Rethinking Zero-Shot Learning: A Conditional Visual Classification Perspective

Kai Li[1], Martin Renqiang Min[2], Yun Fu[1,3]
[1]Department of Electrical and Computer Engineering, Northeastern University, Boston, USA
[2]NEC Laboratories America
[3]College of Computer and Information Science, Northeastern University, Boston, USA
kaili@ece.neu.edu, renqiang@nec-labs.com, yunfu@ece.neu.edu

Abstract

Zero-shot learning (ZSL) aims to recognize instances of unseen classes solely based on the semantic descriptions of the classes. Existing algorithms usually formulate it as a semantic-visual correspondence problem, by learning mappings from one feature space to the other. Despite being reasonable, previous approaches essentially discard the highly precious discriminative power of visual features in an implicit way, and thus produce undesirable results. We instead reformulate ZSL as a conditioned visual classification problem, i.e., classifying visual features based on the classifiers learned from the semantic descriptions. With this reformulation, we develop algorithms targeting various ZSL settings: For the conventional setting, we propose to train a deep neural network that directly generates visual feature classifiers from the semantic attributes with an episode-based training scheme; For the generalized setting, we concatenate the learned highly discriminative classifiers for seen classes and the generated classifiers for unseen classes to classify visual features of all classes; For the transductive setting, we exploit unlabeled data to effectively calibrate the classifier generator using a novel learning-without-forgetting self-training mechanism and guide the process by a robust generalized cross-entropy loss. Extensive experiments show that our proposed algorithms significantly outperform state-of-the-art methods by large margins on most benchmark datasets in all the ZSL settings.

1. Introduction

Deep learning methods have achieved revolutionary successes on many tasks in computer vision owing to the availability of abundant labeled training data [45, 44, 17, 20, 19, 21]. However, labeling large-scale training data for each task is both labor-intensive and unscalable. Inspired by human's remarkable abilities to recognize instances of unseen classes solely based on class descriptions without seeing any visual example of such classes, researchers have extensively studied an image classification setting similar to the human learning called zero-shot learning (ZSL) [41, 31, 22, 33], in which labeled training images of seen classes and semantic descriptions of both seen classes and unseen classes are given and the task is to classify test images into seen and unseen classes.

Existing approaches usually formulate ZSL as a visual-semantic correspondence problem and learn the visual-semantic relationship from seen classes and apply it to unseen classes, considering that the seen and unseen classes are related in the semantic space [1, 43, 13]. These methods usually project either visual features or semantic features from one space to the other, or alternatively project both types of features to an intermediate embedding space. In the shared embedding space, the associations between the two types of features are utilized to guide the learning of the projection functions.

However, these methods fail to recognize the tremendous efforts in obtaining these discriminative visual features over a large number of classes through training powerful deep neural network classifiers with a huge amount of computational and data resources, and thus essentially discard the highly precious discriminative power of visual features in an implicit way. In details, on one hand, the visual features used in most ZSL methods are extracted by some powerful deep neural networks (e.g., ResNet101) trained on large-scale datasets (e.g., ImageNet) [40]. These visual features are already highly discriminative; reprojecting them to any space shall impair the discriminability, especially to a lower dimensional space, because the dimension reduction usually significantly shrinks data variance. It is surprising that the majority of existing ZSL approaches try to transform the visual feature vectors in various ways [22, 33, 13]. On the other hand, by nature of classification problems, the competition information among different classes are crucial for classification performance. But many ZSL approaches ignore the class separation information during training due to focusing on learning the associations between visual and semantic features, and fail to realize that ZSL is essentially a classification problem [43].

Inspired by the above observations, we propose to solve ZSL in a novel conditional visual feature classification framework. In the proposed framework, we effectively generate visual feature classifiers from the semantic attributes, and thus intrinsically preserve the visual feature discriminability while exploiting the competing information among different classes. Within the novel framework, we propose various novel strategies to address different ZSL problems.

