LEARNING SHARED MANIFOLD REPRESENTATION OF IMAGES AND ATTRIBUTES FOR GENERALIZED ZERO-SHOT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

The most prior methods of zero-shot learning have realized predicting labels of unseen images by learning a mapping from images to pre-defined class-attributes. However, recent studies show that these approaches severely suffers from the issue of *biased* prediction under the more realistic generalized zero-shot learning (GZSL) scenarios, i.e., their classifier tends to predict all the examples from both seen and unseen class as one of the seen classes. The cause of this problem is that we can not obtain training data of the unseen class and that the representation of attributes is poor. To solve this, we propose a concept to learn a mapping that embeds both images and attributes to a space that is *robust* to such representations and *generalized* even for unseen data, which we refer to *shared manifold learning*. Furthermore, we propose *modality invariant variational autoencoders*, which can perform shared manifold learning by training variational autoencoders with both images and attributes as inputs. The empirical validation of well-known datasets in GZSL shows that our method achieves the significantly superior performances to the existing relation-based works.

1 INTRODUCTION

The recent high performance of deep neural networks on image classification and object recognition depends greatly on whether we can obtain sufficiently labeled images of classes to predict. However, it is difficult to operate such a AI system as-is in the real world because the number of existing classes is huge and, as long as human beings create or develop new objects, their number can be expected to continue to increase day by day, resulting in difficulty obtaining labeled data of all these classes. Human beings have the excellent ability to infer objects that have hardly seen by exploiting and transferring the knowledge learned in the past so as to survive in the harsh real world. In the machine learning context, this problem is formulated as few-shot learning.

Zero-shot learning (ZSL) is a more extreme problem setting of few-shot learning proposed by Lampert et al. (2009), which is to train by labeled dataset from certain classes called *seen* classes, and then to predict completely *unseen* classes not included in the trained dataset. It is usually accomplished by preparing pre-defined semantic representations of all classes such as attributes, and learning the mapping from images to class-attributes (Lampert et al., 2009; 2014; Frome et al., 2013; Akata et al., 2013; 2015; Romera-Paredes & Torr, 2015; Kodirov et al., 2017). However, this conventional setting has a strong premise that the test dataset does not include the seen classes. That setting is unnatural in application to the real world because the objects we encounter are not always new but are rather likely to have been experienced in the past. Therefore, to apply the ZSL system to the real world, one should change to a more natural setting that the test dataset can include examples of both seen and unseen classes, which is known as generalized zero-shot learning (GZSL) (Chao et al., 2016).

Although the setting of GZSL is more natural, it has been confirmed that existing relation-based approaches suffer greatly in terms of performance (Chao et al., 2016; Xian et al., 2018). The main reason for this difficulty is that the training dataset does not include examples of the unseen class, leading to *biased* prediction of attributes. Therefore, to archive GZSL, it is necessary to consider how to *generalize* to unseen data without training them. Moreover, Chao et al. (2016) found that improvement of the representation of attributes contributes to the higher performance of GZSL.

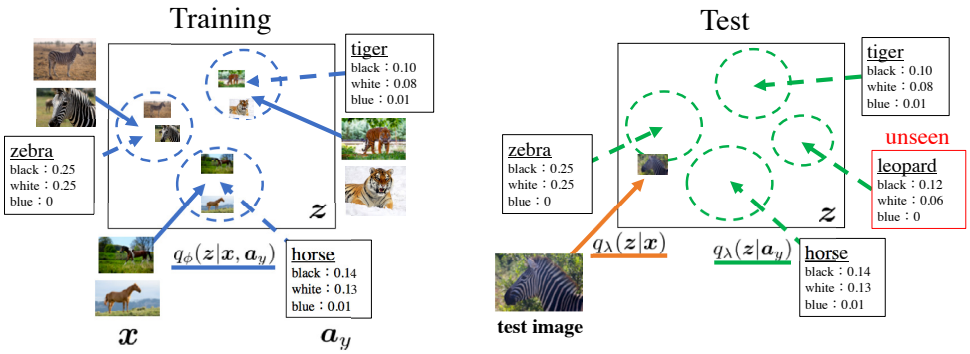

Figure 1: An overview of shared manifold learning with MIVAE to achieve GZSL. At training time, we learn a mapping to latent space $\boldsymbol{z}$, $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a})$, so that it integrates image $\boldsymbol{x}$ and class-attributes $\boldsymbol{a}_y$, and properly places even unseen data. Note that mappings of each of the image and the attributes, $q_\lambda(\boldsymbol{z}|\boldsymbol{x})$ and $q_\lambda(\boldsymbol{z}|\boldsymbol{a}_y)$, are also learned to approximate this mapping, $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a})$. At test time, we can properly arrange the attributes of the unseen classes on the latent space using the learned mapping of the attributes, $q_\lambda(\boldsymbol{z}|\boldsymbol{a}_y)$. Therefore, using the learned mapping of the image, $q_\lambda(\boldsymbol{z}|\boldsymbol{x})$, the test image can be embedded in the vicinity of the correct class-attributes.

However, because we can not always obtain such an appropriate representation, it seems better to learn more *robust* representation from them.

In light of them, we propose a new solution of GZSL called *shared manifold learning*. In shared manifold learning, we learn a mapping that embeds each image and attribute in the same space and representation so that it is robust against representation of inputs, i.e., *shared representation learning* of images and attributes. In addition, we make sure that this mapping is also generalized, which means the inputs of the unseen class are embedded in separate places from examples of other classes. That is, it can be regarded as *manifold learning* of shared representation. If such shared manifold learning can be done properly, we can estimate the corresponding class-attributes from the test image by embedding it in the shared manifold representation whether the class of this image is seen or unseen, that is, the bias problem can be solved.

