The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Zero-Shot Learning via Class-Conditioned Deep Generative Models

**Wenlin Wang,[1*] Yunchen Pu,[1] Vinay Kumar Verma,[3] Kai Fan,[2] Yizhe Zhang,[2]
Changyou Chen,[4] Piyush Rai,[3*] Lawrence Carin[1]**
[1]Department of Electrical and Computer Engineering, Duke University
[2]Compuational Biology and Bioinformatics, Duke University
[3]Department of Computer Science and Engineering, IIT Kanpur, India
[4]Department of Computer Science and Engineering, SUNY at Buffalo
{ww107, yp42, kf96, yz196, lcarin}@duke.edu, {vkverma, piyush}@cse.iitk.ac.in, cchangyou@gmail.com

Abstract

We present a deep generative model for Zero-Shot Learning (ZSL). Unlike most existing methods for this problem, that represent each class as a *point* (via a semantic embedding), we represent each seen/unseen class using a class-specific *latent-space distribution*, conditioned on class attributes. We use these latent-space distributions as a prior for a *supervised* variational autoencoder (VAE), which also facilitates learning highly discriminative feature representations for the inputs. The entire framework is learned end-to-end using only the seen-class training data. At test time, the label for an unseen-class test input is the class that maximizes the VAE lower bound. We further extend the model to a (*i*) semi-supervised/transductive setting by leveraging unlabeled unseen-class data via an *unsupervised* learning module, and (*ii*) few-shot learning where we also have a small number of labeled inputs from the unseen classes. We compare our model with several state-of-the-art methods through a comprehensive set of experiments on a variety of benchmark data sets.

Introduction

A goal of autonomous learning systems is the ability to learn new concepts even when the amount of supervision for such concepts is scarce or non-existent. This is a task that humans are able to perform effortlessly. Endowing machines with similar capability, however, has been challenging. Although machine learning and deep learning algorithms can learn reliable classification rules when supplied with abundant labeled training examples per class, their generalization ability remains poor for classes that are not well-represented (or not present) in the training data. This limitation has led to significant recent interest in zero-shot learning (ZSL) and one-shot/few-shot learning (Socher et al. 2013; Lampert et al. 2014; Lake et al. 2015; Vinyals et al. 2016; Ravi et al. 2017). We provide a more detailed overview of existing work on these methods in the Related Work section.

In order to generalize to previously unseen classes with no labeled training data, a common assumption is the availability of side information about the classes. The side information is usually provided in the form of class attributes (human-provided or learned from external sources such as Wikipedia) representing semantic information about the classes, or in the form of the similarities of the unseen classes with each of the seen classes. The side information can then be leveraged to design learning algorithms (Socher et al. 2013) that try to transfer knowledge from the seen classes to unseen classes (by linking corresponding attributes).

Although this approach has shown promise, it has several limitations. For example, most of the existing ZSL methods assume that each class is represented as a fixed point (e.g., an embedding) in some semantic space, which does not adequately account for intra-class variability (Akata et al. 2015). Another limitation of most existing methods is that they usually lack a proper generative model (Kingma et al. 2014b; Rezende et al. 2014; Kingma et al. 2014a) of the data. Having a generative model has several advantages (Kingma et al. 2014b; Rezende et al. 2014; Kingma et al. 2014a), such as unraveling the complex structure in the data by learning expressive feature representations and the ability to seamlessly integrate unlabeled data, leading to a transductive/semi-supervised estimation procedure. This, in the context of ZSL, may be especially useful when the amount of labeled data for the seen classes is small, but otherwise there may be plenty of unlabeled data from the seen/unseen classes.

Motivated by these desiderata, we design a deep generative model for the ZSL problem. Our model (summarized in Figure 1) learns a set of *attribute-specific* latent space distributions (modeled by Gaussians), whose parameters are outputs of a trainable deep neural network (defined by p_ψ in Figure 1). The attribute vector is denoted as $\boldsymbol{a}$, and is assumed given for each training image, and it is inferred for test images. The class label is linked to the attributes, and therefore by inferring attributes of a test image, there is an opportunity to recognize classes at test time that were not seen when training. These latent-space distributions serve as a prior for a variational autoencoder (VAE) (Kingma et al. 2014b) model (defined by a decoder p_θ and an encoder q_ϕ in Figure 1). This combination further helps the VAE to learn discriminative feature representations for the inputs. Moreover, the generative aspect also facilitates extending our model to semi-supervised/transductive settings (omitted in Figure 1 for brevity, but discussed in detail in the Transductive ZSL section) using a deep *unsupervised* learning module. All the parameters defining the model, including the deep neural-network parameters ψ and the VAE decoder

[*] Corresponding authors

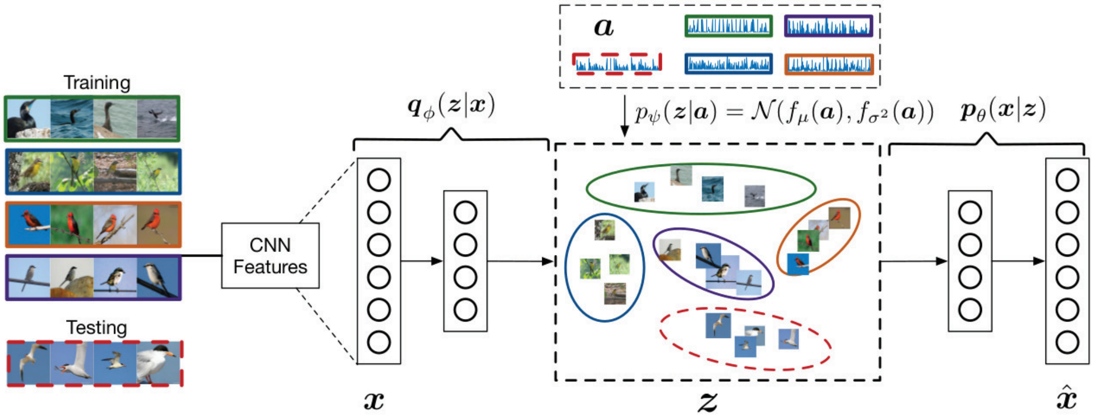

Figure 1: A diagram of our basic model; only the training stage is shown here. In the above figure, $\boldsymbol{a} \in \mathbb{R}^M$ denotes the class attribute vector (given for training data, inferred for test data). Red-dotted rectangle/ellipse correspond to the unseen classes. Note: The CNN module is not part of our framework and is only used as an initial feature extractor, on top of which the rest of our model is built. The CNN can be replaced by any feature extractor depending on the data type

and encoder parameters θ, ϕ, are learned end-to-end, using only the seen-class labeled data (and, optionally, the available unlabeled data when using the semi-supervised/transductive setting).

