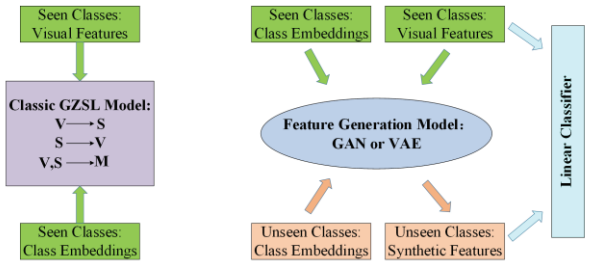
The contributions are as follows: (1) A DE-VAE model is proposed for GZSL. This model combines a deep embedding with a modified VAE to learn a latent space shared by multi-modal data, and then employs the discriminative latent features to train a powerful GZSL classifier. (2) To the best of our knowledge, the proposed model is the first to combine the classic GZSL model with the recently emerging feature generation model. This integration is simple, effective, easy to implement, and can be trained in an end-to-end manner. (3) Extensive experiments are carried out on four GZSL benchmark datasets, the results show that the proposed model significantly outperforms the state of the arts.

Related Work

In this section, we summarize the classic GZSL methods and the feature generation methods and explain their relationship to the proposed model.

Classic GZSL Methods

The classical GZSL methods first learns a mapping between the semantic space (S) and the visual feature space (V), as shown in Figure 2(a). The mapping direction can be divided into three main categories as follow:



(a) Classic GZSL Methods (b) Feature Generation Methods

Figure 2: The basic framework of classic GZSL methods and feature generation methods. The classic GZSL methods learn a mapping between semantic space and visual feature space, and the feature generation methods transform GZSL into a traditional classification task by generating synthetic features of the unseen class. V denotes visual feature space, S denotes semantic space and M denotes intermediate space.

(1) *V→S*. These methods use traditional regression, ranking models (Akata et al. 2013; Lampert, Nickisch, and Harmeling 2014) or deep neural network regression or ranking models (Frome et al. 2013; Reed et al. 2016) to learn the mapping function from the visual feature space to the semantic space.

(2) *S→V*. Selecting visual feature space as the embedding space: the mapping from the semantic space to the visual feature space (Zhang, Xiang, and Gong 2017; Sung et al. 2018). Among them, Zhang, Xiang, and Gong (2017) argues that the key to make ZSL models succeed is to choose the right embedding space. The Mapping from the semantic space to the visual space can alleviate the hubness problem (Radovanović, Nanopoulos, and Ivanović 2010). We adopt this conclusion, thus the deep embedding network of the proposed DE-VAE model learns a mapping from the semantic space to the visual feature space.

(3) *V, S→M*. The third direction is to map the semantic space and visual feature space to a common intermediate space (*M*) (Romera-Paredes and Torr 2015; Wei et al. 2019).

After training, a nearest neighbor (NN) search in the mapped space, i.e. embedding space, is performed to predict the class label. Specifically, given a test image *x*, the mapping model searches for the class embedding with the highest compatibility score, as follow:

$$f(x) = arg\max_{y} F(x, \theta(y); w) \quad (1)$$

Where *F* is the mapping model, *y* represents the class label, $\theta(y)$ represents the class embedding and *w* represents the parameters of the mapping model.

Feature Generation Methods

For the classic GZSL methods mentioned above, only the features of the seen classes are available during training, so the recognition accuracy of the seen classes is often much higher than that of the unseen classes. This large accuracy difference leads to a low harmonic mean *H*. Therefore, some researchers recently utilize GAN or VAE to generate synthetic features of the unseen classes from class embeddings, which we called feature generation methods, as shown in Figure 2(b). The feature generation methods can be divided into three steps as follows: (1) Training a feature generation model with image features and class embeddings of the seen classes. (2) Using the trained feature generation models to generate synthetic features of unseen classes from class embeddings. (3) The synthetic features of the unseen classes and the original features of the seen classes are used to train a linear classifier. The generation of the unseen class synthetic features mitigates the data imbalance between the seen and unseen classes and improves the recognition accuracy of the unseen classes, thus obtains a higher *H*.