To realize this, we propose modality invariant variational autoencoders (MIVAE), which are based on variational autoencoders (VAE) (Kingma & Welling, 2013; Rezende et al., 2014). In this method, first of all, to obtain shared representations from inputs of images $\boldsymbol{x}$ and class-attributes $\boldsymbol{a}_y$ with different dimensions and distributions, we define mappings from each input to the latent space $\boldsymbol{z}$ as the deep probabilistic distributions, $q_\lambda(\boldsymbol{z}|\boldsymbol{x})$ and $q_\lambda(\boldsymbol{z}|\boldsymbol{a}_y)$, which enable nonlinear embedding considering different distributions of the respective inputs. Next, to learn manifold representation from these two inputs, we prepare a VAE with both images and attributes as inputs. In learning of MIVAE, we not only maximize the lower bound of this VAE but also learn so that each mapping approximates the encoder of VAE, $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a})$, and so that each mapping also approaches another, which enables us to capture the mappings that embed inputs in the shared manifold representations. See Figure 1 as the outline of shared manifold learning with MIVAE. If we consider images and attributes as different *modalities*, then we can regard MIVAE as an extension of works examining multimodal VAE (Suzuki et al., 2016; Vedantam et al., 2017).

As a method for solving the bias-related shortcoming of GZSL, a synthesis-based method has been proposed recently (Mishra et al., 2017; Verma et al., 2018), which is to generate images from class-attributes with the conditional deep generative model and to learn the classifier from the generated image to the labels of seen and unseen classes. Although this method is known to give high performance to GZSL, it is necessary to generate a plurality of images for each class. Moreover, it is difficult to extend because the representation of the input image is restricted to make it easier to generate. Therefore, we believe that our proposed method is more widely available.

The contribution of this research is as follows.

- We discuss the reasons why existing methods do not work in GZSL and propose shared manifold learning as a concept to solve the problem.
- We propose MIVAE, which extended VAE as a method to realize shared manifold learning.

- We confirm that MIVAE can perform shared manifold learning properly and that it can contribute to the higher performance of GZSL compared to existing ZSL methods based on shared representation learning.
- We demonstrate that MIVAE has markedly higher performance than state-of-the-art relation-based ZSL methods on several benchmark datasets.

2 PROBLEM FORMULATION

We assume that the dataset $\mathcal{D}_{tr} = \{\boldsymbol{x}_i, y_i\}_{i=1}^{N_{tr}}$ is given as the training set, where $\boldsymbol{x}_i \in \mathcal{X}$ is the input data, e.g. an image, and $y_i \in \mathcal{Y}_s = \{1..., S\}$ is the corresponding label data. The objective of ZSL is to train the classifier using $\mathcal{D}_{tr}$ and to predict labels $\hat{y}_j \in \mathcal{Y}$ from the example $\boldsymbol{x}_j \in \mathcal{X}$ in the test set $\mathcal{D}_{ts} = \{\boldsymbol{x}_j\}_{j=1}^{N_{ts}}$. In the original ZSL, it is assumed that the classes of the test set are completely unseen in the training set, which means $\mathcal{Y} = \mathcal{Y}_u = \{S+1, ..., U\}$. However, our goal is to train the setting of GZSL, which includes both seen and unseen classes in the test set ($\mathcal{Y} = \mathcal{Y}_s \cap \mathcal{Y}_u$).

In the context of ZSL, we also assume that we have the class-attribute matrix $\boldsymbol{A} \in \mathbb{R}^{M \times (S+U)}$ as the semantic information of classes, where each column represents the M-dimensional attribute vector $\boldsymbol{a}_c \in \mathcal{A} = \mathbb{R}^M$ of each class $c = 1, ..., S+U$. Using this attribute vector, the training set $\mathcal{D}_{tr}$ can be replaced as $\{\boldsymbol{x}_i, \boldsymbol{a}_{y_i}\}_{i=1}^{N_{tr}}$.

3 PROPOSED METHOD

3.1 SHARED MANIFOLD LEARNING

GZSL is expected to be a more difficult problem than conventional ZSL because the number of candidate classes to predict becomes much greater. In fact, it is confirmed that the usual method of learning the relation between images and attributes severely fails to predict unseen classes (Chao et al., 2016; Xian et al., 2018). The direct reason for this difficulty is that the training dataset does not include examples of the unseen class, leading to biased map that classifies all images into only seen classes. Moreover, Chao et al. (2016) argues that improvement of the representation of attributes is vital for GZSL, because replacing the attributes with the image features as ideal semantic representation improves the performance of GZSL. However, we can not always obtain such an appropriate semantic representation. In this paper, we consider a learning method of another space that is robust to its representation and also generalizes to unseen data.

First, we consider mapping from each image and attribute to the same space, not mapping from images to attributes, so that the relation between images and attributes is robust against the representation of the attributes. At this time, we enforce the images and attributes to be the same representations in the embedding space (shared representation learning). This representation not only becomes robust against attribute representation but also becomes more informative by integrated with the images. Next, we make sure that the shared representation also generalizes to unseen data (manifold learning of shared representation). For example, given the image and attributes of the unseen class, we make sure that they retain the same representation in the embedding space, and that they are embedded in separate places from examples of other classes. For this study, we designate these series of learning as *shared manifold learning* of images and attributes. We claim that we can deal with the bias-related shortcoming of GZSL by learning shared manifold representation.

3.2 MULTIMODAL INVARIANT VARIATIONAL AUTOENCODER

Next, we propose modality invariant VAE (MIVAE), which is a method of learning mapping from two inputs to a space to be shared and manifold representations.

Here, we consider that the dataset of ZSL $\{\boldsymbol{x}_i, \boldsymbol{a}_{y_i}\}_{i=1}^{N_{tr}}$ has different *modalities*, which means that $\boldsymbol{x}$ and $\boldsymbol{a}_y$ are different in their spaces and distributions, i.e., $\mathcal{X} \neq \mathcal{A}$ and $p(X) \neq p(A)$,where $X = \{x_1, ..., x_{N_{tr}}\}$ and $A = \{a_{y_1}, ..., a_{y_{N_{tr}}}\}$.

At first, we consider two Gaussian distributions parameterized with deep neural networks $q_\lambda(\boldsymbol{z}|\boldsymbol{x})$ and $q_\lambda(\boldsymbol{z}|\boldsymbol{a}_y)$ as mappings from each of $\mathcal{X}$ and $\mathcal{A}$ to $\mathcal{Z}$, where λ is the parameter of the distribution. See Appendix A for the way of parameterization with deep neural networks.