Once the model has been trained, it can be used in the ZSL setting as follows. Assume that there are classes we wish to identify at test time that have not been seen when training. While we have not seen images before from such classes, it is assumed that we know the attributes of these previously unseen classes. The latent space distributions $p_\psi(\boldsymbol{z}|\boldsymbol{a})$ for all the unseen classes (Figure 1, best seen in color, shows this distribution for one such unseen class using a red-dotted ellipse) are inferred by conditioning on the respective class attribute vectors $\boldsymbol{a}$ (including attribute vectors for classes not seen when training). Given a test input $\boldsymbol{x}_*$ from some unseen class, the associated class attributes $\boldsymbol{a}_*$ are predicted by first mapping $\boldsymbol{x}_*$ to the latent space via the VAE recognition model $q_\phi(\boldsymbol{z}_*|\boldsymbol{x}_*)$, and then finding $\boldsymbol{a}_*$ that maximizes the VAE lower bound. The test image is assigned a class label y_* linked with $\boldsymbol{a}_*$. This is equivalent to finding the class latent distribution p_ψ that has the smallest KL divergence w.r.t. the variational distribution $q_\phi(\boldsymbol{z}_*|\boldsymbol{x}_*)$.

Variational Autoencoder

The variational autoencoder (VAE) is a deep generative model (Kingma et al. 2014b; Rezende et al. 2014), capable of learning complex density models for data via latent variables. Given a nonlinear generative model $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ with input $\boldsymbol{x} \in \mathbb{R}^D$ and associated latent variable $\boldsymbol{z} \in \mathbb{R}^L$ drawn from a prior distribution $p_0(\boldsymbol{z})$, the goal of the VAE is to use a recognition model $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ (also called an inference network) to approximate the posterior distribution of the latent variables, i.e., $p_\theta(\boldsymbol{z}|\boldsymbol{x})$, by maximizing the following variational lower bound

$$\mathcal{L}^{\text{V}}_{\theta,\phi}(\boldsymbol{x}) = \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] - \text{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_0(\boldsymbol{z})) \ .$$

Typically, $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ is defined as an isotropic normal distribution with its mean and standard deviation the output of a deep neural network, which takes $\boldsymbol{x}$ as input. After learning the VAE, a probabilistic "encoding" $\boldsymbol{z}$ for the input $\boldsymbol{x}$ can be generated efficiently from the recognition model $q_\phi(\boldsymbol{z}|\boldsymbol{x})$.

We leverage the flexibility of the VAE to design a *structured*, supervised VAE that allows us to incorporate class-specific information (given in the form of class attribute vectors $\boldsymbol{a}$). This enables one to learn a deep generative model that can be used to predict the labels for examples from classes that were not seen at training time (by linking inferred attributes to associated labels, even labels not seen when training).

Deep Generative Model for ZSL

We consider two settings for ZSL learning: inductive and transductive. In the standard inductive setting, during training, we only assume access to labeled data from the seen classes. In the transductive setting (Kodirov et al. 2015), we also assume access to the *unlabeled* test inputs from the unseen classes. In what follows, under the *Inductive ZSL* section, we first describe our deep generative model for the inductive setting. Then, in the *Transductive ZSL* section, we extend this model for the transductive setting, in which we incorporate an *unsupervised* deep embedding module to help leverage the *unlabeled* inputs from the unseen classes. Both of our models are built on top of a variational autoencoder (Kingma et al. 2014b; Rezende et al. 2014). However, unlike the standard VAE (Kingma et al. 2014b; Rezende et al. 2014), our framework leverages attribute-specific latent space distributions which act as the prior (Figure 1) on the latent codes of the inputs. This enables us to adapt the VAE framework for the problem of ZSL.

Notation In the ZSL setting, we assume there are S seen classes and U unseen classes. For each seen/unseen class, we are given side information, in the form of M-dimensional class-attribute vectors (Socher et al. 2013). The side information is leveraged for ZSL. We collectively denote the attribute vectors of all the classes using a matrix $\mathbf{A} \in \mathbb{R}^{M\times(S+U)}$. During training, images are available only for the seen classes, and the labeled data are denoted $\mathcal{D}_s = \{(\boldsymbol{x}_n, \boldsymbol{a}_n)\}_{n=1}^N$, where $\boldsymbol{x}_n \in \mathbb{R}^D$ and $\boldsymbol{a}_n = \mathbf{A}_{y_n}$, $\mathbf{A}_{y_n} \in \mathbb{R}^M$ denotes the y_n^{th} column of $\mathbf{A}$ and $y_n \in \{1,\dots,S\}$ is the corresponding label for $\boldsymbol{x}_n$. The remaining classes, indexed as $\{S+1,\dots,S+U\}$, represent the unseen classes (while we know the U associated attribute vectors, at training we have no corresponding images available). Note that each class has a unique associated attribute vector, and we infer unseen classes/labels by inferring the attributes at test, and linking them to a label.

Inductive ZSL

We model the data $\{\boldsymbol{x}_n\}_{n=1}^N$ using a VAE-based deep generative model, defined by a decoder $p_\theta(\boldsymbol{x}_n|\boldsymbol{z}_n)$ and an encoder $q_\phi(\boldsymbol{z}_n|\boldsymbol{x}_n)$. As in the standard VAE, the decoder $p_\theta(\boldsymbol{x}_n|\boldsymbol{z}_n)$ represents the generative model for the inputs $\boldsymbol{x}_n$, and θ represents the parameters of the deep neural network that define the decoder. Likewise, the encoder $q_\phi(\boldsymbol{z}_n|\boldsymbol{x}_n)$ is the VAE *recognition model*, and ϕ represents the parameters of the deep neural network that define the encoder.