For the conventional ZSL problem where only unseen classes are involved for evaluations, we propose to train a deep neural network that generates visual feature classifiers directly from the semantic attributes. We train the network with a Cosine similarity based cross-entropy loss, which mitigates the impact of variances of features from two different domains when calculating their correlations. Borrowing ideas from meta-learning, we train our model in an episode-based way by composing numerous "fake" new ZSL tasks, so that its generalizability to "real" new ZSL tasks during test is enhanced. For the generalized setting in which seen classes are included for ZSL evaluations, we concatenate the classifiers for seen classes and unseen classes to classify visual features for all classes. Since the classifiers for seen classes are trained with labeled samples, they are highly discriminative to discern whether an incoming image belongs to the seen classes or not. This desirable property prevents our method from significant performance drops when much more classes are involved for evaluations. For the transductive setting in which images of unseen classes are available [34], we take advantage of these unlabeled data to calibrate our classifier generator using the pseudo labels generated by itself. To limit the harm of incorrect pseudo labels and avoid the model being over-adapted to new classes, we propose to use the generalized cross-entropy loss to guide the model calibration process under an effective learning-without-forgetting training scheme.

In summary, our contributions are as follows:

- We reformulate of ZSL as a conditional visual classification problem, by which we can essentially benefit from high discriminability of visual features and inter-class competing information among training classes to solve ZSL problem in various settings.
- We propose various effective techniques to address different ZSL problems uniformly within the proposed framework.
- Experiments show that our algorithms significantly outperform state-of-the-art methods by large margins on most benchmark datasets in all the ZSL settings.

2. Related Work

Zero-Shot Learning (ZSL) aims to recognize unseen classes based on their semantic associations with seen classes. The semantic associations could be within the human-annotated attributes [34, 26, 2], word vectors [10, 43, 4], text descriptions [16, 6], etc. In practice, ZSL is performed by firstly learning an embedding space where semantic vectors and visual features are interacted. Then, within the learned embedding space, the best match among semantic vectors of unseen classes is selected for the visual features of any given image of the unseen classes.

According to the embedding space used, existing methods can be generally categorized into the following three groups. Some approaches select semantic space as embedding space and project visual features to semantic space [15, 10]. Projecting visual features into a often much lower-dimensional semantic space shall shrink the variance of the projected data points and thus aggravate the hubness problem, i.e., some candidates will be biased to be the best matches to many of the queries. Alternatively, some methods project both visual and semantic features into a common intermediate space [1, 35, 47]. However, due to the lack of training samples from unseen classes, these methods are prone to classifying test samples into seen classes [30]. The third category of methods choose the visual space as the embedding space and learned a mapping from the semantic space to visual space [43]. Benefiting from the abundant data diversity in visual space, these methods can mitigate the hubness problem to some extent.

Recently, a new branch of methods come out and approach ZSL in virtue of data augmentation, either by variational auto-encoder (VAE) [25] or Generative Adversarial Network (GAN) [5, 42, 8, 48, 50]. These methods learn from visual and semantic features of seen classes generators that can generate synthesized visual features based on class semantic descriptions. Then, synthesized visual features are used to train a standard classifier for object recognition.

ZSL may turn easier when unlabelled test samples from unseen classes are available during training, i.e., the so-called transductive ZSL. This is because unlabelled test samples can be utilized to help reach clearer decision boundaries for both seen and unseen classes. In fact, it is more like a semi-supervised learning problem. Propagated Semantic Transfer (PST) [29] conducts label propagation from seen classes to unseen classes through exploiting the class manifold structure. Unsupervised Domain Adaption (UDA) [13] formulates the problem as a cross-domain data association problem and solves it by regularized sparse coding. Quasi-Fully Supervised Learning (QFSL) [34] aims to strength the mapping from visual space to semantic space by explicitly requiring the visual features being mapped to the categories (seen and unseen) they belong.

Unlike the above methods, we approach ZSL from the perspective of conditioned visual feature classification. Perhaps most similar to our algorithms are [16, 38], which approach ZSL also by generating classifiers. However, [16] projects visual features to a lower dimensional space, harming discriminability of the visual features. [38] uses graph convolutional network to model the semantic relationships and output classifiers. However, it requires categorical relationship as additional input. We instead generate classifiers directly from attributes by a deep neural network and train the model with a novel cosine similarity based cross-entropy loss. Besides, neither of the two methods uses episode-based training to enhance model adaptability to novel classes. Moreover, they are only feasible for the conventional ZSL setting, while our method is flexible for various ZSL settings.