GAN consists of a generator and a discriminator that compete in a turn-wise min-max game. The generator attempts to fool the discriminator via generating fake data that look like real data, and the discriminator determines the probability that the data is real or fake. VAE contains an encoder and a decoder (Kingma and Welling 2013). The encoder represents the input *x* as a latent variable *z* with Gaussian distribution assumption, and the decoder reconstructs the input from the latent variable as follows:

$$z \sim E(x) = q(z|x) \quad (2)$$

$$x' \sim D(z_x) = p(x|z) \quad (3)$$

where *E* represents the encoder and *D* represents the decoder. The VAE loss can be formulated as:

$$\ell = E_{q(z|x)}[logp(x|z)] - D_{KL}(q(z|x)||p(z)) \quad (4)$$

where the first term is the reconstruction error (REC) and the second term is the Kullback-Leibler divergence (KL-divergence) between $q(z|x)$ and $p(z)$. Let $q(z|x) = \mathcal{N}(\mu, \Sigma)$, so that the encoder learns μ and Σ, from which the latent vector z is generated by the reparameterization trick (Kingma and Welling 2013).

Among these feature generation methods, Xian et al. (2018b) and Li et al. (2019) use GAN, Mishra et al. (2018) and Verma et al. (2018) employs VAE from class embedding synthesizing CNN features of the unseen classes to solve GZSL task, which achieves better performance than classic GZSL methods. However, the training process of GAN is relatively unstable (Gulrajani et al. 2017). Instead of GAN, we employ VAE.

In this paper, the proposed DE-VAE model combines a deep embedding network with a modified VAE to learn a discriminative latent space to help classify, as shown in Figure 3. In other words, the proposed model combines the advantages of the classic GZSL methods with the feature

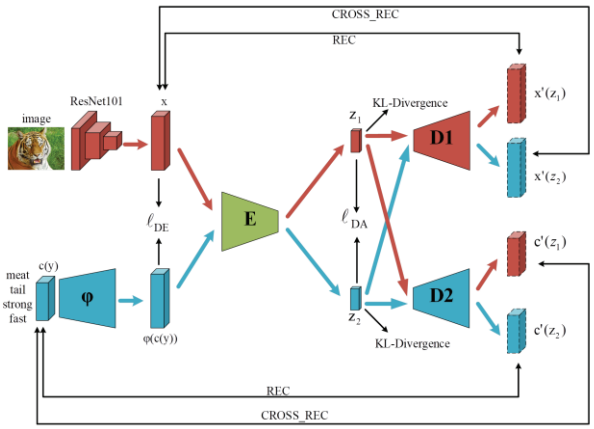


Figure 3: The proposed DE-VAE model contains a deep embedding network (φ) and a cross-modal alignment VAE. The VAE consists of an encoder (E) and two decoders ($D1$ and $D2$).

generation methods. Experimental results and ablation study in this paper demonstrate its effectiveness.

The Proposed DE-VAE Model

Assume X^S and X^U are the image features of seen classes and unseen classes, respectively. The image feature set: $X = X^S \cup X^U$. Y^S and Y^U are the labels of the seen classes and the unseen classes, respectively. We have $Y^S \cap Y^U = \emptyset$, that is, the training classes and the test classes are disjoint. Given a train set of images: $\{(x, y)|x \in X^S,\ y \in Y^S\}$ and their class embedding set: $C = \{c(Y)|\forall y \in Y^S \cup Y^U\}$. In ZSL, the goal is to learn an image classifier $f_{ZSL} : X^U \rightarrow Y^U$, while in GZSL, the task is to learn a classifier $f_{GZSL} : X \rightarrow Y^U \cup Y^S$.

The architecture of the proposed DE-VAE model is shown in Figure 3. It consists of a deep embedding network φ and a cross-modal alignment VAE with an encoder (E) and two decoders ($D1$ and $D2$).