However, in contrast to the standard VAE prior that assumes each latent embedding $\boldsymbol{z}_n$ to be drawn from the same latent Gaussian (e.g., $p_\psi(\boldsymbol{z}_n) = \mathcal{N}(0,\mathbf{I})$), we assume each $\boldsymbol{z}_n$ to be drawn from a *attribute-specific* latent Gaussian, $p_\psi(\boldsymbol{z}_n|\boldsymbol{a}_n) = \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{a}_n), \boldsymbol{\Sigma}(\boldsymbol{a}_n))$, where

$$\boldsymbol{\mu}(\boldsymbol{a}_n) = f_\mu(\boldsymbol{a}_n),\;\; \boldsymbol{\Sigma}(\boldsymbol{a}_n) = \mathrm{diag}(\exp(f_\sigma(\boldsymbol{a}_n))) \qquad (1)$$

where we assume $f_\mu(\cdot)$ and $f_\sigma(\cdot)$ to be linear functions, *i.e.*, $f_\mu(\boldsymbol{a}_n) = \mathbf{W}_\mu \boldsymbol{a}_n$ and $f_\sigma(\boldsymbol{a}_n) = \mathbf{W}_\sigma \boldsymbol{a}_n$; $\mathbf{W}_\mu$ and $\mathbf{W}_\sigma$ are learned parameters. One may also consider $f_\mu(\cdot)$ and $f_\sigma(\cdot)$ to be a deep neural network; this added complexity was not found necessary for the experiments considered. Note that once $\mathbf{W}_\mu$ and $\mathbf{W}_\sigma$ are learned, the parameters $\{\boldsymbol{\mu}(\boldsymbol{a}), \boldsymbol{\Sigma}(\boldsymbol{a})\}$ of the latent Gaussians of unseen classes $c = S+1,\dots,S+U$ can be obtained by plugging in their associated class attribute vectors $\{\mathbf{A}_c\}_{c=S+1}^{S+U}$, and inferring which provides a better fit to the data.

Given the class-specific priors $p_\psi(\boldsymbol{z}_n|\boldsymbol{a}_n)$ on the latent code $\boldsymbol{z}_n$ of each input, we can define the following variational lower bound for our VAE based model (we omit the subscript n for simplicity)

$$\mathcal{L}_{\theta,\phi,\psi}(\boldsymbol{x},\boldsymbol{a}) = \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] - \mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\psi(\boldsymbol{z}|\boldsymbol{a})) \qquad (2)$$

Margin Regularizer The objective in (2) naturally encourages the inferred variational distribution $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ to be close to the class-specific latent space distribution $p_\psi(\boldsymbol{z}|\boldsymbol{a})$. However, since our goal is classification, we augment this objective with a *maximum-margin* criterion that promotes $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ to be as far away as possible from all other class-specific latent space distributions $p_\psi(\boldsymbol{z}|\mathbf{A}_c)$, $\mathbf{A}_c \neq \boldsymbol{a}$. To this end, we replace the $-\mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\psi(\boldsymbol{z}|\boldsymbol{a}))$ term in our original VAE objective (2) by $-[\mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\psi(\boldsymbol{z}|\boldsymbol{a})) - R^*]$ where "margin regularizer" term R^* is defined as the minimum of the KL divergence between $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ and all other class-specific latent space distributions:

$$\begin{aligned} R^* &= \min_{c:c\in\{1..,y-1,y+1,..,S\}} \{\mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\psi(\boldsymbol{z}|\mathbf{A}_c))\} \\ &= -\max_{c:c\in\{1..,y-1,y+1,..,S\}} \{-\mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\psi(\boldsymbol{z}|\mathbf{A}_c))\} \end{aligned} \qquad (3)$$

Intuitively, the regularizer $-[\mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\psi(\boldsymbol{z}|\boldsymbol{a})) - R^*]$ encourages the true class and the *next best* class to be separated maximally. However, since R^* is non-differentiable, making the objective difficult to optimize in practice, we approximate R^* by the following surrogate:

$$R = -\log \sum_{c=1}^{S} \exp(-\mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\psi(\boldsymbol{z}|\mathbf{A}_c))) \qquad (4)$$

It can be easily shown that

$$R^* \leq R \leq R^* + \log S \qquad (5)$$

Therefore when we maximize R, it is equivalent to maximizing a lower bound on R^*. Finally, we optimize the variational lower bound together with the margin regularizer as

$$\begin{aligned} \hat{\mathcal{L}}_{\theta,\phi,\psi}(\boldsymbol{x},\boldsymbol{a}) &= \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] - \mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\psi(\boldsymbol{z}|\boldsymbol{a})) \\ & \underbrace{-\lambda \log \sum_{c=1}^{S} \exp(-\mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\psi(\boldsymbol{z}|\mathbf{A}_c)))}_{R} \end{aligned} \qquad (6)$$

where λ is a hyper-parameter controlling the extent of regularization. We train the model using the *seen-class* labeled examples $\mathcal{D}_s = \{(\boldsymbol{x}_n, \boldsymbol{a}_n)\}_{n=1}^N$ and learn the parameters (θ, ϕ, ψ) by maximizing the objective in (6). Once the model parameters have been learned, the label for a new input $\hat{\boldsymbol{x}}$ from an *unseen* class can be predicted by first predicting its latent embedding $\hat{\boldsymbol{z}}$ using the VAE recognition model, and then finding the "best" label by solving

$$\begin{aligned} \hat{y} &= \arg\max_{y\in\mathcal{Y}_u} \mathcal{L}_{\theta,\phi,\psi}(\hat{\boldsymbol{x}}, \mathbf{A}_y) \\ &= \arg\min_{y\in\mathcal{Y}_u} \mathrm{KL}(q_\phi(\hat{\boldsymbol{z}}|\hat{\boldsymbol{x}})||p_\psi(\hat{\boldsymbol{z}}|\mathbf{A}_y)) \end{aligned} \qquad (7)$$

where $\mathcal{Y}_u = \{S+1,\dots,S+U\}$ denotes the set of unseen classes. Intuitively, the prediction rule assigns $\hat{\boldsymbol{x}}$ to that unseen class whose class-specific latent space distribution $p_\psi(\hat{\boldsymbol{z}}|\boldsymbol{a})$ is most similar to the VAE posterior distribution $q_\phi(\hat{\boldsymbol{z}}|\hat{\boldsymbol{x}})$ of the latent embeddings. Unlike the prediction rule of most ZSL algorithms that are based on simple Euclidean distance calculations of a point embedding to a set of "class prototypes" (Socher et al. 2013), our prediction rule naturally takes into account the possible *multi-modal* nature of the class distributions and therefore is expected to result in better prediction, especially when there is a considerable amount of intra-class variability in the data.

