A Generative Model For Zero Shot Learning Using Conditional Variational Autoencoders

Ashish Mishra*[1], M Shiva Krishna Reddy *[1], Anurag Mittal[1], and Hema A Murthy[1]

[1]Indian Institute of Technology Madras

September 5, 2017

Abstract

Zero shot learning in image classification refers to the setting where images from some novel classes are absent in the training data. Images from the novel classes can still be correctly classified by taking cues from other modalities such as language. This setting is important in the real world since one cannot account for all the possible classes during training. We present a novel generative model for zero shot learning using conditional variational autoencoders. By extensive testing on four benchmark datasets, we show that our model can outperform the state of the art, particularly in the more realistic generalized setting where training classes can also appear at the test time along with novel classes.

1 Introduction

Availability of labeled image data has helped in making great advances in computer vision. However even the largest image dataset i.e Imagenet [1] has only 21841 classes, with many classes having very few images. Hence it is not practically possible to collect and train on all the possible classes of images. Moreover new classes come into existence every day. Human beings are excellent at recognizing novel objects that have not been visually encountered before. For instance given the information that an *auroch* is *an ancient cow, has large horns, has large built* one can easily identify an image of an auroch from other animals such as a pig or sheep. Zero shot learning tries to capture this intuition by assuming that a semantic representation of the novel class concept is available although no image from that class is available in the training set. [2, 3, 4, 5, 6]

More formally let X_{tr} and Y_{tr} represent the training images and their class labels respectively. Similarly X_{te} and Y_{te} represent the test images and their corresponding labels. The zero shot setting states that $Y_{te} \not\subset Y_{ts}$. However for each label y_i in $Y = Y_{tr} \cup Y_{te}$, we have an embedding vector called class embedding vector A_i that is semantically related to the class corresponding to that label. This vector could come from other modalities, such as language. Recently Zero shot learning has emerged as an active area of research in the interplay between vision and language. [7, 8, 9, 10, 11, 12, 13]

The problem can be viewed as finding a relation between the embedding vector of a class and the visual features of the images in that class. Indeed most zero shot learning approaches learn a projection from image space to the class embedding space. For a novel class image given at test time, the class with the closest class embedding vector to the projection in the class embedding space is assigned. Similarly it is also possible to learn a mapping function the other way round i.e from class embedding to the image space. In many models, the mapping is a simple linear function with various kinds of regularizations. The challenges faced in learning such a mapping are well documented, the primary problem being that of the domain shifting. This was first identified by [14]. The mapping learned from seen class images may not correctly capture the relationship for unseen classes. Also, the existing methods may not capture the underlying generative process of the data, which may be much more complex and non-linear.

Recent advances in unsupervised learning, have led to better architectures for learning probability distributions, primary among them being generative adversarial networks [15] and variational autoencoders [16]. These models can also be used for condition specific image generation [17]. For example one can generate

*Equal Contribution

images conditioned on attributes. A natural question to ask is: How much can this attribute specific image generation, generalize to unseen classes?

In this work we take a slightly different approach to solving the zero shot learning problem. We view the problem as a case of missing data problem, i.e data from some classes is missing. A natural way to solve such a problem would be to fill in the missing data. We propose a generative model, where in we learn the probability distribution of the image features conditioned on the class embedding vector. Once such a distribution is learned, we can sample feature vectors corresponding to any number of novel class images, since the class embedding vectors of novel classes are known. This helps us to fill the missing data corresponding to the novel classes.

ZSL models are typically evaluated in two ways. In the standard setting [18], it is assumed that the train and test classes are disjoint ($Y_{tr} \cap Y_{te} = \Phi$) i.e the training class images do not occur at test time. However this is hardly true in the real world. Hence the generalized zero shot setting has been proposed [19] where both train and test classes may occur during the test time. Note that the latter setting is much harder than the former since the classifiers are typically biased towards classes seen at training time. We present our evaluation on both settings with greatest improvements in the much harder generalized setting. We follow the evaluation protocol recently proposed by [20]. The main contributions of this paper are as follows:

- We present a different approach of the zero shot problem by viewing it as a missing data problem. We train a conditional variational autoencoder to learn the underlying probability distribution of the image features(X) conditioned on the class embedding vector(A). We show that such an approach reduces the domain shift problem that is inherent to the methods that learn a simple mapping.
- We show that our model performs better than, or comparably on four benchmark datasets. We also show that our model can be easily extended to the much harder generalized zero shot setting, where we bring an improvement of over **20%**(absolute) on the benchmark Animals with Attributes (AwA) dataset.
- Since the model is able to generate data features of previously unseen classes, our work also adds to the evidence that conditional variational autoencoders can capture the underlying distribution of the data.