In MIVAE, we aim to learn so that these mappings embed inputs in shared manifold representation. Note that it is important to set these mappings to the deep probabilistic distributions in obtaining shared representations. For a detailed discussion on this, see Appendix C.

Next, we use variational autoencoders (VAEs) (Kingma & Welling, 2013; Rezende et al., 2014), which has been widely used as a model of representation learning and manifold learning. Moreover, we consider VAE with two modalities as input, which is called joint VAE (Vedantam et al., 2017).

Joint VAE has a latent concept $\boldsymbol{z}$ that is common to the two modalities $\boldsymbol{x}, \boldsymbol{a}_y$, each of which is generated conditionally and independently from the concept. That is, their generative processes are $\boldsymbol{z} \sim p(\boldsymbol{z})$ and $\boldsymbol{x}, \boldsymbol{a}_y \sim p_\theta(\boldsymbol{x}, \boldsymbol{a}_y|\boldsymbol{z}) = p_\theta(\boldsymbol{x}|\boldsymbol{z})p_\theta(\boldsymbol{a}_y|\boldsymbol{z})$, where θ corresponds to model parameters. These processes indicate that the latent variable $\boldsymbol{z}$ can be regarded as a shared representation containing different modality information.

To train this, we should maximize the joint distribution $p(\boldsymbol{x}, \boldsymbol{a}_y) = \int p_\theta(\boldsymbol{x}, \boldsymbol{a}_y|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z}$ over the training set. However, it is difficult to perform this maximization directly because this distribution is intractable. Therefore, we instead maximize the following evidence lower bound (ELBO).

$$\mathcal{L}_{vae} = -D_{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a}_y)||p(\boldsymbol{z})) + E_{q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a}_y)}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] + E_{q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a}_y)}[\log p_\theta(\boldsymbol{a}_y|\boldsymbol{z})], \quad (1)$$

where $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a}_y)$ is an approximate distribution of posterior and ϕ represents parameters. We call $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a}_y)$ as encoder and $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ and $p_\theta(\boldsymbol{a}_y|\boldsymbol{z})$ as decoder. These models are parameterized by deep neural networks. The distribution of the encoder is Gaussian, and that of the decoder depends on the distribution of the input data.

From the encoder of trained VAE, we can extract shared representations that integrate the information of different modalities. In addition, the VAE can obtain the manifold representation of the input in the latent space and appropriately place the unseen input on the latent space. Because we use images and attributes as inputs to joint VAE here, we can obtain the encoder that integrates and maps them into manifold representation, i.e., *we can perform shared manifold learning*.

Therefore, if we train each modality's distribution to minimize the distance between the encoder, then we can obtain such representations from the distribution of each modality. We define this distance using Kullback-Liebler (KL) divergence as

$$\mathcal{L}_{dist} = D_{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a})||q_\lambda(\boldsymbol{z}|\boldsymbol{x})) + D_{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a})||q_\lambda(\boldsymbol{z}|\boldsymbol{a})). \quad (2)$$

Intuitively, because the encoder has more information to be conditioned, it seems that the mapping distributions, especially $q_\lambda(\boldsymbol{z}|\boldsymbol{a})$), might degenerate if each is brought closer to the encoder. Actually, Vedantam et al. (2017) showed that minimization of $E_{p_{data}(\boldsymbol{x},\boldsymbol{a})}[D_{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a})||q_\lambda(\boldsymbol{z}|\boldsymbol{a}))]$ is equivalent to minimization of $E_{p_{data}(\boldsymbol{x},\boldsymbol{a})}[D_{KL}(q_\phi^{avg}(\boldsymbol{z}|\boldsymbol{a})||q_\lambda(\boldsymbol{z}|\boldsymbol{a}))]$, where $q_\phi^{avg}(\boldsymbol{z}|\boldsymbol{a}) = E_{p_{data}(\boldsymbol{x}|\boldsymbol{a})}[q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a})]$, i.e., this is the average of the encoder over all $\boldsymbol{x}$ corresponding to $\boldsymbol{a}$. Therefore, the minimization of equation 2 encourages each mapping to close to the distribution complementing the other modality of the encoder, thereby resulting in stabilization.

When equation 2 becomes minimum, $q_\lambda(\boldsymbol{z}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a}) = q_\lambda(\boldsymbol{z}|\boldsymbol{a})$ is satisfied. However, if the objective function of VAE is also optimized simultaneously, then two mapping distributions might not be brought close to each other explicitly during training. Therefore, to force the two distributions to be explicitly close at the training time, we also minimize the following equation:

$$\mathcal{L}_{map} = D_{KL}(q_\lambda(\boldsymbol{z}|\boldsymbol{x})||q_\lambda(\boldsymbol{z}|\boldsymbol{a})). \quad (3)$$

Therefore, the objective function of MIVAE is

$$\mathcal{L} = -\mathcal{L}_{vae} + \lambda_{dist}\mathcal{L}_{dist} + \lambda_{map}\mathcal{L}_{map}, \quad (4)$$

where λ_{dist} and λ_{map} are hyperparameters. When $\lambda_{map} = 0$, this objective function is the same as JMVAE (Suzuki et al., 2016), which means that this model can be regarded as an extension of JMVAE.

After training MIVAE, we can predict the labels of test images simply as $\hat{y}_j = \arg\min_{y \in \mathcal{Y}} D_{KL}(q_\lambda(\boldsymbol{z}|\boldsymbol{x}_j)||q_\lambda(\boldsymbol{z}|\boldsymbol{a}_y))$.

4 Related works

ZSL is one of the greatest attention problems among image recognition tasks, and various methods have been proposed. Here, we explain them separately on relation-based and synthetic-based approaches.

Table 1: Difference in approaches of ZSL.

Method	Type	Training Cost	Complex Data	Evading Biased Problem
Shared Manifold Learning	Relation-based	Low	Yes	Yes
Image Attribute Projection	Relation-based	Low	Yes	No
Shared Representation Learning	Relation-based	Low	Yes	No
Image Syntesis	Syntesis-based	High	No	Yes

4.1 RELATION-BASED

Relation-based methods are intended to train relations between images and class-attributes $F(x, a_y)$ from training data and to predict a class whose relation is strong in test data, i.e. $\hat{y} = F(x, a_y)$. This approach can be classified further into the following two methods: training of mapping from image space to attribute space (image-attribute projection) and training of shared representation that embed both image and attribute space (shared representation learning).