Transductive ZSL

We now present an extension of the model for the *transductive* ZSL setting (Kodirov et al. 2015), which assumes that the test inputs $\{\hat{\boldsymbol{x}}_i\}_{i=1}^{N'}$ from the unseen classes are also available while training the model. Note that, for the inductive ZSL setting (using the objective in (6), the KL term between an unseen class test input $\hat{\boldsymbol{x}}_i$ and its class based prior is given by $-\mathrm{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i)||p_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{a})))$. If we had access to the true labels of these inputs, we could add those directly to the original optimization problem ((6)). However, since we do not know these labels, we propose an unsupervised method that can still use these unlabeled inputs to *refine* the inductive model presented in the previous section.

A naïve approach for directly leveraging the unlabeled inputs in (6) without their labels would be to add the following reconstruction error term to the objective

$$\tilde{\mathcal{L}}_{\theta,\phi,\psi}(\hat{\boldsymbol{x}},\boldsymbol{a}) = \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\hat{\boldsymbol{x}}|\boldsymbol{z})] \tag{8}$$

However, since this objective completely ignores the label information of $\hat{\boldsymbol{x}}$, it is not expected to work well in practice and only leads to marginal improvements over the purely inductive case (as corroborated in our experiments).

To better leverage the unseen class test inputs in the transductive setting, we augment the inductive ZSL objective (6) with an additional unlabeled data based regularizer that uses only the unseen class test inputs.

This regularizer is motivated by the fact that the inductive model is able to make reasonably confident predictions (as measured by the predicted class distributions for these inputs) for unseen class test inputs, and these confident predicted class distributions can be emphasized in this regularizer to guide those ambiguous test inputs. To elaborate the regularizer, we first define the inductive model's predicted *probability* of assigning an unseen class test input $\hat{\boldsymbol{x}}_i$ to class $c \in \{S+1,\ldots,S+U\}$ to be

$$q(\hat{\boldsymbol{x}}_i, c) = \frac{\exp(-\mathrm{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i)||p_{\boldsymbol{\psi}}(\boldsymbol{z}|\mathbf{A}_c)))}{\sum_c \exp(-\mathrm{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i)||p_{\boldsymbol{\psi}}(\boldsymbol{z}|\mathbf{A}_c)))} \tag{9}$$

Our proposed regularizer (defined below in (10)) promotes these class probability estimates $q(\hat{\boldsymbol{x}}_i, c)$ to be sharper, i.e., the most likely class should dominate the predicted class distribution $q(\hat{\boldsymbol{x}}_i, c)$) for the unseen class test input $\hat{\boldsymbol{x}}_i$.

Specifically, we define a sharper version of the predicted class probabilities $q(\hat{\boldsymbol{x}}_i, c)$ as $p(\hat{\boldsymbol{x}}_i, c) = \frac{q(\hat{\boldsymbol{x}}_i,c)^2/g(c)}{\sum_{c'} q(\hat{\boldsymbol{x}}_i,c')^2/g(c')}$, where $g(c) = \sum_{i=1}^{N'} q(\hat{\boldsymbol{x}}_i, c)$ is the marginal probability of unseen class c. Note that normalizing the probabilities by $g(c)$ prevents large classes from distorting the latent space.

We then introduce our KL based regularizer that encourages $q(\hat{\boldsymbol{x}}_i, c)$ to be close to $p(\hat{\boldsymbol{x}}_i, c)$. This can be formalized by defining the sum of the KL divergences between $q(\hat{\boldsymbol{x}}_i, c)$ and $p(\hat{\boldsymbol{x}}_i, c)$ for all the unseen class test inputs, i.e,

$$\mathrm{KL}(P(\hat{\mathbf{X}})||Q(\hat{\mathbf{X}})) \triangleq \sum_{i=1}^{N'} \sum_{c=S+1}^{S+U} p(\hat{\boldsymbol{x}}_i, c) \log \frac{p(\hat{\boldsymbol{x}}_i, c)}{q(\hat{\boldsymbol{x}}_i, c)} \tag{10}$$

A similar approach of *sharpening* was recently utilized in the context of learning deep embeddings for clustering problems (Xie et al. 2016) and data summarization (Wang et al. 2016b), and is reminiscent of self-training algorithms used in semi-supervised learning (Nigam et al. 2000).

Intuitively, unseen class test inputs with *sharp* probability estimates will have a more significant impact on the gradient norm of (10), which in turn leads to improved predictions on the ambiguous test examples (our experimental results corroborate this). Combining (8) and (10), we have the following objective (which we seek to *maximize*) defined exclusively over the unseen class unlabeled inputs

$$U(\hat{\mathbf{X}}) = \sum_{i=1}^{N'} \mathbb{E}_{q_\phi(\boldsymbol{z}|\hat{\boldsymbol{x}}_i)}[\log p_{\boldsymbol{\theta}}(\hat{\boldsymbol{x}}_i|\boldsymbol{z})] - \mathrm{KL}(P(\hat{\mathbf{X}})||Q(\hat{\mathbf{X}})) \tag{11}$$

We finally combine this objective with the original objective ((6)) for the inductive setting, which leads to the overall objective $\sum_{n=1}^{N} \hat{\mathcal{L}}_{\theta,\phi,\psi}(\boldsymbol{x}_n, \boldsymbol{a}_n) + U(\hat{\mathbf{X}})$, defined over the seen class labeled training inputs $\{(\boldsymbol{x}_n, \boldsymbol{a}_n)\}_{n=1}^{N}$ and the unseen class unlabeled test inputs $\{\hat{\boldsymbol{x}}_i\}_{i=1}^{N'}$.

Under our proposed framework, it is also straightforward to perform few-shot learning (Lake et al. 2015; Vinyals et al. 2016; Ravi et al. 2017) which refers to the setting when a small number of labeled inputs may also be available for classes $c = S+1,\ldots,S+U$. For these inputs, we can directly optimize (6) on classes $c = S+1,\ldots,S+U$.