3. Method

Zero-shot learning (ZSL) is to recognize objects of unseen classes given only semantic descriptions of the classes. Formally, suppose we have three sets of data $\mathcal{D} = \{\mathcal{D}_s, \mathcal{D}_a, \mathcal{D}_u\}$, where $\mathcal{D}_s = \{\mathcal{X}_s, \mathcal{Y}_s\}$ and $\mathcal{D}_u = \{\mathcal{X}_u, \mathcal{Y}_u\}$ are training and test sets, respectively. $\mathcal{X}_s$ and $\mathcal{X}_u$ are the images, while $\mathcal{Y}_s$ and $\mathcal{Y}_u$ the corresponding labels. There is no overlap between training classes and test classes, i.e., $\mathcal{Y}_s \cap \mathcal{Y}_u = \emptyset$. The goal of ZSL is to learn transferable information from $\mathcal{D}_s$ that can be used to classify unseen classes from $\mathcal{D}_u$, with the help of semantic descriptions $\mathcal{D}_a = \mathcal{A}_s \cup \mathcal{A}_u$ for both seen ($\mathcal{A}_s$) and unseen ($\mathcal{A}_u$) classes. $\mathcal{D}_a$ can be human-annotated class attributes [42] or articles describing the classes [49].

We solve ZSL in a conditional visual feature classification framework. Specifically, we predict $p(y|\mathbf{x}; \mathbf{a}_y)$ of an image $\mathbf{x}$ belonging to class y given the semantic description $\mathbf{a}_y$ of the class, where $y \in \mathcal{Y}_u$ in the standard setting, while $y \in \mathcal{Y}_s \cup \mathcal{Y}_u$ in the generalized setting. When $\mathcal{X}_u$ is available during training, we call the problem transductive ZSL. For convenience, sometimes we call the setting inductive ZSL where $\mathcal{X}_u$ is unavailable.

3.1. Zero-Shot Learning

By approaching ZSL in virtue of visual classification conditioned on attributes, we need to generate visual feature classifiers from the attributes. We achieve this by learning a deep neural network f which takes a semantic feature vector of a class as input and outputs the classifier weight vector for the class. Since the model f is going to generate classifiers for novel classes when tested, we adopt the episode-based training mechanism, an effective and popular technique in meta-learning [37, 9, 18], to mimic this scenario during training.

The key to episode-based training is to sample in each mini-batch a "fake" new task that matches the scenario where the model is tested, and train the model with the sampled task. This process is called an episode. The goal is to expose the model with numerous "faked" new tasks during training, such that it can generalize better for real new tasks when tested. To construct an ZSL episode, we keep randomly sampling from $\mathcal{D}_t = \{\mathcal{X}_t, \mathcal{Y}_t\}$ and $\mathcal{A}_t$ a ZSL task $\mathcal{T} = \{\mathcal{V}, \mathcal{A}\}$ where $\mathcal{V} = \{\mathbf{x}_{i,j}\}_{i=1}^{N}, y_j\}_{j=1}^{M}$ contains N labeled samples from each of the M classes. Note for each sample $(\mathbf{x}_{i,j}, y_j)$, we dismiss its global (dataset-wise) label and replace it with a local (minibatch-wise) label, while still keeping the class separation information (samples of the same global label still have the same local label), i.e., $y_j \in \{1, 2, \cdots, M\}$. This is to cut off the connections between individual tasks brought by the shared global label pool, so that each mini-batch can be treated as a new task. $\mathcal{A} = \{\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_M\}$ is the associated M attribute vectors.

Algorithm 1. Proposed ZSL approach

```
Input: Training set $\mathcal{D}_s = \{\mathcal{X}_s, \mathcal{Y}_s\}$ and attributes $\mathcal{A}_s$.
Output: Classifier weight generation network $f$
while not done do
    1. Randomly sample from $\mathcal{D}_s$ and $\mathcal{A}_s$ a ZSL task
       $\mathcal{T} = \{\mathcal{V}, \mathcal{A}\}$, where $\mathcal{V} = \{\{\mathbf{x}_{i,j}\}_{i=1}^{N}, y_j\}_{j=1}^{M}$
       and $\mathcal{A} = \{\mathbf{a}_j\}_{j=1}^{M}$.
    2. Calculate loss according to Eq. (3)
    3. Update $f$ through back-propagation.
end while
```

For each task $\mathcal{T} = \{\mathcal{V}, \mathcal{A}\}$, f generates a classifier for the M sampled classes as

$$\mathbf{W} = f(\mathcal{A}). \tag{1}$$

With the classifier $\mathbf{W}$, we can calculate classification scores of visual features from $\mathcal{V}$. Rather than using the extensively used dot product, we use cosine similarity.