Deep Embedding Network

The first component of the proposed DE-VAE model is a deep embedding network, whose purpose is to learn the mapping from semantic space to visual feature space. Specifically, it takes the semantic embedding vector of the corresponding class as input, and after passing through two fully connected (FC) linear + Rectified Linear Unit (ReLU) layers, outputs a visual embedding vector $\varphi(c(y))$, which has the same dimensions as the visual feature vector of this class, optimizing:

$$\ell_{DE} = \sum_{i=1}^{N} \left\| x_i - \varphi\left(c(y_i)\right) \right\|^2 \quad (5)$$

where x_i is the visual feature vector of the i^{th} training image and $c(y_i)$ is the semantic embedding vector of the i^{th} training image.

Cross-Modal Alignment VAE

The modified VAE in the proposed DE-VAE model is made of an encoder and two decoders, which aims to align data from two different modalities (i.e. image feature and class embedding) in the latent space. As shown in Figure 3. After mapping class embedding into the visual feature space by the deep embedding network, the encoder E transforms the original image feature x and the feature obtained from class embedding vector $\varphi(c(y))$ into low-dimensional latent vectors z_1 and z_2, respectively. Then, through decoders $D1$ and $D2$, z_1 and z_2 are reconstructed into $x'(z_1)$ and $c'(z_2)$, which are the same dimensions as the input image feature and class embedding, respectively. In this process, optimizing:

$$\ell_{VAE} = \sum_{i=1}^{M} E_{q_\phi(z|x)}[\, log p_\theta(x^{(i)}|z)] - \beta D_{KL}(q_\phi(z|x^{(i)})||p_\theta(z)) \quad (6)$$

where β is a hyperparameter to control the weighting of the KL-Divergence (Higgins et al. 2017). There are two kinds of data modalities: image feature and semantic embedding, so M=2, $x^{(1)} \in X^S, x^{(2)} \in c(Y^S)$.

In addition, we minimize the Wasserstein distance between the latent multivariate Gaussian distributions of the two modal data, called Distribution-Alignment (DA) (Schonfeld et al. 2019). The DA loss is:

$$\ell_{DA} = \sum_{i}^{M} \sum_{j \neq i}^{M} (||\mu_i - \mu_j||_2^2 + ||\Sigma_i^{\frac{1}{2}} - \Sigma_j^{\frac{1}{2}}||_{Frobenius}^2)^{\frac{1}{2}} \quad (7)$$

In order to make the modality-specific autoencoder further learn similar representations across modalities, we also perform a Cross-Reconstruction (CROSS-REC). That is, every modality-specific decoder obtains reconstructions by decoding the latent feature vector of a sample from another modality (but the same class), called Cross-Alignment (CA). The CA loss is:

$$\ell_{CA} = \sum_{i}^{M} \sum_{j \neq i}^{M} \left| x^{(j)} - D_j\left(E\left(x^{(i)}\right)\right) \right| \quad (8)$$

where $M = 2$, and D_j is the decoder of j^{th} modality. When $i = 1$ and $j = 2$, $x^{(1)} \in X^S, x^{(2)} \in c(Y^S)$; when $i = 2$ and $j = 1$, $x^{(1)} \in X^S, x^{(2)} \in \varphi(c(Y^S))$.

The overall loss of the proposed DE-VAE model is defined as:

$$\ell_{DE-VAE} = \ell_{VAE} + \lambda \ell_{DE} + \alpha \ell_{DA} + \gamma \ell_{CA} \quad (9)$$

where λ, α and γ are the weighting factors of the deep embedding network, distribution-alignment and cross-alignment loss, respectively.