The paper is organized as follows: In section-2 we present a brief overview of the various methods for zero shot learning. Section-3 presents the motivation and method description of our model. Section-4 describes the evaluation settings. Section-4 and 5 present the experiments, results, and comparison with existing methods. Finally in section-6 we present many directions for future work and a summary of our contributions.

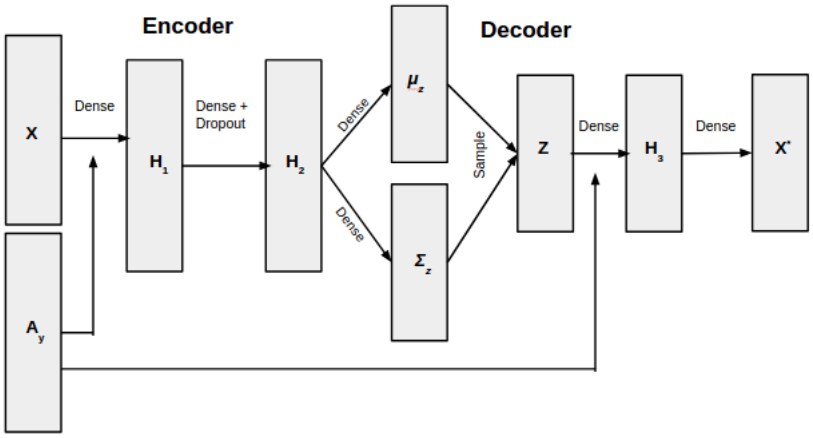
2 Related Work

Zero shot learning was first introduced by [18], where they consider disjoint train and test classes and propose an attribute based classification. Other traditional methods are based on learning an embedding from the visual space to the semantic space. During test time, for an unseen class example, the semantic vector is predicted and the nearest neighbor class is assigned [21, 22]. The embedding is learned via a parameterized mapping. The most popular approach to ZSL is learning a linear compatibility between the visual and semantic space [9, 8, 10]. [11, 13] provide novel regularizations while learning a linear compatibility function. ESZSL [11] models the relationship between features, attributes, and classes as a two linear layers network while explicitly regularizing the objective. SAE[13] adds an autoencoder loss to the projection which encourages reconstructability from attribute space to visual space.

Image classification is typically require non linear decision boundaries. Linear methods have strong bias and often are not sufficient to model the problem. Hence non linear compatibility learning methods have also been proposed. LATEM [23] proposes piecewise linear multimodel learning which learns non-linear compatibility function overall. CMT [22] trains a neural network with one hidden layer with tanh activations.

In another popular approach to ZSL, the seen class attributes are treated as the basis vectors[24] which map images(visual features) into the semantic embedding space via the convex combination of the class label embedding vectors weighed by the predictive probabilities for different training class labels. SYNC [25] tries to align semantic space to model space by learning manifold embedding of graphs composed of object classes. [26] present a sparse coding framework based on unsupervised domain adaptation for ZSL.

Figure 1: Network Architecture



The input x and the semantic class embedding vector A_y are concatenated and passed through a dense layer, followed by dropout and another dense layer. This is followed by a dense layer, that gives μ_z and Σ_z. A z is sampled from the variational distribution $\mathcal{N}(\mu_z, \Sigma_z)$. The sampled z is projected to a hidden layer via dense connections and then to the image space to reconstruct the original x. All activations are ReLU except the outputs of encoder and decoder which are linear.

A key component of zero-shot learning is the semantic embedding of class labels defined. Previous work in ZSL has used human labeled visual attributes to help detecting unseen object categories [27]. This work also uses attribute vectors as the semantic class embeddings. Distributed representations of he class name such as word2vec [28] can also be used as the semantic embedding. In this work we use word2vec for datasets where attribute vectors are not available.