Cosine similarity based classification score function. Traditional multi-layer neural networks use dot product between the output vector of previous layer and the incoming weight vector as the input to activation function. [23, 11] recently showed that replacing the dot product with cosine similarity can bound and reduce the variance of the neurons and thus result in models of better generalization. Considering that we are trying to calculate the correlation between data from two dramatically different domains, especially for the attribute domain in which the features are discontinuous and have high variances. Using cosine similarity shall mitigate the harmful effect of the high variances and bring us desirable Softmax activations. With this consideration, we define our classification score function as

$$p(y = i|\mathbf{x}) = \frac{\exp(\sigma \cos(\mathbf{w}_i, \mathbf{x}))}{\sum_{j=1}^{N} \exp(\sigma \cos(\mathbf{w}_j, \mathbf{x}))}, \tag{2}$$

where σ is a learnable scalar controlling the peakiness of the probability distribution generated by the Softmax operator. $\mathbf{w}_i$ is the classifier weight vector for class i.

With this definition, the loss of a typical ZSL task $\mathcal{T}$ is defined as follows,

$$\begin{aligned}\mathcal{L} &= \textstyle\sum_{(\mathbf{x},y)\in\mathcal{T}} \big[-\sigma\cos(\mathbf{w}_i,\mathbf{x}) + \\ &\log(\textstyle\sum_{j=1}^{N} \exp(\sigma\cos(\mathbf{w}_j,\mathbf{x})))\big] + \lambda\|\phi\|_2,\end{aligned} \tag{3}$$

where λ is a hyper-parameter weighting the l_2-norm regularization of the learnable parameters of neural network f_ϕ.

Algorithm 1 outlines our training procedures.

3.2. Generalized Zero-Shot Learning

With the learned classifier generator f, given attributes of unseen classes $\mathcal{A}_u$ in the test stage, we generate the corresponding classifier weights $\mathbf{W}_u = f(\mathcal{A}_u)$ and use it to classify visual features of unseen classes $\mathcal{X}_u$ according to Eq. (2).

When both seen and unseen classes are involved for evaluations, i.e., the generalized ZSL setting, we combine the classifiers for both seen and unseen classes to classify images from all classes. Specifically, with $\mathcal{A}_u$ and $\mathcal{A}_s$, we can get classifiers $\mathbf{W}_u = f(\mathcal{A}_u)$ and $\mathbf{W}_s = f(\mathcal{A}_s)$ for unseen and seen classes, respectively. We use their concatenation $\mathbf{W}_b = [\mathbf{W}_u, \mathbf{W}_s]$ as the classifier for all classes.

It is worth noting that since f has already been trained with labeled samples, the resulting $\mathbf{W}_s$ should be very discriminative to discern whether an incoming image belongs to the seen classes or not. As will be shown later in the experiments, this desirable property prevents our method from significant recognition accuracy drops when much more classes are involved for evaluations.

3.3. Transductive Zero-Shot Learning

Thanks to the conditional visual classification formulation of ZSL, the above inductive approach can be readily adapted to the transductive ZSL setting. We can utilize test data during training to calibrate our classifier generator and output classifiers of better decision boundaries for both seen and unseen classes. We achieve this in virtue of self-training. Specifically, we alternate between generating pseudo labels for images of unseen classes using the classifier generator and updating it using the generated pseudo labels. With this idea, two key problems need to be solved. The first is how to prevent the generator from over-adapting to unseen classes such that the knowledge previously learned from seen classes is lost, resulting in unsatisfactory performance for seen classes. The second is how to avoid the generator being impaired by the incorrect pseudo labels. We propose a novel self-training based transductive ZSL algorithm to avoid both problems. Figure 1 illustrates our algorithm.