Related Work

Several prior methods for zero-shot learning (ZSL) are based on embedding the inputs into a semantic vector space, where nearest-neighbor methods can be applied to find the most likely class, which is represented as a point in the same semantic space (Socher et al. 2013; Norouzi et al. 2013). Such approaches can largely be categorized into three types: (*i*) methods that learn the projection from the input space to the semantic space using either a linear regression or a ranking model (Akata et al. 2015; Lampert et al. 2014), or using a deep neural network(Socher et al. 2013); (*ii*) methods that perform a "reverse" projection from the semantic space to the input space(Zhang et al. 2016a), which helps to reduce the *hubness problem* encountered when doing nearest neighbor search at test time (Radovanović et al. 2010); and (*iii*) methods that learn a shared embedding space for the inputs and the class attributes (Zhang et al. 2016b; Changpinyo et al. 2016).

Another popular approach to ZSL is based on modeling each unseen class as a linear/convex combination of seen classes (Norouzi et al. 2013), or of a set of shared "abstract" or "basis" classes (Romera-Paredes et al. 2015; Changpinyo et al. 2016). Our framework can be seen as a flexible generalization to the latter type of models since the parameters $\mathbf{W}_\mu$ and $\mathbf{W}_\sigma$ defining the latent space distributions are shared by the seen and unseen classes.

One general issue in ZSL is the *domain shift* problem – when the seen and unseen classes come from very different domains. Standard ZSL models perform poorly under these situations. However, utilizing some additional unlabeled data from those unseen domains can somewhat alleviates the problem. To this end, (Kodirov et al. 2015) presented a transductive ZSL model which uses a dictionary-learning-based approach for learning unseen-class classifiers. In their approach, the dictionary is adapted to the unseen-class domain using the unlabeled test inputs from unseen classes. Other methods that can leverage unlabeled data include (Fu et al. 2015a; Rohrbach et al. 2013; Li et al. 2015; Zhao et al. 2016). Our model is robust to the *domain shift* problem due to its ability to incorporate unlabeled data from unseen classes.

Somewhat similar to our VAE based approach, recently (Kodirov et al. 2017) proposed a semantic autoencoder for ZSL. However, their method does not have a proper generative model. Moreover, it assumes each class to be represented as a fixed point and cannot extend to the transductive setting.

Deep encoder-decoder based models have recently gained much attention for a variety of problems, ranging from image generation (Rezende et al. 2016) and text matching (Shen et al. 2017). A few recent works exploited the idea of applying sematic regularization to the latent embedding spaced shared between encoder and decoder to make it suitable for ZSL tasks (Kodirov et al. 2017; Tsai et al. 2017). However, these methods lack a proper generative model; moreover (i) these methods assume each class to be represented as a fixed point, and (ii) these methods cannot extend to the transductive setting. Variational autoencoder (VAE) (Kingma et al. 2014b) offers an elegant probabilistic framework to generate continues samples from a latent gaussian distribution and its supervised extensions (Kingma et al. 2014a) can be used in supervised and semi-supervised tasks. However, supervised/semi-supervised VAE (Kingma et al. 2014a) assumes all classes to be seen at the training time and the label space $p(y)$ to be discrete, which makes it unsuitable for the ZSL setting. In contrast to these methods, our approach is based on a deep generative framework using a supervised variant of VAE, treating each class as a distribution in a latent space. This naturally allows us to handle the intra-class variability. Moreover, the supervised VAE model helps learning highly discriminative representations of the inputs.

Some other recent works have explored the idea of generative models for zero-shot learning (Li et al. 2017; Verma et al. 2017). However, these are primarily based on linear generative models, unlike our model which can learn discriminative and highly nonlinear embeddings of the inputs. In our experiments, we have found this to lead to significant improvements over linear models (Li et al. 2017; Verma et al. 2017).

Deep generative models have also been proposed recently for tasks involving learning from limited supervision, such as one-shot learning (Rezende et al. 2016). These models are primarily based on feedback and attention mechanisms. However, while the goal of our work is to develop methods to help recognize previously unseen classes, the focus of methods such as (Rezende et al. 2016) is on tasks such as generation, or learning from a very small number of labeled examples. It will be interesting to combine the expressiveness of such models within the context of ZSL.

Experiments

We evaluate our framework for ZSL on several benchmark datasets and compare it with a number of state-of-the-art baselines. Specifically, we conduct our experiments on the following datasets: (i) Animal with Attributes (AwA) (Lampert et al. 2014); (ii) Caltech-UCSD Birds-200-2011 (CUB-200) (Wah et al. 2011); and (iii) SUN attribute (SUN) (Patterson et al. 2012). For the large-scale dataset (ImageNet), we follow (Fu et al. 2016), for which 1000 classes from ILSVRC2012 (Russakovsky et al. 2015) are used as seen classes, while 360 non-overlapped classes of ILSVRC2010 (Deng et al. 2009) are used as unseen classes. The statistics of these datasets are listed in Table 1.

Dataset	# Attribute	training(+validation)		testing	
		# of images	# of classes	# of images	# of classes
AwA	85	24,295	40	6,180	10
CUB	312	8,855	150	2,933	50
SUN	102	14,140	707	200	10
ImageNet	1,000	200,000	1,000	54,000	360

Table 1: Summary of datasets used in the evaluation

In all our experiments, we consider VGG-19 fc7 features (Simonyan et al. 2014) as our raw input representation, which is a 4096-dimensional feature vector. For the semantic space, we adopt the default class attribute features provided for each of these datasets. The only exception is ImageNet, for which the semantic word vector representation is obtained from word2vec embeddings (Mikolov et al. 2013) trained on a skip-gram text model on 4.6 million Wikipedia documents. For the reported experiments, we use the standard train/test split for each dataset, as done in the prior work. For hyper-parameter selection, we divide the training set into training and validation set; the validation set is used for hyper-parameter tuning, while setting $\lambda = 1$ across all our experiments.

For the VAE model, a multi-layer perceptron (MLP) is used for both encoder $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ and decoder $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$. The encoder and decoder are defined by an MLP with two hidden layers, with 1000 nodes in each layer. ReLU is used as the nonlinear activation function on each hidden layer and dropout with constant rate 0.8 is used to avoid overfitting. The latent space $\boldsymbol{z}$ was set to be 100 for small datasets and 500 for ImageNet. Our results with variance are reported by repeating with 10 runs. Our model is written in Tensorflow and trained on NVIDIA GTX TITAN X with 3072 cores and 11GB global memory.