Prior work on conditional image generation based on textual descriptions was successful in generating real looking images [29]. The authors also generate real looking images in zero shot setting. This serves as the primary motivation behind our approach. However we work in the feature space instead of the image space, which is an easier task.

3 Method Description

We are given a set of train classes (also called seen classes) $\mathcal{Y}_s = \{y_s^1, y_s^2, ...y_s^n\}$ and a set of test classes (also called unseen classes) $\mathcal{Y}_u = \{y_u^1, y_u^2, ...y_u^m\}$. Zero shot setting states that $\mathcal{Y}_u \nsubseteq \mathcal{Y}_s$. For each class y in $\mathcal{Y} = \mathcal{Y}_u \cup \mathcal{Y}_s$, we have a class semantic embedding vector A_y, that *describes* the class. We are given d dimensional labelled training data from the seen classes $\mathcal{Y}_s$, i.e $\{X_s, Y_s\}$. The goal is to construct a model $f : \mathbb{R}^d \rightarrow \mathcal{Y}_u$, that can classify the examples from the unseen classes $\mathcal{Y}_u$. In the generalized zero shot, we aim to construct a more generic model $f_{gen} : \mathbb{R}^d \rightarrow \mathcal{Y}_s \cup \mathcal{Y}_u$, that can classify the data from both seen and unseen classes correctly.

Variational Autoencoder [16] is a graphical model, which tries to relate the distribution of the hidden latent representations z to that of the data x. In variational inference the posterior $p(z|x)$ is approximated by a parametrized distribution $q_\Phi(z|x)$ called the variational distribution. The lower bound for $p(x)$ can be written as the following :

$$\mathcal{L}(\Phi, \theta; x) = -KL(q_\Phi(z|x)||p_\theta(z)) + \mathbb{E}_{q_\Phi(z|x)}\left[\log p_\theta(x|z)\right]$$

Here the $p_\theta(x|z)$ can be seen as a decoder, from latent space to data space, while $q_\Phi(z|x)$ can be seen as an encoder from data space to latent space. Note that one can also view this optimization as minimizing the reconstruction loss with the KL divergence as the regularizer. Conditional variational autoencoders (CVAE) first introduced by [17] maximizes the variational lower bound of the conditional likelihood $p(x|c)$ which helps to generate samples having desired properties(encoded by c).

$$\mathcal{L}(\Phi, \theta; x, c) = -KL(q_\Phi(z|x, c)||p_\theta(z|c))$$
$$+\mathbb{E}_{q_\Phi(z|c)}\left[\log p_\theta(x|z, c)\right]$$

In this work, we train the conditional variational autoencoder to generate the data features x , given the conditional variable A_y, the semantic embedding vector of a particular class. This helps us to model $p(x|A_y)$. Learning such a probability distribution helps in reducing the domain shift problem, that occurs in learning a direct mapping. The network has two components, an encoder network E with parameters Φ and a decoder D with parameters θ.

3.1 Encoder

The encoder represents the probability distribution $q(z|x, A_y)$ which we assume to be a isotropic gaussian. q is a distribution over the latent space that gives high probability mass to those z that are most likely to produce x(which belongs to class y). Thus the encoder takes in a data feature point x concatenated with the semantic embedding A_y and outputs the (μ_x, Σ_x) , the parameters of $q(z|x, A_y)$. We also want the $q(z|x, A_y)$ to be close to the standard normal distribution in the KL divergence sense.

3.2 Decoder

The decoder on the other hand, tries to map the latent space to the data space. For an input $\{z \circ A_y\}$ it tries to reconstruct that x of class y which is most likely under the latent variable z. If the CVAE is properly trained, one can use the decoder part of the network to generate any number of samples of a particular class using a simple algorithm : Sample z from a standard normal, concat A_y, and pass it through the decoder.