Image-attribute projection In ZSL, the simplest and most widely adopted approach is to obtain mapping from image space to class-attribute space with training data. Some early studies used classifiers such as SVM to learn mapping from images to class attributes (Lampert et al., 2009; 2014), but many other methods use linear embedding into attributes (or semantic representations) and typically set a ranking loss as the objective function to train such embedding (Frome et al., 2013; Akata et al., 2013; 2015; Romera-Paredes & Torr, 2015; Kodirov et al., 2017). Actually, Romera-Paredes & Torr (2015) adds a regularization term of the mapping to this objective function. Also, Kodirov et al. (2017) minimizes the embedding error in both directions with the objective. Recently, nonlinear models are also used(Socher et al., 2013) to consider nonlinearity between images and attributes. However, classifiers tend to predict only one class from many images at testing, which is known as the hubness problem because of the difference in the distribution of the two modalities. To avoid this difficulty, some works have proposed training of reverse mapping from attribute to image space with such as deep neural networks (Zhang et al., 2017). Verma & Rai (2017) suggests that mapping from attributes to an image are a conditional distribution by exponential family rather than deterministic, which can incorporate consideration of uncertainty among modalities. Therefore the hubness problem can be relaxed further.

Shared representation learning Another means of obtaining relations is to learn mapping that embeds images and attributes into a shared latent representation. This learning can be realized by shared representation learning in which images and attributes are regarded as different modalities. In ZSL, several methods of learning shared embedding spaces have been proposed Zhang & Saligrama (2015; 2016), but few methods use deep neural networks as embedding maps. VZSL (Wang et al., 2017) trains shared representations of images and attributes using deep generative models, which most closely approximates our method. As with Verma & Rai (2017), uncertainty can be taken into consideration, and furthermore, deep neural networks are used.

Although these relation-based methods work well in conventional ZSL settings, in GZSL, their performance is much worse, as shown in Chao et al. (2016). As described in Chapter 1, one of the reason is that the representation of attributes is poor. The approach of shared representation learning is robust to the representation of attributes, but it does not perform manifold learning at two inputs, which means that it does not perform shared manifold learning. Indeed, VZSL performs manifold learning only with image input, which results in poor generalization performance in examples of unseen classes, as shown later in an experiment.

4.2 SYNTHESIS-BASED

Recently, several synthesis-based methods have been proposed (Verma et al., 2018; Mishra et al., 2017). These approaches are attempts to generate images from class-attributes and to train a classifier of unseen classes by these synthesis samples. To generate samples from attributes properly, they train conditional deep generative models of images given class-attributes in training data. It is apparently difficult to generate realistic test images that are useful to train the classifier. However, in many ZSL settings, it is common to use feature extraction representations from images as input so that they can generate these inputs with high quality.

The greatest advantage of the synthesis-based approaches is that they can achieve high performance in a GZSL setting because we train classification with a new classifier after generating all samples of seen or unseen classes from the trained deep generative model. For that reason, we no longer need

Table 2: Summary of datasets of our experiments.

Dataset	Images	Attributes	Seen/Unseen classes
SUN	14340	102	150/50
CUB	11788	312	645/72
AwA1	30475	85	40/10
AwA2	37322	85	40/10
aPY	15339	64	20/12

to consider differences between seen and unseen classes. However, training the classifier requires generation of multiple images of all classes. Therefore, if the image size increases or the number of classes increases, then it might be costly to generate these synthetic samples. Moreover, this method works well only if the input representations of images are easy to generate. In recent years, several deep generative models have been proposed that can produce high-quality images (Kingma & Dhariwal, 2018; Karras et al., 2017), but it still might be difficult to generate arbitrary types of inputs to be useful for classification of unseen classes.

Therefore, in this paper, we take the relation-based approach and propose the concept of shared manifold learning to deal with the issue of GZSL. Table 1 presents a brief comparison of our method with others.

5 Experiments

For experiment, we use the following four datasets commonly used for ZSL: Animals with Attributes (AWA1) (Lampert et al., 2014), CUB-200-2011 Bird (CUB) (Wah et al., 2011), SUN Attribute (SUN) (Patterson & Hays, 2012), and Attribute Pascal and Yahoo (aPY) (Farhadi et al., 2009). In addition, because the images of AWA1 were not public available, Xian et al. (2018) additionally introduced Animals with Attributes 2 (AWA2) composed of images collected from public sites. Therefore, in this experiment, we will use five datasets including AWA2. Each dataset contains an image of each class, where each class is represented by semantic attributes. Table 2 presents the numbers of images, attributes, and classes in respective datasets. For fair comparisons, we use 2048-dim top-layer embedding of the 101-layered ResNet (He et al., 2016) provided by Xian et al. (2018) as the image vector. Furthermore, for the class-attribute representation we use the continuous valued attributes between 0 and 1 provided with each dataset. We followed Xian et al. (2018) for the splits of images of each class at training and test times.

For the metric of evaluation at the test time, we use the average per-class top-1 accuracy on both the seen and unseen classes (referred as $acc_{\mathcal{Y}_s}$ and $acc_{\mathcal{Y}_u}$). In addition, to evaluate the performance on GZSL, we calculate the harmonic mean of $acc_{\mathcal{Y}_s}$ and $acc_{\mathcal{Y}_u}$, which is $H = (2 \cdot acc_{\mathcal{Y}_s} \cdot acc_{\mathcal{Y}_u})/(acc_{\mathcal{Y}_s} + acc_{\mathcal{Y}_u})$.

We used the Adam optimization algorithm (Kingma & Ba, 2014) with a learning rate of 10^{-3}. Also, in order to suppress over-fitting, we applied batch normalization to each layer and added L2 regularization to the parameters, where the coefficient of this term is 10^{-4}. For details of network architectures, see the appendix. In all experiments, we set $\lambda_{dist} = 1$, which works well in multi-modal learning according to reports of related studies (Suzuki et al., 2016; Vedantam et al., 2017). For another parameter λ_{map}, we will verify the effect by changing this value in the experiments. All models in this paper were implemented with PyTorch (Paszke et al., 2017).