To generate pseudo labels for test images $\mathcal{X}_u$, we first generate classifier weights $\mathbf{W}_u$ for unseen classes as

$$\mathbf{W}_u = f(\mathcal{A}_u). \tag{4}$$

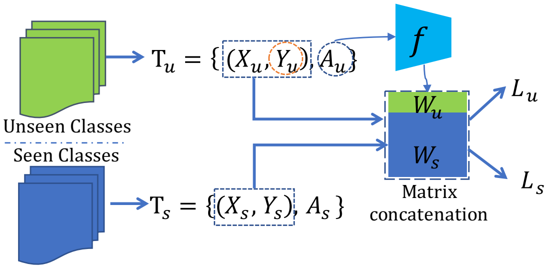

Unseen Classes
Seen Classes
$\mathrm{T}_u = \{(X_u, Y_u), A_u\}$
f
W_u
W_s
L_u
L_s
Matrix concatenation
$\mathrm{T}_s = \{(X_s, Y_s), A_s\}$

Figure 1. Illustration of the transductive ZSL algorithm. We sample ZSL tasks $\mathcal{T}_s$ from seen classes and $\mathcal{T}_u$ from unseen classes (with pseudo labels). The classifier $\mathbf{W}_u$ generated from $\mathbf{A}_u$ are concatenated with classifier $\mathbf{W}_s$ to classify visual features from both $\mathcal{T}_u$ and $\mathcal{T}_s$, which results in loss $\mathcal{L}_u$ and $\mathcal{L}_s$, respectively. The pseudo labels for unseen classes are updated in a self-training way.

With $\mathbf{W}_u$, we calculate classification score $\mathbf{S}$ of $\mathcal{X}_u$ according to Eq. (2). Pseudo labels $\tilde{\mathcal{Y}}_u$ of $\mathcal{X}_u$ can be obtained from $\mathbf{S}$. There inevitably exist noises among $\tilde{\mathcal{Y}}_u$. We propose to mitigate their impact by a novel classification score peakiness based filtering strategy.

Let $\mathbf{s}^i \in \mathbb{R}^{N_u}$ be the classification score of $\mathbf{u}_i \in \mathcal{X}_u$ according to all the N_u classes. Let $s^i_{y_m}$ and $s^i_{y_n}$ be the highest and second highest score among $\mathbf{s}^i$. The pseudo label assigned to $\mathbf{u}_i$ should be y_m. However, we regard this assignment as a "confident" one unless $s^i_{y_m}$ is peaky enough:

$$\frac{s^i_{y_m}}{s^i_{y_n}} > \gamma, \tag{5}$$

where γ is a threshold controlling the peakiness. This constraint prevents ambiguous label assignment from being exploited for classifier generator calibration.

After obtaining the confident set $\hat{\mathcal{D}}_u = \{\hat{\mathcal{X}}_u, \hat{\mathcal{Y}}_u\}$, as well as the the corresponding attributes $\hat{\mathcal{A}}_u$, we can use them to adjust f. However, finetuning f with only $\hat{\mathcal{D}}_u$ shall cause strong bias towards unseen classes such that the knowledge previously acquired about seen classes will be forgotten after a few iterations. What is worse, the incorrect pseudo labels among $\hat{\mathcal{Y}}_s$ may damage f when they are of a high portion. We propose a novel learning-without-forgetting training scheme to avoid this.

Along with sampling a ZSL task $\mathcal{T}_u$ from ($\hat{\mathcal{D}}_u$, $\hat{\mathcal{A}}_u$) to calibrate f to unseen classes, we sample another ZSL task $\mathcal{T}_s$ from ($\mathcal{D}_s$, $\mathcal{A}_s$) to keep the memory of f to seen classes and dilute the impact of noisy labels from $\mathcal{T}_u$. Further, while updating f, we update as well classifier $\mathbf{W}_s$ to adjust the decision boundaries of seen classes towards unseen ones.

Moreover, we introduce the very recently proposed generalized cross-entropy loss [46] to handle task $\mathcal{T}_u$ and limit the impact of incorrect pseudo labels to the classifier weight generator:

$$\mathcal{L}_u = \sum_{(\mathbf{x}_u, y_u)\in\mathcal{T}_u} \frac{1-(\mathbf{w}_{y_u})^q}{q}, \tag{6}$$

```
Algorithm 2. Proposed approach for transductive ZSL
Input: Training set $\mathcal{D}_s = \{\mathcal{X}_s, \mathcal{Y}_s\}$, attribute set
  $\mathcal{D}_a = \mathcal{A}_s \cup \mathcal{A}_u$, and test images $\mathcal{X}_u$, parameters $\gamma$ and $q$
Output: Class label $\tilde{\mathcal{Y}}_u$ of $\mathcal{X}_u$, weight generator $f$,
  classifier weight $\mathbf{W}_s$ for seen classes.
1. Obtain $f$ with $\mathcal{D}_s$ and $\mathcal{A}_s$ using Algorithm 1.
2. Obtain $\mathbf{W}_s = f(\mathcal{A}_s)$.
for $r = 1, 2, ...N_r$ do
    3. Calculate classifier weights for unseen classes
       $\mathbf{W}_u = f(\mathcal{A}_u)$.
    4. Generate pseudo labels $\tilde{\mathcal{Y}}_u$ for $\mathcal{X}_u$ according to Eq. (2).
    5. Select confident test set $\hat{\mathcal{D}}_u = \{\hat{\mathcal{X}}_u, \hat{\mathcal{Y}}_u\}$ and $\hat{\mathcal{A}}_u$
       based on Eq. (5).
    for $i = 1, 2, ...N_i$ do
        6. Sample ZSL tasks $\mathcal{T}_s$ from ($\mathcal{D}_s$, $\mathcal{A}_s$), and
           $\mathcal{T}_u$ from ($\hat{\mathcal{D}}_u$ $\hat{\mathcal{A}}_u$).
        7. Calculate loss according to Eq. (7).
        8. Update $f$ and $\mathbf{W}_s$ through back-propagation.
    end while
end while
```

where $\mathbf{w}_{y_u}$ is the possibility of $\mathbf{x}_u$ belonging to class y_u, which is calculated according to Eq. (2). $q \in$ (0,1] is a hyper-parameter of which a higher value is preferred when the noise level is high. It can be shown that Eq. (6) turns to Eq. (3) when q infinitely approaches 0. On the other hand, it turns to the Mean Absolute Error (MAE) loss when $q = 1$. Cross-entropy loss is powerful for classification tasks but noise-sensitive, while MAE loss performs worse for conventional classification task but is robust to noisy labels. Tuning q between 0 and 1 fits different noise levels.

By handling $\mathcal{T}_u$ with generalized cross-entropy loss and $\mathcal{T}_s$ with conventional cross-entropy loss, our loss function for the transductive ZSL is as follows:

$$\mathcal{L}(\phi, \mathbf{W}_s) = \mathcal{L}_u + \mathcal{L}_s, \quad (7)$$

where $\mathcal{L}_s$ is defined in Eq (3). **Algorithm 2** outlines the training procedures.

4. Experiments

4.1. Datasets and Evaluation Settings

We employ the most widely-used zero-shot learning datasets for performance evaluation, namely, CUB [39] AwA1 [15], AwA2 [41], SUN [28] and aPY [7]. The statistics of the datasets are shown in Table 1. We follow the GBU setting proposed in [41] and evaluate both the conventional ZSL setting and the generalized ZSL (GZSL) setting. In the conventional ZSL, test samples are restricted to the unseen classes, while in GZSL, they may come from either seen classes or unseen classes. For both settings, we use top-1 (T1) Mean Class Accuracy (MCA) as the evaluation metric in our experiments. For GZSL, we evaluate the MCA for both seen (S) and unseen classes (U), and also calculate their harmonic mean $H = 2 * U * S/(U + S)$.

		CUB	AwA1	AwA2	aPY	SUN
#Class	#Seen	150	40	40	20	645
	#Unseen	50	10	10	12	72
# VisDim		2048	2048	2048	2048	2048
# AttDim		312	85	85	312	102

Table 1. Information of zero-shot classification datasets.

4.2. Implementation details

Following [41], we use ResNet101 [12] trained on ImageNet for feature extraction, which results in a 2048-dimension vector for each input image. The classifier generation model f consists of two pairs of FC+ReLU layers, i.e., FC-ReLU-FC-ReLU, which maps semantic vectors to visual classifier weights. The dimension of the intermediate hidden layer is 1600 for all the five datasets. We train f with Adam optimizer and a learning rate 10^{-5} for all datasets by 1,000,000 randomly sample ZSL tasks. Each task consists of 32 randomly sampled classes, 4 samples for each class, i.e., $M = 32$ and $N = 4$, except aPY where we set $M = 16$ and $N = 4$ because there are in total only 20 classes for training. The hyper-parameter λ is chosen as 10^{-4}, 10^{-3}, 10^{-3}, 10^{-5} and 10^{-4} for AwA1, AwA2, CUB, SUN and aPY, respectively.