Algorithm 1 CVAE-ZSL

procedure ZSL CLASSIFIER(X_s, Y_s)
  $N = 200$
  $model \leftarrow$ Initialize $Encoder$, $Decoder$
  Train $model$ on X_s, Y_s
  $S = \Phi$
  for $y_u \in \mathcal{Y}_u$ **do**
    for i in $[1, 2, ..N]$ **do**
      $z \sim \mathcal{N}(0, I)$
      $V_i = y_u \circ z$
      $X_i \leftarrow Decoder(V_i)$
      $S \leftarrow S \cup \{(X_i, y_u)\}$
  $clf \leftarrow SVM$
  fit clf on S
  return clf

During training, for each training datapoint $x^{(i)}$, we estimate the $q(z^{(i)}|x^{(i)}, A_{y_i}) = \mathcal{N}(\mu_{x_i}, \Sigma_{x_i})$ using the encoder. Then a $\widetilde{z}$ is sampled from $\mathcal{N}(\mu_{x_i}, \Sigma_{x_i})$. We pass the $\widetilde{z}$ concatenated with $A_y$to the decoder and expect it to reconstruct x. We also want the $q(z|x, A_y)$ to be close to the standard normal distribution. Let x be the input to the encoder and $\hat{x}$ be the reconstructed output, the training loss becomes:

$$\mathcal{L}(\theta, \Phi; x, A_y) = \mathcal{L}_{reconstr}(x, \hat{x}) + KL\left(\mathcal{N}(\mu_x, \Sigma_x), \mathcal{N}(0, I)\right)$$

We use L_2 norm for the reconstruction loss. The KL divergence term has a nice closed form expression (see [16]) The network is shown in the Figure-1. Once the network is trained, one can sample any number of examples from each unseen class, since their semantic class embedding vectors are known. We call this the *pseudo train data*. Once the data is generated, one can train any classifier for the unseen classes. We use an SVM classifier [30] in this work. The pipeline of the zero shot classification algorithm is now simple:

1. Using X_{train}, Y_{train}, A , train the conditional variational autoencoder.
2. For each unseen class $y_u^{(i)}$, generate samples of datapoints belonging to that class. For this, sample N latent vectors (z) from a standard normal, concatenate them with $A_{y_u^{(i)}}$ and pass through the decoder part of the network. We call this generated data as pseudo traindata
3. Train an SVM classifier on the pseudo traindata.

Table 1: Dataset Details

Dataset	#images	seen/unseen
AWA-1	30475	40/10
AWA-2	37322	40/10
CUB	11788	150/50
SUN	14340	645/72
Imagenet	218000	1000/360

One straightforward way to extend our model to the generalized setting is to train the SVM with both the original training data of the seen classes, and the generated data of the unseen classes. However, we noticed that this leads to a bias towards the seen classes during classification. Hence we also generate pseudo data for seen classes along with unseen classes in the generalized zero shot setting. The change comes in step-6 of the algorithm, where $\mathcal{Y}_u$ is replaced with $\mathcal{Y}_s \cup \mathcal{Y}_u$.

4 Evaluation Protocol

Typically zero shot learning methods are evaluated in two settings. In the first setting, we make an assumption that seen classes do not occur at test time [18], i.e $Y_{tr} \cap Y_{te} = \Phi$. We call this the disjoint assumption. However this is a very strong assumption and is usually not true in the real world. Nonetheless such a setting is useful to evaluate how the training generalizes to unseen classes. In the generalized zero shot setting [19], there are no such assumptions made on the test data. Both seen and unseen class' images can occur at test time. Such a setting while being realistic, is also very difficult compared to the disjoint assumption. We follow the benchmark laid out recently by [20, 31].

5 Experiments

We present out results on four benchmark datasets Animals with Attributes(AwA)[4, 31], CUB-200-2011 Bird(CUB)[32], SUN Attribute(SUN) [33] and Imagenet [1]. We also present results on AwA, CUB, and SUN for generalized zero shot setting. While AwA is a medium sized coarse grained dataset, SUN and CUB are medium sized fine grained datasets. We also present our results on the large scale Imagenet dataset. The 1000 classes of ILSVRC2012[34] are used as seen classes, whereas 360 non-overlapping classes of ILSVRC2010 are used for testing.(same as [35, 36]). The details of test train splits in each dataset are presented in Table-1. We use keras [37] with tensorflow backend [38] for the implementation. We will make the code public for reproducibility.