We conduct two experiments. First, compared with shared representation learning, we demonstrate that MIVAE with shared manifold learning can learn with high performance qualitatively and quantitatively in GZSL. The second experiment compares our method with the state-of-the-art relation-based methods and confirms that our method has much higher performance than those methods.

5.1 Comparison with shared representation learning methods

To the best of our knowledge, VZSL (Wang et al., 2017) is the only existing method of ZSL used to train shared representation learning with deep probabilistic embedding maps, $q_\lambda(\boldsymbol{z}|\boldsymbol{x})$ and $q_\lambda(\boldsymbol{z}|\boldsymbol{a})$. Furthermore, we consider the following three methods including VZSL as shared representation learning to compare.

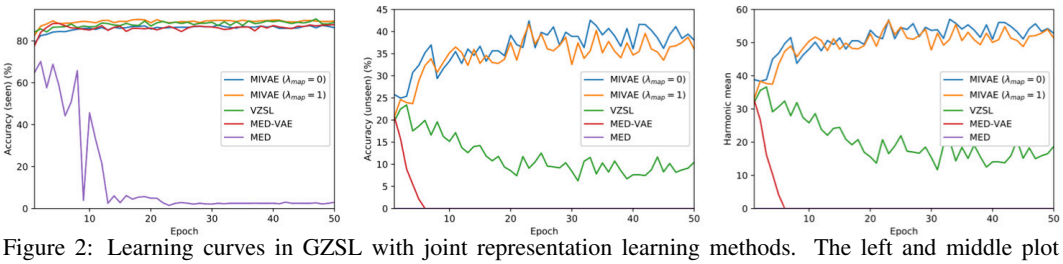

Figure 2: Learning curves in GZSL with joint representation learning methods. The left and middle plot respectively present the accuracy of seen and unseen classes in the test data. The right shows the harmonic mean evaluation.

Minimization of embedded divergence (MED) This simply minimizes KL divergence of the two distributions, i.e. the loss function of MED is $D_{KL}(q_\lambda(\boldsymbol{z}|\boldsymbol{x})||q_\lambda(\boldsymbol{z}|\boldsymbol{a}))$.

MED-VAE To train the manifold representation in the latent space, we add the reconstruction error of the VAE whose input is the image vector to that of the MED, i.e., $-E_{q_\lambda(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] + D_{KL}(q_\lambda(\boldsymbol{z}|\boldsymbol{x})||q_\lambda(\boldsymbol{z}|\boldsymbol{a}))$.

VZSL (Wang et al., 2017) Wang et al. (2017) proposed margin regularization that encourages separation of the true class from the next class. The loss function of VZSL is the addition of the margin regularization to that of MED-VAE. See Wang et al. (2017) for more mathematical details.

This experiment used AWA2 for the dataset. The distribution of the embedding maps was Gaussian. The networks structures were set the same in all methods. Furthermore, the prediction methods of the classes of each example in the test data are all the same: $\hat{y}_j = \arg\min_{y\in\mathcal{Y}} D_{KL}(q_\lambda(\boldsymbol{z}|\boldsymbol{x}_j)||q_\lambda(\boldsymbol{z}|\boldsymbol{a}_y))$.

Figure 2 shows learning curves obtained with these three methods and MIVAE in GZSL. First, for seen classes, all models except MED have nearly equal accuracy. However, because the accuracy of MED is markedly low, it is apparent that manifold learning on latent representation by VAEs is important. In MIVAE, we confirm that the accuracy score is higher when λ_{map} is set to 1.

Next, the result of the unknown classes shows that both the MED and the MED-VAE deteriorate considerably during early training. Regarding MED, failure to train generalized representations during training caused bias to known classes, leading to bad prediction in unknown classes. However, in MED-VAE, accuracy becomes very poor despite manifold training of images, probably because we were unable to train the connection between attributes and images on the embedding space. VZSL tries to link images and corresponding attributes more closely during training, which contributes to making them better than MED-VAE. However, as training progresses, the accuracy drops, i.e., the training becomes unstable.

Finally, it is apparent that MIVAE is markedly more accurate than these methods. From these results, we confirmed that, to predict the unseen class in the GZSL setting it might be important to train a shared manifold representation that integrates images and attributes. Moreover, we found that MIVAE can properly obtain such a representation.

The result of the harmonic mean is almost identical to that of the accuracy of unseen classes. In addition, in these AWA2 experiments, the MIVAE's results on accuracy of the unseen classes and the harmonic mean are apparently less dependent on the λ_{map} value.

Figure 3 shows representations mapped respectively by the embedding distribution trained with MED-VAE, VZSL, and MIVAE. From this result as well, it is apparent that MIVAE can obtain representations of seen and unseen classes in the latent space.

These results clarify that shared representation learning is insufficient in GZSL and clarify that shared manifold learning is important for high performance.

5.2 Comparison with state-of-the-art methods

Next, we compare MIVAE with state-of-the-art relation-based ZSL. All results of these methods were reprinted from Xian et al. (2018).

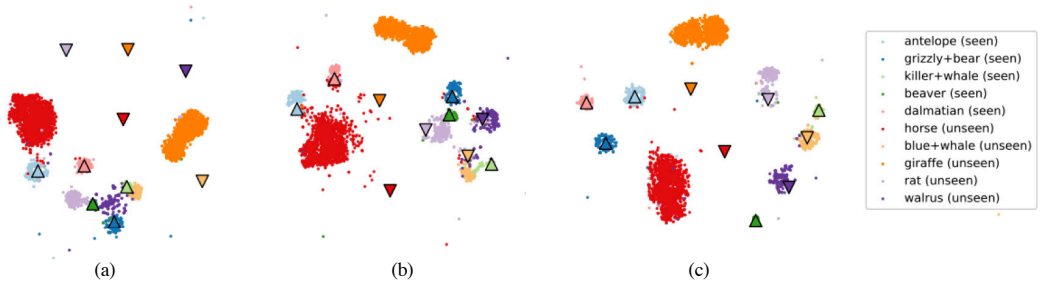

Figure 3: t-SNE plots of joint latent representations. After selecting five seen classes and five unseen classes from AWA2, we embedded their images and attributes. Circle plots are the embedding of the images. Triangles represent embedding of the attributes ($\triangle$ denotes a seen class and $\bigtriangledown$ is an unseen class). Each color of the plot corresponds to a different class. Additionally, the normal triangles represent the seen classes. Inverted triangles denote the unseen classes: (a) MED-VAE, (b) VZSL, and (c) MIVAE, where $\lambda_{map} = 1$.