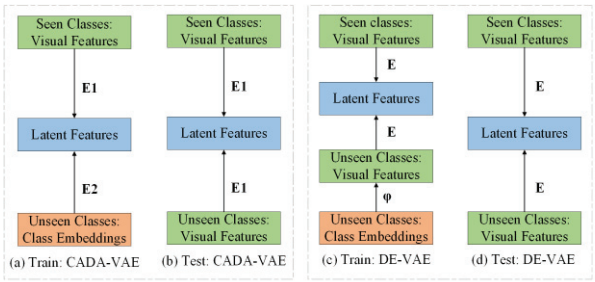


Figure 4: Comparison of the proposed model (DE-VAE) with CADA-VAE when using trained model to transform information from two different modalities in seen and unseen classes into low-dimensional latent features for training and testing of the softmax classifier. $E1$ and $E2$ represent the two encoders of CADA-VAE, φ represents the deep embedding network and E represents the encoder of the proposed DE-VAE model.

Comparison of Baseline VAE and DE-VAE Model

CADA-VAE (Schonfeld et al. 2019) takes image feature and class embedding as the input of VAE directly, so it requires two different encoder($E1$ and $E2$) instead of our E. CADA-VAE achieves stable training and excellent GZSL performance. However, it fails to learn the mapping between class embeddings and visual features before inputting them into encoders of VAE. Although cross-alignment and distribution-alignment are performed, there is still a large bias problem. Specifically, at the softmax classifier training stage, the seen classes latent features are obtained from the image features of the seen classes using $E1$, and the unseen classes latent features are obtained from the class embeddings of the unseen classes using $E2$. However, at test time, the latent features are all obtained from image features (whether seen classes or unseen classes) by $E1$. Therefore, the latent features obtained from two different embedding space will result in an inherent bias, as shown in Figure 4.

The proposed DE-VAE model effectively alleviates this problem. Before the image features and class embeddings are fed into the VAE, the proposed DE-VAE model first maps them to the same embedding space. Therefore, the DE-VAE model only uses one encoder E. This improvement is simple and effective, and can narrow the inherent bias between seen and unseen classes latent features, thus make the generated latent features more discriminative.

Implementation Details

The deep embedding network (φ), encoder (E) and decoders ($D1$ and $D2$) in the proposed model are implemented as multilayer perceptron (MLP) with one hidden layer. We use 1200 and 1560 hidden units for the deep embedding network and the encoder, respectively. In addition, the $D1$ and $D2$ have 1450 and 660 hidden units, respectively. λ is set to 0.9, α and γ follow the settings in (Schonfeld et al. 2019). The size of the latent feature is 70 (CUB), 80 (SUN), 65 (AWA1 and AWA2), respectively. The model is trained for 100 epochs using the Adam optimizer, and the batch size is 50. We uses the L_2 distance to construct the loss of the deep embedding network, and all other losses use the L_1 distance. After the model training, the image features and class embeddings from the seen and unseen classes are transformed into the latent space via the trained deep embedding network φ and the encoder E, then the training and testing of the final Softmax classifier is performed using the latent features.

Table 1: Details of the number of seen and unseen classes images at training time and test time. S represents the number of seen classes images and U represents the number of unseen classes images.

Number of Images		Train		Test	
Dataset	Total	S	U	S	U
CUB	11788	7057	0	1764	2967
SUN	14340	10320	0	2580	1440
AWA1	30475	19832	0	4958	5685
AWA2	37322	23527	0	5882	7913

Experiments

In this section, we first detail the benchmark datasets and the evaluation protocol used in the experiments, then present the result of the proposed model and compare it with other models. Finally, we conduct an ablation study and compare performance of ZSL to further verify the effectiveness of our model.