5.1 Features

Similar to the recent papers, we use deep features extracted from Convolutional Neural Networks(CNN) [39] for our experiments. The images from AwA dataset are not publicly available. Hence we use VGG features [40] provided by the authors. Recently [31] have released a new dataset with the same classes as AwA, however with publicly available images. They call this AwA-2. We use Resnet101 features provided by [31] for AwA-2, CUB and SUN for fair comparison. We empirically observe that VGG net features perform slightly better(3%) for AwA. For Imagenet, we use the Alexnet features [41] for fair comparison with the competitors.

The class embedding features also play an equally important role in zero shot learning. For the AwA, SUN, and CUB datasets, we use the attribute annotations provided by the respective authors. However Imagenet has no such annotated features. Hence we use 1000 dimensional word2vec features [28] similar to [35].

5.2 Train-Test splits : An Important Note

Most recent zero shot learning models were evaluated on a particular test-train split of classes, or as an average of n (4 or 10) random splits. However as pointed out by [20] there is a significant problem with this approach. Some of the test classes overlap with the training classes of Imagenet on which the feature extraction CNN was pretrained on. This makes the model perform better on such overlapping classes, thereby showing significantly greater accuracy on such classes (contributing to greater overall accuracy).

Table 2: Results on disjoint assumption zero shot

Method	AWA-1	AWA-2	CUB	SUN
DAP	44.1	46.1	40.0	39.9
IAP	35.9	35.9	24.0	19.4
CONSE	45.6	44.5	34.3	38.8
DEVISE	54.2	59.7	52.0	56.5
ALE	59.9	62.5	54.9	58.1
SJE	65.6	61.9	53.9	53.7
ESZSL	58.2	58.6	53.9	54.5
SAE	53.0	54.1	33.3	40.3
SYNC	54.0	46.6	**55.6**	56.3
Ours	**71.4**	**65.8**	52.1	**61.7**

However such an evaluation is not representative of the true performance on the model. Hence [20] provide a novel test train split for each dataset, ensuring that none of the test classes occur in the Imagenet. They observe that the performance of all models reduces significantly with such splits. We corraborate their observation with our model also. Thus our main results are based on the novel proposed splits of [20], although we mention [1] the results on the older splits for the sake of completeness.

5.3 Parameters

The parameters of the neural network are trained with a batch size of 50 and Adam optimizer [42] with a learning rate of 10^{-3}. There are two kinds of hyper parameters in our model. The network hyperparameters (such as batch size, size of the latent variable) and the SVM cost parameter. The latent variable size was set to 100 for the medium sized datasets, and 500 for Imagenet. We empirically observe that the model is robust to the change in number of generated pseudo data samples, saturating after about 300. The SVM cost parameter was set to 100 by cross valiadation on training classes.

5.4 Evaluation Metric

For AwA, CUB, and SUN we use the average per class accuracy as the metric for a comparative evaluation with results in CITE. It is defined as follows :

$$acc_{avg}^{per-class} = \frac{1}{|Y|}\sum_{i=0}^{|Y|}\left(\frac{N_{correct}^{(class-i)}}{N_{total}^{(class-i)}}\right)$$

[31] observe that due to the class imbalance in the dataset particularly AwA, there is a significant difference (about 4%) in the average per class accuracy and the per image accuracy defined below.

$$acc_{avg}^{per-image} = \frac{N_{correct}}{N_{total}}$$

For the Imagenet dataset, we measure the top-K accuracy i.e classification of a test image is correct if the true label occurs in the top K predictions of the model. Similar to [35] and [7], we set the value of K to 5.

5.5 Generalized Zero Shot

For the generalized zero shot, we follow the protocol by [19]. From the training images from seen classes, we set aside 20% of data and train only on the remaining 80%. To reduce the bias towards seen data, we use the generator to sample data features of both seen and unseen classes to train the SVM. The SVM is evaluated separately on both the set aside seen classes data and the test data from unseen classes. As proposed by [20], we use the harmonic mean of the two accuracies as a measure of performance for Generalized zero shot.