Table 3: Results in the GZSL setting. We evaluate them by using the harmonic mean of class-mean top-1 accuracy (%) on unseen and seen classes. We also show the average of the harmonic mean in all datasets.

Methods	**SUN**	**CUB**	**AWA1**	**AWA2**	**aPY**	**Average**
CONSE (Szegedy et al., 2015)	11.6	3.1	0.8	1.0	0.0	3.3
CMT (Socher et al., 2013)	11.8	12.6	1.8	1.0	2.8	6.0
SSE (Zhang & Saligrama, 2015)	4.0	14.4	12.9	14.8	0.4	9.1
LATEM (Xian et al., 2016)	19.5	24.0	13.3	20.0	0.2	15.4
ALE (Akata et al., 2016)	26.3	34.4	27.5	23.9	8.7	24.2
DEVISE (Frome et al., 2013)	20.9	32.8	22.4	27.8	9.2	20.3
SJE (Akata et al., 2015)	19.8	33.6	19.6	14.4	6.9	18.9
ESZSL (Romera-Paredes & Torr, 2015)	15.8	21.0	12.1	11.0	4.6	12.9
SYNC (Changpinyo et al., 2016)	13.4	19.8	16.2	18.0	13.3	16.1
SAE (Kodirov et al., 2017)	11.8	13.6	3.5	2.2	0.0	6.2
GFZSL (Verma & Rai, 2017)	0.0	0.0	3.5	4.8	0.0	1.7
MIVAE ($\lambda_{map} = 0$)	26.9	36.8	**51.2**	**53.2**	**27.1**	39.1
MIVAE ($\lambda_{map} = 1$)	**32.8**	**41.4**	51.0	48.2	26.0	**39.9**

As Table 3 shows, our proposed method largely outperforms all existing methods of relation-based. Particularly, Table 3 shows that performance is poor in datasets with few seen classes such as AWA (both 1 and 2) and aPY. By contrast MIVAE, despite being a relation-based method, can properly generalize unseen classes even in these difficult datasets. Relation-based methods are known to tend to be biased to seen classes, so that the results of unseen classes become considerably worse (Chao et al., 2016). However, since our results are much greater than those, we can see that our approach solves the bias problem of GZSL. Furthermore, for seen classes, the performance of our method is equal to or higher than that of other methods. For harmonic mean, the proposed method is far superior to the others. Also, for seen classes, our method is equal to or higher than other methods, so that for harmonic mean, the proposed method is much superior to the others. For the results of accuracy on both seen and unseen classes, see Appendix D.

In MIVAE, the result of $\lambda_{map} = 1$ is better than that of $\lambda_{map} = 0$ in SUN and CUB, but there is not much difference between them in terms of AWA and aPY; rather it is better when $\lambda_{map} = 1$. This result is probably related to the number of seen classes in the dataset, i.e., if the number of seen classes is large, a positive influence on prediction of the seen class, e.g. minimizing equation 3 at training time, will also have a positive influence on that of the unseen classes.

Note that we do not reprint the results of VZSL in Wang et al. (2017) to Table 3 because the training/test split and the evaluation metric in Wang et al. (2017) is different from here. In our reproduction of VZSL in the same split and metric here, the harmonic means in each dataset and their average (SUN, CUB, AWA1, AWA2, aPY, Average) are (20.9, 30.7, 16.7, 18.7, 11.6, 19.7) respectively, which are all inferior to our method.

Finally, we evaluated the performance in the conventional ZSL setting. As Table 4 shows, our proposed method has nearly the same performance as that of existing methods in most datasets. Therefore, our method, shared manifold learning, deals *directly* with the bias-related shortcoming of a relation-based approach on GZSL.

Table 4: Results in the conventional ZSL setting. We evaluate them by using class-mean top-1 accuracy (%) on unseen classes.

Methods	**SUN**	**CUB**	**AWA1**	**AWA2**	**aPY**	**Average**
CONSE (Szegedy et al., 2015)	38.8	34.3	45.6	44.5	26.9	38.0
CMT (Socher et al., 2013)	39.9	34.6	39.5	37.9	28.0	36.0
SSE (Zhang & Saligrama, 2015)	51.5	43.9	60.1	61.0	34.0	50.1
LATEM (Xian et al., 2016)	55.3	49.3	55.1	55.8	35.2	50.1
ALE (Akata et al., 2016)	58.1	54.9	59.9	62.5	39.7	55.0
DEVISE (Frome et al., 2013)	56.5	52.0	54.2	59.7	**39.8**	52.4
SJE (Akata et al., 2015)	53.7	53.9	65.6	61.9	32.9	53.6
ESZSL (Romera-Paredes & Torr, 2015)	54.5	53.9	58.2	58.6	38.3	52.7
SYNC (Changpinyo et al., 2016)	56.3	55.6	54.0	46.6	23.9	47.3
SAE (Kodirov et al., 2017)	40.3	33.3	53.0	54.1	8.3	37.8
GFZSL (Verma & Rai, 2017)	60.6	49.3	**68.3**	63.8	38.4	**56.1**
MIVAE ($\lambda_{map} = 0$)	**60.8**	51.8	64.1	64.1	32.8	54.7
MIVAE ($\lambda_{map} = 1$)	60.2	**56.5**	63.1	**64.7**	35.4	56.0

6 Conclusion

In this paper, we addressed the setting of generalized zero-shot learning (GZSL) and specifically demonstrated that shared manifold learning is important to resolve the bias-related shortcoming. Furthermore, to demonstrate that point, we proposed modality invariant variational autoencoders (MIVAE), which realizes shared manifold learning using variational autoencoders that take both images and attributes as inputs. We first compared MIVAE with some ZSL methods that perform only shared representation learning. Thereby, we confirmed that performing manifold learning with both images and attributes is important for good performance of GZSL. Next, we experimented with several benchmark datasets. Results show that MIVAE has markedly higher performance than state-of-the-art relation-based ZSL methods, which suggests that the bias difficulty can be mitigated. Furthermore, the fact that the performance in the conventional ZSL is not much different from that of the existing method means that MIVAE can directly address difficulties posed by GZSL. Actually, MIVAE is a robust and extensible method for the input information format such that we can address more realistic and complicated problems. For example, it might be possible to accommodate text information as more sophisticated side information of classes with an encoder-decoder RNN model.