Datasets

The proposed model is evaluated on four widely used ZSL/GZSL benchmark datasets, namely Caltech-UCSD-Birds 200-2011 (CUB) (Wah et al. 2010), SUN Attribute (SUN) (Patterson et al. 2012), Animals with Attributes 1 (AWA1) (Lampert et al. 2014) and Animals with Attributes 2 (AWA2) (Xian et al. 2018a). CUB is a medium-scale fine-grained dataset with 312 attributes. Among the total number of 200 classes, there are 150 seen classes and 50 unseen classes. SUN is a fine-grained and medium-scale dataset with 102 attributes, which contains 645 seen classes and 72 unseen classes, a total of 717 classes. AWA1 is a medium-scale coarse-grained dataset with 85 attributes. Its image comes from 50 animal categories, 40 of which are seen classes and the other 10 are unseen classes. Images of the AWA1 dataset are unavailable due to copyright restrictions. In order to visually study the categories in the AWA1 dataset, Xian et al. (2018a) introduced the AWA2 dataset, which has the same 50 animal categories and 85 attributes as AWA1 dataset.

Table 2: Comparing DE-VAE with the state of the art (GZSL). Top: classic GZSL methods; Middle: feature generation methods. We report average per-class top-1 accuracy for unseen (U) and seen (S) classes and their harmonic mean (H) - all results are shown in percentage. The best results are highlighted with bold numbers. Since the results on AWA2 dataset were not reported in the papers of f-CLSWGAN and LisGAN, we implemented them using the author's codes, the results are highlighted with italic numbers.

Methods	CUB			SUN			AWA1			AWA2		
	U	S	**H**	U	S	**H**	U	S	**H**	U	S	**H**
ALE (Akata et al. 2013)	23.7	62.8	34.4	21.8	33.1	26.3	16.8	76.1	27.5	14.0	81.8	23.9
DeViSE (Frome et al. 2013)	23.8	53.0	32.8	16.9	27.4	20.9	13.4	68.7	22.4	17.1	74.7	27.8
SJE (Akata et al. 2015)	23.5	59.2	33.6	14.7	30.5	19.8	11.3	74.6	19.6	8.0	73.9	14.4
ESZSL (Paredes and Torr 2015)	12.6	63.8	21.0	11.0	27.9	15.8	6.6	75.6	12.1	5.9	77.8	11.0
LATEM (Xian et al. 2016)	15.2	57.3	24.0	14.7	28.8	19.5	7.3	71.7	13.3	11.5	77.3	20.0
SAE (Kodirov, Xiang, and Gong 2017)	7.8	54.0	13.6	8.8	18.0	11.8	1.8	77.1	3.5	1.1	82.2	2.2
DEM (Zhang, Xiang, and Gong 2017)	19.6	57.9	29.2	20.5	34.3	25.6	32.8	84.7	47.3	30.5	86.4	45.1
SP-AEN (Chen et al. 2018)	34.7	70.6	46.6	24.9	38.6	30.3	-	-	-	23.3	90.9	37.1
CVAE-ZSL (Mishra et al. 2018)	-	-	34.5	-	-	26.7	-	-	47.2	-	-	51.2
SE-GZSL (Verma et al. 2018)	41.5	53.3	46.7	40.9	30.5	34.9	56.3	67.8	61.5	58.3	68.1	62.8
f-CLSWAGAN (Xian et al. 2018b)	43.7	57.7	49.7	42.6	36.6	39.4	57.9	61.4	59.6	*53.8*	*68.2*	*60.2*
LisGAN (Li et al. 2019)	46.5	57.9	51.6	42.9	37.8	40.2	52.6	76.3	62.3	*54.3*	*68.5*	*60.6*
CADA-VAE (Schonfeld et al. 2019)	51.6	53.5	52.4	47.2	35.7	40.6	57.3	72.8	64.1	55.8	75.0	63.9
Baseline (Schonfeld et al. 2019)	50.8	54.4	52.5	44.4	36.2	39.9	55.5	74.5	63.6	54.3	78.1	64.1
DE-VAE (ours)	52.5	56.3	**54.3**	45.9	36.9	**40.9**	59.6	76.1	**66.9**	58.8	78.9	**67.4**