[1] AwA:85.81%(40/10 splits, as per CITE), CUB:52.96%(ten random splits), SUN:88.5%(707/10 train/test classes, ten random splits)

Table 3: Results on Generalized zero shot

Method	AWA-1	AWA-2	CUB	SUN
DAP	0.0	0.0	3.3	7.2
IAP	4.1	1.8	0.4	1.8
CONSE	0.8	1.0	3.1	11.6
SJE	19.6	14.4	32.8	19.8
ESZSL	12.1	11.0	21.0	15.8
SYNC	16.2	18.0	19.8	13.4
DEVISE	22.4	27.8	32.8	20.9
ALE	27.5	23.9	**34.4**	26.3
Ours	**47.2**	**51.2**	**34.5**	**26.7**

Table 4: Imagenet

Method	Accuracy
AMP	13.1
DEVISE	12.8
CONSE	15.5
SS-Voc	16.8
Ours	**24.7**

5.6 Results

The results on the disjoint assumption zero shot learning with the proposed splits from [20] are presented in Table-2, and for the generalized zero shot in Table-3. On the AwA dataset which is the most used dataset for zero shot, we obtain a relative improvement of **8.7%** in the disjoint assumption zero shot. Performance is slightly lower on the CUB dataset. We suspect that this is because of the extremely fine grained classes in that dataset. On SUN which has only about 20 images per class, the results are better than the state of the art. We believe that although the classes are fine grained, our model can learn better even with less number of examples from each class. Imagenet is a much more challenging dataset, particularly due to the lack of explicit attribute vectors. On this we achieve significantly better than the competitors.

On the generalized zero shot the improvements are even more. We attribute this to two reasons: The underlying distribution is better captured using a variational auto-encoder, than by just learning a linear mapping function from image to attribute space. Also, since our model generates image features for seen classes also, it has lesser bias towards the seen classes, which is inherent to other methods. Thus our model provides an easy way to generalize to this setting.

5.7 Limitations

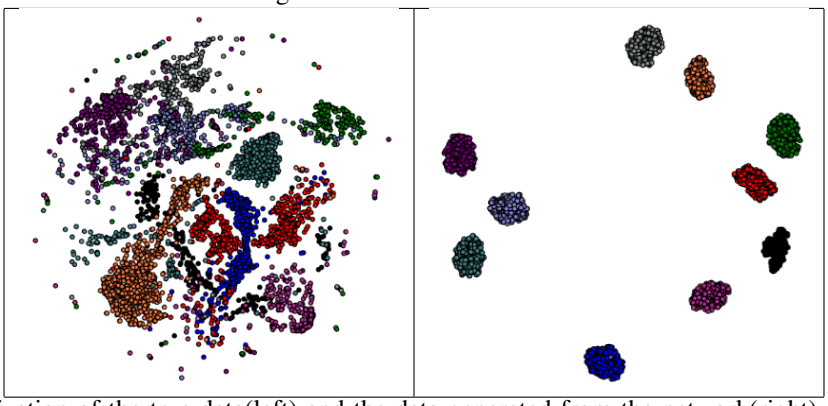
Variational autoencoder learns to map a prior distribution, in this case a multivariate normal distribution to the data distribution. It is known that a three layered neural networks can learn any mapping given enough number of intermediate layer neurons. However this is often hard to achieve. We observed that training with two hidden layers in the decoder quickly overfits to the seen classes, even with batch normalization and dropout. Challenges in training deep VAEs have been well documented [43]. Thus training deeper variational autoencoders that generalize to unseen classes is still an open problem.

Another frequent problem in generative models is that of mode collapse, i.e the distribution captures only few modes of the data distribution. This was also observed in the T-SNE [44] visualization of the generated samples from unseen classes from the AwA dataset, shown in Figure-2. The generated image features although well separated, are tightly clustered. Recently many methods for avoiding mode collapse in GANs have been studied [45], which may help to generate sufficiently diverse data. We leave this for future work.

6 Conclusion

We present a novel model for zero shot learning by modeling it as a missing data problem. We show that our model beats the state of the art in or performs comparably with the recent state of the art methods on four benchmark datasets, while outperforming them in the much harder generalized zero shot setting. A

Figure 2: T-SNE visualization



T-SNE Visualization of the true data(left) and the data generated from the network(right) for the unseen classes of AwA dataset. Note that the data is generated only from the attribute vectors of the class, without looking at even a single image. For most classes, the predicted vectors are close to the true vectors. However, the model suffers from the mode dropping problem (see red, blue).

natural extension to our model is to train other generative models such as GAN in place of CVAE or to train both the models parallely. There are several areas for improvements such as automatically generating attribute vectors for classes using wikipedia articles, end to end training to learn good features for images and class embeddings so as to obtain better representations for ZSL etc. We leave this for future work.