References

Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for attribute-based classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 819–826, 2013.

Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. 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.

Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE transactions on pattern analysis and machine intelligence*, 38(7): 1425–1438, 2016.

Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5327–5336, 2016.

Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *European Conference on Computer Vision*, pp. 52–68. Springer, 2016.

Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 1778–1785. IEEE, 2009.

Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Tomas Mikolov, et al. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pp. 2121–2129, 2013.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Diederik P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *arXiv preprint arXiv:1807.03039*, 2018.

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

Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. *arXiv preprint arXiv:1704.08345*, 2017.

Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 951–958. IEEE, 2009.

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

Ashish Mishra, M Reddy, Anurag Mittal, and Hema A Murthy. A generative model for zero shot learning using conditional variational autoencoders. *arXiv preprint arXiv:1709.00663*, 2017.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 2751–2758. IEEE, 2012.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

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

Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *Advances in neural information processing systems*, pp. 935–943, 2013.

Masahiro Suzuki, Kotaro Nakayama, and Yutaka Matsuo. Joint multimodal learning with deep generative models. *arXiv preprint arXiv:1611.01891*, 2016.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

Ramakrishna Vedantam, Ian Fischer, Jonathan Huang, and Kevin Murphy. Generative models of visually grounded imagination. *arXiv preprint arXiv:1705.10762*, 2017.

Vinay Kumar Verma and Piyush Rai. A simple exponential family framework for zero-shot learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 792–808. Springer, 2017.

Vinay Kumar Verma, Gundeep Arora, Ashish Mishra, and Piyush Rai. Generalized zero-shot learning via synthesized examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

Wenlin Wang, Yunchen Pu, Vinay Kumar Verma, Kai Fan, Yizhe Zhang, Changyou Chen, Piyush Rai, and Lawrence Carin. Zero-shot learning via class-conditioned deep generative models. *arXiv preprint arXiv:1711.05820*, 2017.

Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 69–77, 2016.

Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

Li Zhang, Tao Xiang, Shaogang Gong, et al. Learning a deep embedding model for zero-shot learning. 2017.

Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via semantic similarity embedding. In *Proceedings of the IEEE international conference on computer vision*, pp. 4166–4174, 2015.

Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via joint latent similarity embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6034–6042, 2016.

A Parameterization of distributions with deep neural networks

The Gaussian distribution can be parameterized with deep neural networks as

$$\begin{aligned}
&\mathcal{N}(\boldsymbol{z}; \boldsymbol{\mu}, \mathrm{diag}(\boldsymbol{\sigma}^2)),\\
&\boldsymbol{\mu} = f_\mu(f_{\mathrm{MLP}}(\boldsymbol{x})),\\
&\boldsymbol{\sigma}^2 = \mathrm{Softplus}(f_{\sigma^2}(f_{\mathrm{MLP}}(\boldsymbol{x}))),
\end{aligned}$$

where f_μ and f_{σ^2} are respectively denote linear single layer neural networks and where f_{MLP} represents a deep neural network with an arbitrary number of layers. Moreover, applying the softplus function for each element of a vector is denoted as $\mathrm{Softplus}$.

The Bernoulli distribution is parameterized as

$$p_\theta(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{B}(\boldsymbol{x}; \boldsymbol{\mu}), \boldsymbol{\mu} = \mathrm{Sigmoid}(f_\mu(f_{\mathrm{MLP}}(\boldsymbol{z}))),$$

where $\mathrm{Sigmoid}$ is represents the sigmoid function.

B Network architectures

For the notation of model structures, we denote a linear fully-connected layer with k units, batch normalization, and ReLU as `DkBR`. Also, we denote `DkBR` without batch normalization and ReLU as `Dk`. In addition, the process of applying `J` after `I` is denoted as `I-J`, and the process of concatenating the last layers of the two networks `I, J` into one layer is denoted as `(I,J)`.

Therefore, the network structures of distributions of MIVAE are as follows:

- $p(\boldsymbol{x}|\boldsymbol{z})$ (Gaussian, where $\sigma^2 = 1$)
  - f_μ: `D2048`
  - f_{MLP}: `z-D1000BR`
- $p(\boldsymbol{a}|\boldsymbol{z})$ (Bernoulli)

Table 5: Results in the GZSL setting. Each of **u** and **s** represents class-mean top-1 accuracy (%) on unseen and seen classes and **H** means the harmonic mean of **u** and **s**. We also show the average of **H** in all datasets.