Setting and Evaluation Protocol

For all datasets in our experiment, the DE-VAE model uses attribute vectors as class embeddings. In addition, all image features are extracted using the 2048-dim top pooling units of ResNet101 (He et al. 2016) pre-trained on ImageNet 1K (Deng et al. 2009) and not fine-tuned. In order to make fair comparisons with other methods and avoid violating the zero-shot assumption (i.e. test classes need to be disjoint with the classes in ImageNet 1k), we follow the splits and evaluation protocol proposed by Xian et al. (2018a). In the splits proposed by Xian et al. (2018a), the details of the number of seen and unseen classes images at training and test time are shown in Table 1. Since we don't use the original image features of the unseen classes during training, our method is inductive rather than transductive (Fu et al. 2015; Guo et al. 2018). Xian et al. (2018a) believe that the use of harmonic mean H in the GZSL task can take into account the performance of both the seen classes and the unseen classes. Define acc_s as the average per class top-1 accuracy of the seen classes, acc_u as the average per-class top-1 accuracy of the unseen classes, then the harmonic mean:

$$H = \frac{2 * acc_s * acc_u}{acc_s + acc_u} \tag{10}$$

Comparing with the State-of-the-Art

In the GZSL setting, we compare the proposed DE-VAE model with 13 the best performing recent methods. The results are shown in Table 2. Among them, the classic GZSL methods ALE (Akata et al. 2013), DeViSE (Frome et al. 2013), SJE (Akata et al. 2015), ESZSL (Paredes and Torr 2015), LATEM (Xian et al. 2016), SAE (Kodirov, Xiang, and Gong 2017), DEM (Zhang, Xiang, and Gong 2017) and SP-AEN (Chen et al. 2018) learn a linear or nonlinear mapping between semantic space and image feature space. On the other hand, the feature generation methods CVAE-ZSL (Mishra et al. 2018), SE-GZSL (Verma et al. 2018), f-CLSWGAN (Xian et al. 2018b), LisGAN (Li et al. 2019) and CADA-VAE (Schonfeld et al. 2019) treat GZSL as a missing data problem, and use VAE or GAN to generate synthetic features of unseen classes from class embeddings. DE-VAE combines the strengths of these two approaches. The results show that DE-VAE is superior to all other models on all datasets. It can also be seen that the feature generation methods outperform the classic GZSL methods obviously. Since f-VAEGAN-D2 (Xian et al. 2019) operates in transductive zero-shot setting and fine-tune RseNet-101 on each benchmark dataset, so we don't compare with it in Table 2. However, under the same settings (inductive and no fine-tuning), our performance is better than the results reported in the paper of f-VAEGAN-D2, as follows: 54.3% vs 53.6% on CUB, 40.9% vs 41.3% on SUN, 66.9% vs 63.5% on AWA1.

Our method is based on CADA-VAE (Schonfeld et al. 2019), for fair comparisons, we re-implemented it in the same environment as ours. In this paper, the CADA-VAE results we re-implemented are called **Baseline**. The accuracy difference between our model and the Baseline is as follows: 54.3% vs 52.5% on CUB, 40.9% vs 39.9% on SUN, 66.9% vs 63.6% on AWA1, 67.4% vs 64.1% on AWA2. This is because our model projects class embeddings into the visual feature space before inputting class

Table 3: The results of ablation study. We compare GZSL accuracy when using different components of our model alone. The best results are highlighted with bold numbers.

Model	CUB			SUN			AWA1			AWA2		
	U	S	**H**	U	S	**H**	U	S	**H**	U	S	**H**
DE	21.7	48.6	30.0	19.9	38.2	26.2	34.9	83.9	49.3	32.3	86.5	47.0
VAE	50.8	54.4	52.5	44.4	36.2	39.9	55.5	74.5	63.6	54.3	78.1	64.1
DE-VAE	52.5	56.3	**54.3**	45.9	36.9	**40.9**	59.6	76.1	**66.9**	58.8	78.9	**67.4**

Table 4: Comparison of ZSL accuracy (per-class top-1 accuracy) - all results are shown in percentage. The best results are highlighted with bold numbers. References are the same as Table 2.