References

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[2] Sheng Huang, Mohamed Elhoseiny, Ahmed M. Elgammal, and Dan Yang. Learning hypergraph-regularized attribute predictors. In *CVPR*, pages 409–417. IEEE Computer Society, 2015.

[3] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In Dieter Fox and Carla P. Gomes, editors, *AAAI*, pages 646–651. AAAI Press, 2008.

[4] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(3):453–465, 2014.

[5] Marcus Rohrbach, Michael Stark, and Bernt Schiele. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In *CVPR*, pages 1641–1648. IEEE Computer Society, 2011.

[6] Xiaodong Yu and Yiannis Aloimonos. Attribute-based transfer learning for object categorization with zero/one training example. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *ECCV (5)*, volume 6315 of *Lecture Notes in Computer Science*, pages 127–140. Springer, 2010.

[7] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013.

[8] Andrea Frome, Gregory S. Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *NIPS*, pages 2121–2129, 2013.

[9] Zeynep Akata, Florent Perronnin, Zaïd Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *CoRR*, abs/1503.08677, 2015.

[10] Zeynep Akata, Scott E. Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, pages 2927–2936. IEEE Computer Society, 2015.

[11] Bernardino Romera-Paredes and Philip H. S. Torr. An embarrassingly simple approach to zero-shot learning. In Francis R. Bach and David M. Blei, editors, *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 2152–2161. JMLR.org, 2015.

[12] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. *CoRR*, abs/1603.00550, 2016.

[13] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. *CoRR*, abs/1704.08345, 2017.

[14] Yanwei Fu, Timothy M. Hospedales, Tao Xiang, and Shaogang Gong. Transductive multi-view zero-shot learning. *CoRR*, abs/1501.04560, 2015.

[15] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *NIPS*, pages 2672–2680, 2014.

[16] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013.

[17] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 3483–3491. Curran Associates, Inc., 2015.

[18] C. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[19] 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. *CoRR*, abs/1605.04253, 2016.

[20] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning - the good, the bad and the ugly. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2017.

[21] Donghui Wang, Yanan Li, Yuetan Lin, and Yueting Zhuang. Relational knowledge transfer for zero-shot learning. In *AAAI*, volume 2, page 7, 2016.

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

[23] 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*, pages 69–77, 2016.

[24] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013.

[25] 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*, pages 5327–5336, 2016.

[26] Elyor Kodirov, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Unsupervised domain adaptation for zero-shot learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2452–2460, 2015.

[27] 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*, pages 1778–1785. IEEE, 2009.

[28] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *NIPS*, pages 3111–3119, 2013.

[29] Scott E. Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. *CoRR*, abs/1605.05396, 2016.

[30] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:273–297, 1995.

[31] Yongqin Xian, H. Christoph Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning - a comprehensive evaluation of the good, the bad and the ugly. *arXiv preprint arXiv:1707.00600*, 2017.

[32] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.

[33] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*, pages 2751–2758. IEEE Computer Society, 2012.

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[35] Yanwei Fu and Leonid Sigal. Semi-supervised vocabulary-informed learning. *CoRR*, abs/1604.07093, 2016.

[36] Zhen-Yong Fu, Tao A. Xiang, Elyor Kodirov, and Shaogang Gong. Zero-shot object recognition by semantic manifold distance. In *CVPR*, pages 2635–2644. IEEE Computer Society, 2015.

[37] François Chollet et al. Keras. https://github.com/fchollet/keras, 2015.

[38] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[39] Y. Lecun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989.

[40] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[41] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[42] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[43] Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. In Daniel D. Lee, Masashi Sugiyama, Ulrike V. Luxburg, Isabelle Guyon, and Roman Garnett, editors, *NIPS*, pages 3738–3746, 2016.

[44] L.J.P. van der Maaten and G.E. Hinton. Visualizing high-dimensional data using t-sne. 2008.

[45] Mihaela Rosca, Balaji Lakshminarayanan, David Warde-Farley, and Shakir Mohamed. Variational approaches for auto-encoding generative adversarial networks. *CoRR*, abs/1706.04987, 2017.