	SUN			CUB			AWA1			AWA2			aPY			Average
Methods	**u**	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**	**H**
CONSE	6.8	39.9	11.6	1.6	72.2	3.1	0.4	**88.6**	0.8	0.5	**90.6**	1.0	0.0	**91.2**	0.0	3.3
CMT	8.1	21.8	11.8	7.2	49.8	12.6	0.9	87.6	1.8	0.5	90.0	1.0	1.4	85.2	2.8	6.0
SSE	2.1	36.4	4.0	8.5	46.9	14.4	7.0	80.5	12.9	8.1	82.5	14.8	0.2	78.9	0.4	9.1
LATEM	14.7	28.8	19.5	15.2	57.3	24.0	7.3	71.7	13.3	11.5	77.3	20.0	0.1	73.0	0.2	15.4
ALE	21.8	33.1	26.3	23.7	62.8	34.4	16.8	76.1	27.5	14.0	81.8	23.9	4.6	73.7	8.7	24.2
DEVISE	16.9	27.4	20.9	23.8	53.0	32.8	13.4	68.7	22.4	17.1	74.7	27.8	4.9	76.9	9.2	20.3
SJE	14.7	30.5	19.8	23.5	59.2	33.6	11.3	74.6	19.6	8.0	73.9	14.4	3.7	55.7	6.9	18.9
ESZSL	11.0	27.9	15.8	12.6	**63.8**	21.0	6.6	75.6	12.1	5.9	77.8	11.0	2.4	70.1	4.6	12.9
SYNC	7.9	**43.3**	13.4	11.5	70.9	19.8	8.9	87.3	16.2	10.0	90.5	18.0	7.4	66.3	13.3	16.1
SAE	8.8	18.0	11.8	7.8	54.0	13.6	1.8	77.1	3.5	1.1	82.2	2.2	0.0	83.3	0.0	6.2
GFZS	0.0	39.6	0.0	0.0	45.7	0.0	1.8	80.3	3.5	2.5	80.1	4.8	0.0	83.3	0.0	1.7
MIVAE ($\lambda_{map} = 0$)	20.0	40.8	26.9	27.5	55.7	36.8	**36.7**	84.7	**51.2**	**38.3**	87.2	**53.2**	**16.1**	85.7	**27.1**	39.1
MIVAE ($\lambda_{map} = 1$)	**27.4**	41.0	**32.8**	**31.5**	60.4	**41.4**	35.9	87.8	51.0	32.9	90.4	48.2	15.3	87.4	26.0	**39.9**

  - f_μ: `DdimA`
  - f_{MLP}: `z-D1000BR`
- $q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{a})$ (Gaussian)
  - f_μ and f_{σ^2}: `D1000`
  - f_{MLP}: `(x,a)-D1000BR-D1000BR`
- $q(\boldsymbol{z}|\boldsymbol{x})$ (Gaussian)
  - f_μ and f_{σ^2}: `D1000`
  - f_{MLP}: `x-D1000BR-D1000BR`
- $q(\boldsymbol{z}|\boldsymbol{a})$ (Gaussian)
  - f_μ and f_{σ^2}: `D1000`
  - f_{MLP}: `a-D1000BR-D1000BR`

C DISCUSSION ON PROBABILISTIC MAPPING TO OBTAIN SHARED REPRESENTATION

In shared representation learning on ZSL, we consider nonlinear functions that map each modality as input into the same space $\mathcal{Z}$, i.e. $f_{\boldsymbol{x}} : \mathcal{X} \to \mathcal{Z}$ and $f_{\boldsymbol{a}} : \mathcal{A} \to \mathcal{Z}$.

If each mapping can be trained so that the same representation can be extracted from each modality of the same example, i.e. $\forall i \boldsymbol{z}_{xi} = \boldsymbol{z}_{ay_i}$, where $\boldsymbol{z}_{xi} = f_{\boldsymbol{x}}(\boldsymbol{x}_i)$ and $\boldsymbol{z}_{ay_i} = f_{\boldsymbol{a}}(\boldsymbol{a}_{y_i})$, we should be able to predict labels of the images in the test set by selecting the class that corresponds to the closest representation, i.e. $\hat{y}_j = \arg\min_{y \in \mathcal{Y}} dis(\boldsymbol{z}_{xj}, \boldsymbol{z}_{ay})$, where $dis(a, b)$ is an arbitrary distance function between a and b. Therefore, the simplest approach is to train the mappings so that the distance of the representations of the two modalities is small on the training set, i.e. $\min_{f_{\boldsymbol{x}}, f_{\boldsymbol{a}}} \sum_{i=1}^{N_{tr}} dis(f_{\boldsymbol{x}}(\boldsymbol{x}_i), f_{\boldsymbol{a}}(\boldsymbol{a}_{y_i}))$.

However, this method is highly likely to fail for the following two reasons. First, although only one example of attributes corresponding to a class exists, there are countless examples of images corresponding to that class. Therefore, if we attempt to make representations of attributes and images contained in one class closer, then these representations will degenerate to one point. Another reason is that if one can train to correspond well between each modality of training data, then that representation might not generalize to test data.

To solve these problems, we consider the uncertainty of each example of each modality. To account for this uncertainty, we replace the deterministic embedding functions, $f_{\boldsymbol{x}}$ and $f_{\boldsymbol{a}}$ with the deep probabilistic distributions, $p(\boldsymbol{z}_i|\boldsymbol{x}_i)$ and $p(\boldsymbol{z}_i|\boldsymbol{a}_{y_i})$.

The benefits of considering distributions for each example are twofold. First, we can prevent the degeneracy of representation which occurs when making examples of modalities that do not closely correspond one-to-one. This is true because we can consider the distance between distributions of each example, not between each example. Second, minimizing the distance between distributions of examples of each modality implies that the distribution of each modality is closer, i.e.,, $\forall i \in \{1..., N_{tr}\}[p(\boldsymbol{z}_i|\boldsymbol{x}_i) = p(\boldsymbol{z}_i|\boldsymbol{a}_{y_i})] \Rightarrow p(Z_x) = p(Z_a)$, where $Z_x = \{\boldsymbol{z}_{x1}..., \boldsymbol{z}_{xN_{tr}}\}$ and $Z_a =$

$\{\boldsymbol{z}_{ay_1}...,\boldsymbol{z}_{ay_{N_{tr}}}\}$. This point might be readily apparent from $\prod_{i=1}^{N_{tr}} p(\boldsymbol{z}_i|\boldsymbol{x}_i) = p(Z|X) = p(Z_x)$. Therefore, if we can train to bring each distribution closer, then we can obtain shared representation of them.

D COMPARISON WITH STATE-OF-THE-ART METHODS IN ALL METRICS

Table 5 shows that results evaluated with all metrics. Relation-based methods are known to tend to be biased to seen classes, so that the results of unseen classes become considerably worse (Chao et al., 2016). However, since our results are much greater than those, we can see that our approach solves the bias problem of GZSL. Furthermore, for seen classes, the performance of our method is equal to or higher than that of other methods. For harmonic mean, the proposed method is far superior to the others. Also, for seen classes, our method is equal to or higher than other methods, so that for harmonic mean, the proposed method is much superior to the others.