Method	CUB	SUN	AWA1	AWA2
ALE	54.9	58.1	59.9	62.5
DeViSE	52.0	56.5	54.2	59.7
SJE	53.9	53.7	65.6	61.9
ESZSL	53.9	54.5	58.2	58.6
LATEM	49.3	55.3	55.1	55.8
SAE	33.3	40.3	53.0	54.1
DEM	51.7	61.9	68.4	67.1
SP-AEN	55.4	59.2	-	58.5
CVAE-ZSL	52.1	61.7	**71.4**	65.8
SE-GZSL	59.6	63.4	69.5	69.2
f-CLSWGAN	57.3	60.8	68.2	-
LisGAN	58.8	61.7	70.6	-
CADA-VAE	60.4	61.8	62.3	64.0
Baseline	60.6	62.8	65.0	64.3
DE-VAE	**63.1**	**64.0**	69.4	**69.3**

embeddings and image features into VAE, which narrows the inherent bias between seen classes and unseen classes latent features, and further alleviates the projection domain shift. The latent features we generate are more discriminative and more conducive to classification.

Ablation Study

The key strength of our model comes from the combination of the classic GZSL model and the feature generation model. In order to evaluate how important this integration is, we conduct an ablation study. Specifically, we use the deep embedding network (DE), the variational autoencoders (VAE) and the DE-VAE to train a classifier for GZSL, respectively. The results in Table 3 show that our DE-VAE achieves the highest accuracy on all four GZSL benchmark datasets compared to DE and VAE. It confirms that our model is effective and can significantly improve GZSL performance. In addition, compared with DE, the harmonic mean (H) significantly increases when using VAE. This is because the unseen classes latent features are generated by VAE, which leads to a more balanced data distribution and the learned classifier is not heavily biased to seen classes.

We also perform ablation experiments for the hyper-parameter λ. That is, we fixed the others hyper-parameters and only changed the value of λ to perform experiments on CUB dataset. The results are as follows: λ=0.01(H=51.99), λ=0.1 (H=53.10), λ=1 (H=54.23), λ=10 (H=53.94), λ=100 (H=52.60). In this paper, λ = 0.9 (H = 54.36).

Performance of ZSL

Although this work focused on the more practical and challenging GZSL, to further verify the effectiveness of the proposed DE-VAE model, we also experiment in the legacy ZSL setting. The results in Table 4 show that DE-VAE model provides competitive ZSL performance.

Conclusion

In this paper, a novel GZSL model is proposed, which integrates a deep embedding network with a modified cross-modal alignment VAE. The key to the proposed DE-VAE model is to embed image features and class embeddings into a same space before transforming them into latent feature through the encoder. Extensive experiments show that the proposed DE-VAE model achieves the most advanced performance on four widely used GZSL benchmark dataset, which confirms its effectiveness in learning discriminative latent features and further mitigating the bias problem. In the future, we will consider extending the proposed DE-VAE model to other cross-modal problems or transfer learning tasks.

References

Akata, Z.; Perronnin, F.; Harchaoui, Z.; and Schmid, C. 2013. Label-embedding for attribute-based classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 819-826.

Akata, Z.; Reed, S.; Walter, D.; Lee, H.; and Schiele, B. 2015. Evaluation of output embeddings for fine-grained image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2927-2936.

Chen, L.; Zhang, H.; Xiao, J.; Liu, W.; and Chang, S. F. 2018. Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1043-1052.

Deng, J.; Dong, W.; Socher, R.; Li, L. J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248-255.

Farhadi, A.; Endres, I.; Hoiem, D.; and Forsyth, D. 2009. Describing objects by their attributes. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 1778-1785.

Fei-Fei, L.; Fergus, R.; and Perona, P. 2006. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence* 28(4): 594-611.

Ferrari, V., and Zisserman, A. 2008. Learning visual attributes. In *Advances in neural information processing systems*, 433-440.

Frome, A.; Corrado, G. S.; Shlens, J.; Bengio, S.; Dean, J.; and Mikolov, T. 2013. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, 2121-2129).