We compare our method (referred to as VZSL) with a variety of state-of-the-art baselines using VGG-19 fc7 features and specifically we conduct our experiments on the following tasks:

- **Inductive ZSL:** This is the standard ZSL setting where the unseen class latent space distributions are learned using only seen class data.
- **Transductive ZSL:** In this setting, we also use the unlabeled test data while learning the unseen class latent space distributions. Note that, while this setting has access to more information about the unseen class, it is only through unlabeled data.
- **Few-Shot Learning:** In this setting (Lake et al. 2015; Vinyals et al. 2016; Ravi et al. 2017), we also use a small number of labeled examples from each unseen class.

In addition, through a visualization experiment (using t-SNE (Maaten et al. 2008)), we also illustrate our model's

Method	AwA	CUB-200	SUN	Average	Method	ImageNet
(Lampert et al. 2014)	57.23	–	72.00	–	DeViSE (Frome et al. 2013)	12.8
ESZSL (Romera-Paredes et al. 2015)	75.32 ± 2.28	–	82.10 ± 0.32	–	ConSE (Norouzi et al. 2013)	15.5
MLZSC (Bucher et al. 2016)	77.32 ± 1.03	43.29 ± 0.38	84.41 ± 0.71	68.34	AMP (Fu et al. 2015b)	13.1
SDL (Zhang et al. 2016b)	80.46 ± 0.53	42.11 ± 0.55	83.83 ± 0.29	68.80	SS-Voc (Fu et al. 2016)	16.8
BiDiLEL (Wang et al. 2016a)	79.20	46.70	–	–		
SSE-ReLU (Zhang et al. 2015)	76.33 ± 0.83	30.41 ± 0.20	82.50 ± 1.32	63.08		
JFA (Zhang et al. 2016a)	81.03 ± 0.88	46.48 ± 1.67	84.10 ± 1.51	70.53		
SAE (Kodirov et al. 2017)	83.40	56.60	84.50	74.83		
GFZSL (Verma et al. 2017)	80.83	56.53	86.50	74.59		
VZSL$^{\#}$	84.45 ± 0.74	55.37 ± 0.59	85.75 ± 1.93	74.52	-	22.88
VZSL	$\mathbf{85.28 \pm 0.76}$	$\mathbf{57.42 \pm 0.63}$	$\mathbf{86.75 \pm 2.02}$	**76.48**	-	**23.08**

Table 2: Top-1 classification accuracy (%) on AwA, CUB-200, SUN and Top-5 accuracy(%) on ImageNet under inductive ZSL. VZSL$^{\#}$ denotes our model trained with the reconstruction term from (6) ignored.

behavior in terms its ability to separate the different classes in the latent space.

Inductive ZSL

Table 2 shows our results for the inductive ZSL setting. The results of the various baselines are taken from the corresponding papers or reproduced using the publicly available implementations. From Table 2, we can see that: (i) our model performs better than all the baselines, by a reasonable margin on the small-scale datasets; (ii) On large-scale datasets, the margin of improvement is even more significant and we outperform the best-performing state-of-the art baseline by a margin of 37.4%; (iii) Our model is superior when including the reconstruction term, which shows the effectiveness of the generative model; (iv) Even without the reconstruction term, our model is comparable with most of the other baselines. The effectiveness of our model can be attributed to the following aspects. First, as compared to the methods that embed the test inputs in the semantic space and then find the most similar class by doing a Euclidean distance based nearest neighbor search, or methods that are based on constructing unseen class classified using a weighted combination of seen class classifiers (Zhang et al. 2015), our model finds the "most probable class" by computing the distance of each test input from *class distributions*. This naturally takes into account the shape (possibly multi-modal) and spread of the class distribution. Second, the reconstruction term in the VAE formulation further strengthens the model. It helps leverage the intrinsic structure of the inputs while projecting them to the latent space. This aspect has been shown to also help other methods such as (Kodirov et al. 2017) (which we use as one of the baseline), but the approach in (Kodirov et al. 2017) lacks a generative model. This explains the favorable performance of our model as compared to such methods.

Method	AwA	CUB-200	SUN	Average
SMS (Guo et al. 2016)	78.47	–	82.00	–
ESZSL (Romera-Paredes et al. 2015)	84.30	–	37.50	–
JFA+SP-ZSR (Zhang et al. 2016a)	88.04 ± 0.69	55.81 ± 1.37	85.35 ± 1.56	77.85
SDL (Zhang et al. 2016b)	92.08 ± 0.14	55.34 ± 0.77	86.12 ± 0.99	76.40
DMaP (Li et al. 2017)	85.66	61.79	–	–
TASTE (Yu et al. 2017a)	89.74	54.25	–	–
TSTD (Yu et al. 2017b)	90.30	58.20	–	–
GFZSL (Verma et al. 2017)	94.25	63.66	87.00	80.63
VZSL$^{\#}$	93.49 ± 0.54	59.69 ± 1.22	86.37 ± 1.88	79.85
VZSL*	87.59 ± 0.21	61.44 ± 0.98	86.66 ± 1.67	77.56
VZSL	$\mathbf{94.80 \pm 0.17}$	$\mathbf{66.45 \pm 0.88}$	$\mathbf{87.75 \pm 1.43}$	**83.00**

Table 3: Top-1 classification accuracy (%) obtained on AwA, CUB-200 and SUN under transductive setting. VZSL$^{\#}$ denotes our model with VAE reconstruction term ignored. VZSL* denotes our model with only Eq (8) for unlabeled data. The '-' indicates the results was not reported

Transductive ZSL

Our next set of experiments consider the transductive setting. Table 3 reports our results for the transductive setting, where we compare with various state-of-the-art baselines that are designed to work in the transductive setting. As Table 3 shows, our model again outperforms the other state-of-the-art methods by a significant margin. We observe that the generative framework is able to effectively leverage unlabeled data and significantly improve upon the results of inductive setting. On average, we obtain about 8% better accuracies as compared to the inductive setting. Also note that in some cases, such as CUB-200, the classification accuracies drop significantly once we remove the VAE reconstruction term. A possible explanation to this behavior is that the CUB-200 is a relative difficult dataset with many classes are very similar to each other, and the inductive setting may not achieve very confident predictions on the unseen class examples during the inductive pre-training process. However, adding the reconstruction term back into the model significantly improves the accuracies. Further, compare our entire model with the one having only (8) for the unlabeled, there is a margin for about 5% on AwA and CUB-200, which indicates the necessity of introduced KL term on unlabeled data.

Few-Shot Learning (FSL)

In this section, we report results on the task of FSL (Salakhutdinov et al. 2013; Mensink et al. 2014) and transductive FSL (Frome et al. 2013) (Socher et al. 2013). In contrast to standard ZSL, FSL allows leveraging a few labeled inputs from the unseen classes, while the transductive FSL additionally also allows leveraging unseen class unlabeled test inputs. To see the effect of knowledge transfer from the seen classes, we use a multiclass SVM as a baseline that is provided the same number of labeled examples from each unseen class. In this setting, we vary the number of labeled examples from 2 to 20 (for SUN, we only use 2, 5 and 10 due to the small number of labeled examples). In Figure 3, we also compared with standard inductive ZSL which does not have access to the labeled examples from the unseen classes. Our results are shown in Figure 3.

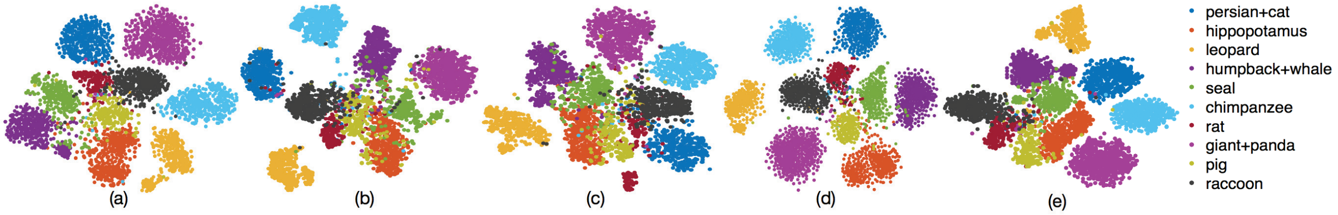

Figure 2: t-SNE visualization for AwA dataset (a) Original CNN features (b) Latent code for our VZSL under inductive zero-shot setting (c) Reconstructed features under inductive zero-shot setting (d) Latent code for our VZSL under transductive zero-shot setting (e) Reconstructed features under transductive setting. Different colors indicate different classes.

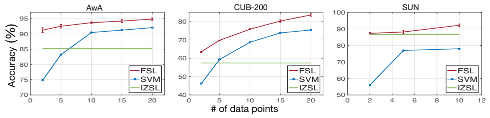

Figure 3: Accuracies (%) in FSL setting: For each data set, results are reported using 2,5,10,15,20 labeled examples for each unseen class

As can be seen, even with as few as 2 or 5 additional labeled examples per class, the FSL significantly improves over ZSL. We also observe that the FSL outperform a multiclass SVM which demonstrates the advantage of the knowledge transfer from the seen class data. Table 4 reports our results for the transductive FSL setting where we compare with other state-of-the-art baselines. In this setting too, our approach outperforms the baselines.

Table 4: Transductive few-shot recognition comparison using top-1 classification accuracy (%). For each test class, 3 images are randomly labeled, while the rest are unlabeled

Method	AwA	CUB-200	Average
DeViSE (Frome et al. 2013)	92.60	57.50	75.05
CMT (Socher et al. 2013)	90.60	62.50	76.55
ReViSE (Tsai et al. 2017)	94.20	68.40	81.30
VZSL	$\mathbf{95.62 \pm 0.24}$	$\mathbf{68.85 \pm 0.69}$	**82.24**

t-SNE Visualization

To show the model's ability to learn highly discriminative representations in the latent embedding space, we perform a visualization experiment. Figure 2 shows the t-SNE (Maaten et al. 2008) visualization for the raw inputs, the learn latent embeddings, and the *reconstructed* inputs on AwA dataset, for both inductive ZSL and transductive ZSL setting.

As can be seen, both the reconstructions and the latent embeddings lead to reasonably separated classes, which indicates that our generative model is able to learn a highly discriminative latent representations. We also observe that the inherent correlation between classes might change after we learn the latent embeddings of the inputs. For example, "giant+panda" is close to "persian+cat" in the original CNN features space but far away from each other in our learned latent space under transductive setting. A possible explanation could be that the sematic features and image features express information from different views and our model learns a representation that is sort of a compromise of these two representations.

Conclusion

We have presented a deep generative framework for learning to predict unseen classes, focusing on inductive and transductive zero-shot learning (ZSL). In contrast to most of the existing methods for ZSL, our framework models each seen/unseen class using a class-specific latent-space distribution and also models each input using a VAE-based decoder model. Prediction for the label of a test input from any unseen class is done by matching the VAE posterior distribution for the latent representation of this input with the latent-space distributions of each of the unseen class. This distribution matching method in the latent space provides more robustness as compared to other existing ZSL methods that simply use a point-based Euclidean distance metric. Our VAE based framework leverages the intrinsic structure of the input space through the generative model. Moreover, we naturally extend our model to the transductive setting by introducing an additional regularizer for the unlabeled inputs from unseen classes. We demonstrate through extensive experiments that our generative framework yields superior classification accuracies as compared to existing ZSL methods, on both inductive ZSL as well as transductive ZSL tasks. Finally, although we use isotropic Gaussian to model each model each seen/unseen class, it is possible to model with more general Gaussian or any other distribution depending on the data type. We leave this possibility as a direction for future work.

Acknowledgements: This research was supported in part by grants from DARPA, DOE, NSF and ONR. PR and VKV also acknowledge support from Tower Research CSR, Dr.Deep Singh and Daljeet Kaur Fellowship, and Visvesvaraya Ph.D.

References

Akata, Z.; Reed, S.; Walter, D.; Lee, H.; and Schiele, B. 2015. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, 2927–2936.

Bucher, M.; Herbin, S.; and Jurie, F. 2016. Improving semantic embedding consistency by metric learning for zero-shot classiffication. In *ECCV*, 730–746. Springer.

Changpinyo, S.; Chao, W.-L.; Gong, B.; and Sha, F. 2016. Synthesized classifiers for zero-shot learning. In *CVPR*, 5327–5336.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*, 248–255. IEEE.