Fu, Y.; Hospedales, T. M.; Xiang, T.; and Gong, S. 2015. Transductive multi-view zero-shot learning. *IEEE transactions on pattern analysis and machine intelligence* 37(11): 2332-2345.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; ... and Bengio, Y. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, 2672-2680.

Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; and Courville, A. C. 2017. Improved training of wasserstein gans. In *Advances in neural information processing systems*, 5767-5777.

Guo, Y.; Ding, G.; Jin, X.; and Wang, J. 2016. Transductive zero-shot recognition via shared model space learning. In *Thirtieth AAAI Conference on Artificial Intelligence*, 3494-3500.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770-778.

Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; ... and Lerchner, A. 2017. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. *ICLR*, 2(5), 6.

Kingma, D. P., and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kodirov, E.; Xiang, T.; and Gong, S. 2017. Semantic autoencoder for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3174-3183.

Kumar Verma, V.; Arora, G.; Mishra, A.; and Rai, P. 2018. Generalized zero-shot learning via synthesized examples. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4281-4289.

Lampert, C. H.; Nickisch, H.; and Harmeling, S. 2009. Learning to detect unseen object classes by between-class attribute transfer. In 2009 IEEE Conference on Computer Vision and Pattern Recognition, 951-958.

Lampert, C. H.; Nickisch, H.; and Harmeling, S. 2014. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36(3): 453-465.

Larochelle, H.; Erhan, D.; and Bengio, Y. 2008. Zero-data learning of new tasks. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, 646-651.

Li, J.; Jin, M.; Lu, K.; Ding, Z.; Zhu, L.; and Huang, Z. 2019. Leveraging the Invariant Side of Generative Zero-Shot Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, 3111-3119.

Mishra, A.; Krishna Reddy, S.; Mittal, A.; and Murthy, H. A. 2018. A generative model for zero shot learning using conditional variational autoencoders. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2188-2196.

Patterson, G.; Xu, C.; Su, H.; and Hays, J. 2014. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision* 108(1-2): 59-81.

Radovanović, M.; Nanopoulos, A.; and Ivanović, M. 2010. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research* 11(Sep): 2487-2531.

Reed, S.; Akata, Z.; Lee, H.; and Schiele, B. 2016. Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 49-58.

Romera-Paredes, B., and Torr, P. 2015. An embarrassingly simple approach to zero-shot learning. In *International Conference on Machine Learning*, 2152-2161.

Scheirer, W. J.; de Rezende Rocha, A.; Sapkota, A.; and Boult, T. E. 2012. Toward open set recognition. *IEEE transactions on pattern analysis and machine intelligence* 35(7): 1757-1772.

Schonfeld, E.; Ebrahimi, S.; Sinha, S.; Darrell, T.; and Akata, Z. 2019. Generalized zero-and few-shot learning via aligned variational autoencoders. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8247-8255.

Sung, F.; Yang, Y.; Zhang, L.; Xiang, T.; Torr, P. H.; and Hospedales, T. M. 2018. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1199-1208.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology.

Wei, K.; Yang, M.; Wang, H.; Deng, C.; and Liu, X. 2019. Adversarial Fine-Grained Composition Learning for Unseen Attribute-Object Recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 3741-3749.

Xian, Y.; Akata, Z.; Sharma, G.; Nguyen, Q.; Hein, M.; and Schiele, B. 2016. Latent embeddings for zero-shot classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 69-77.

Xian, Y.; Lampert, C. H.; Schiele, B.; and Akata, Z. 2018a. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence.*

Xian, Y.; Lorenz, T.; Schiele, B.; and Akata, Z. 2018b. Feature generating networks for zero-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5542-5551.

Xian, Y.; Sharma, S.; Schiele, B.; and Akata, Z. 2019. f-VAEGAN-D2: A feature generating framework for any-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 10275-10284.

Zhang, L.; Xiang, T.; and Gong, S. 2017. Learning a deep embedding model for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021-2030.