Frome, A.; Corrado, G. S.; Shlens, J.; Bengio, S.; Dean, J.; Mikolov, T.; et al. 2013. Devise: A deep visual-semantic embedding model. In *NIPS*, 2121–2129.

Fu, Y.; Hospedales, T. M.; Xiang, T.; and Gong, S. 2015a. Transductive multi-view zero-shot learning. *TPAMI* 37(11):2332–2345.

Fu, Z.; Xiang, T.; Kodirov, E.; and Gong, S. 2015b. Zero-shot object recognition by semantic manifold distance. In *CVPR*, 2635–2644.

Fu, Y., and Sigal, L. 2016. Semi-supervised vocabulary-informed learning. In *CVPR*, 5337–5346.

Guo, Y.; Ding, G.; Jin, X.; and Wang, J. 2016. Transductive zero-shot recognition via shared model space learning. In *AAAI*, volume 3, 8.

Kingma, D. P.; Mohamed, S.; Rezende, D. J.; and Welling, M. 2014a. Semi-supervised learning with deep generative models. In *NIPS*, 3581–3589.

Kingma, D. P., and Welling, M. 2014b. Auto-encoding variational bayes. In *ICLR*.

Kodirov, E.; Xiang, T.; Fu, Z.; and Gong, S. 2015. Unsupervised domain adaptation for zero-shot learning. In *ICCV*, 2452–2460.

Kodirov, E.; Xiang, T.; and Gong, S. 2017. Semantic autoencoder for zero-shot learning. In *CVPR*.

Lake, B. M.; Salakhutdinov, R.; and Tenenbaum, J. B. 2015. Human-level concept learning through probabilistic program induction. *Science* 350(6266):1332–1338.

Lampert, C. H.; Nickisch, H.; and Harmeling, S. 2014. Attribute-based classification for zero-shot visual object categorization. *TPAMI* 36(3):453–465.

Li, X.; Guo, Y.; and Schuurmans, D. 2015. Semi-supervised zero-shot classification with label representation learning. In *ICCV*, 4211–4219.

Li, Y., and Wang, D. 2017. Zero-shot learning with generative latent prototype model. *arXiv preprint arXiv:1705.09474*.

Maaten, L. v. d., and Hinton, G. 2008. Visualizing data using t-sne. *JMLR* 9(Nov):2579–2605.

Mensink, T.; Gavves, E.; and Snoek, C. G. 2014. Costa: Co-occurrence statistics for zero-shot classification. In *CVPR*, 2441–2448.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, 3111–3119.

Nigam, K., and Ghani, R. 2000. Analyzing the effectiveness and applicability of co-training. In *Proceedings of the ninth international conference on Information and knowledge management*, 86–93. ACM.

Norouzi, M.; Mikolov, T.; Bengio, S.; Singer, Y.; Shlens, J.; Frome, A.; Corrado, G. S.; and Dean, J. 2013. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*.

Patterson, G., and Hays, J. 2012. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*, 2751–2758. IEEE.

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

Ravi, S., and Larochelle, H. 2017. Optimization as a model for few-shot learning. In *ICLR*, volume 1, 6.

Rezende, D. J.; Mohamed, S.; and Wierstra, D. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 1278–1286.

Rezende, D.; Danihelka, I.; Gregor, K.; Wierstra, D.; et al. 2016. One-shot generalization in deep generative models. In *ICML*, 1521–1529.

Rohrbach, M.; Ebert, S.; and Schiele, B. 2013. Transfer learning in a transductive setting. In *NIPS*.

Romera-Paredes, B., and Torr, P. H. 2015. An embarrassingly simple approach to zero-shot learning. In *ICML*, 2152–2161.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *IJCV* 115(3):211–252.

Salakhutdinov, R.; Tenenbaum, J. B.; and Torralba, A. 2013. Learning with hierarchical-deep models. *TPAMI* 35(8):1958–1971.

Shen, D.; Zhang, Y.; Henao, R.; Su, Q.; and Carin, L. 2017. Deconvolutional latent-variable model for text sequence matching. *arXiv preprint arXiv:1709.07109*.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Socher, R.; Ganjoo, M.; Manning, C. D.; and Ng, A. 2013. Zero-shot learning through cross-modal transfer. In *NIPS*, 935–943.

Tsai, Y.-H. H.; Huang, L.-K.; and Salakhutdinov, R. 2017. Learning robust visual-semantic embeddings. *arXiv preprint arXiv:1703.05908*.

Verma, V. K., and Rai, P. 2017. A simple exponential family framework for zero-shot learning. *arXiv preprint arXiv:1707.08040*.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D.; et al. 2016. Matching networks for one shot learning. In *NIPS*, 3630–3638.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset.

Wang, Q., and Chen, K. 2016a. Zero-shot visual recognition via bidirectional latent embedding. *arXiv preprint arXiv:1607.02104*.

Wang, W.; Chen, C.; Chen, W.; Rai, P.; and Carin, L. 2016b. Deep metric learning with data summarization. In *ECML-PKDD*, 777–794. Springer.

Xie, J.; Girshick, R.; and Farhadi, A. 2016. Unsupervised deep embedding for clustering analysis. In *ICML*.

Yu, Y.; Ji, Z.; Guo, J.; and Pang, Y. 2017a. Transductive zero-shot learning with adaptive structural embedding. *arXiv preprint arXiv:1703.08897*.

Yu, Y.; Ji, Z.; Li, X.; Guo, J.; Zhang, Z.; Ling, H.; and Wu, F. 2017b. Transductive zero-shot learning with a self-training dictionary approach. *arXiv preprint arXiv:1703.08893*.

Zhang, Z., and Saligrama, V. 2015. Zero-shot learning via semantic similarity embedding. In *ICCV*, 4166–4174.

Zhang, Z., and Saligrama, V. 2016a. Learning joint feature adaptation for zero-shot recognition. *arXiv preprint arXiv:1611.07593*.

Zhang, Z., and Saligrama, V. 2016b. Zero-shot learning via joint latent similarity embedding. In *CVPR*, 6034–6042.

Zhao, B.; Wu, B.; Wu, T.; and Wang, Y. 2016. Zero-shot learning via revealing data distribution. *arXiv preprint arXiv:1612.00560*.