For transductive ZSL, the experimental setting is the same as that in the corresponding inductive case for each dataset. For all the datasets, we update the pseudo labels of unseen classes every 10,000 iterations and execute 50 updates, i.e., $N_r = 50$ and $N_i = 10,000$. We apply $\gamma = 1.2$ and $q = 0.5$ for all the datasets. We develop our algorithms based on PyTorch.

4.3. Ablation Studies

By formulating ZSL as a visual classification problem conditioned on the attributes, we can naturally benefit from the high discriminability of visual features. Meanwhile, to combat with the significant variance of visual and attribute features, we propose to replace the widely-used dot product with cosine similarity to calculate the classification score. Moreover, we introduce the episode-based training scheme to enhance the adaptability of our model to new tasks. We conduct ablation study to evaluate the effectiveness of our ingenious designs.

Preserving visual feature discriminability. To study the importance of preserving visual discriminability, we implement two baseline methods: one we project visual features to attribute space and the other we project visual features to an intermediate space (of half dimension as the visual space). All the other settings are the same as our method.

Table 2 shows that the performance degrades significantly by projecting visual features to either semantic space

V→A	✓	✓					
V → I ← A			✓	✓			
A→V					✓	✓	✓
Dot product	✓		✓		✓		
Cosine similarity		✓		✓		✓	✓
Episode based training							✓
ZSL	36.3	45.1	34.2	42.8	27.0	67.7	70.9
GZSL-U	24.5	10.1	25.9	11.2	22.7	59.8	62.7
GZSL-S	62.5	86.8	68.9	81.8	53.2	75.2	77.0
GZSL-H	35.2	18.0	37.6	19.6	31.9	66.6	69.1

Table 2. Ablation study on the AwA1 dataset. "V→A", "A→V", and "V → I ← A" refer to projecting visual features to attribute space, projecting attributes to visual space, and projecting both visual and attribute features into an intermediate space, respectively.

or intermediate space, no matter using dot product or Cosine similarity based classification score functions. As analyzed before, image feature embeddings for ZSL are usually generated offline by some powerful feature extraction networks such that high discriminatibility has already been secured. Reprojecting them to either attribute or intermediate space shall inevitably impair the discriminability. What is worse, the attribute space or the intermediate space are often of lower dimension than the visual embedding space. The visual variance, which is crucial to ensure discriminability, shall be shrunk once the feature embeddings are reprojected to the lower-dimensional spaces. Due to the damage of the discriminability of visual features, the hubness problem becomes even more intense, leading to much worse results.

Cosine similarity based classification score function. We compare dot product and cosine similarity based loss functions under all the three classification spaces. Table 2 shows that the classification space seems a more dominant factor: neither of the two score functions works well if the classification space is not appropriate. When the visual embedding space is selected for classification, the proposed cosine similarity based score function results in much better performance than that based on dot product. We speculate the reason is that values of class attribute are not continuous such that there are large variance among the attribute vectors of different classes. Consequently, classifier weights derived from them also possess large variance, which might cause high variances of inputs to the Softmax activation function [23]. Unlike dot product, our cosine similarity based score function normalizes the classifier weights before calculating its dot product with visual embeddings. This normalization procedure can bound and reduce the variance of the classifier weights, contributing to better performance.

Episode-based training mechanism The proposed episode-based training mechanism is to train our classifier weight generator in the way it works during test. From Table 2, we can observe that there are about 3% performance gains for both the ZSL setting and GZSL setting when this unique training mechanism is adopted. This is within our expectation because after exposing our weight generator with numerous (faked) new ZSL tasks during the training, it acquires the knowledge how to deal with real new ZSL tasks during the test. So, better performance is more likely to be guaranteed.

4.4. Comparative Results

Zero-shot learning. Table 3 shows the comparative results of the proposed method and the state-of-the-art ones for the inductive ZSL problem. For conventional ZSL, our method reaches the best for three out of the five datasets. Remarkably, for the AwA2 dataset, our method beats the second best by about 4%.

Generalized zero-shot learning. More interesting observations can be made for the GZSL setting where classification is performed over both seen and unseen classes. With more classes involved, the classification accuracy of unseen classes drops for all methods. However, our method exhibits much more robustness than the other ones and drops moderately on these datasets. Remarkably, our method sometimes secures accuracy that is even by about 100% (*aPY*) higher than the second best. We analyze this striking improvements owning to our consideration of inter-class separation during training so that the resultant classifiers for seen classes possess good separation property after training. When they are combined with classifiers generated from semantic descriptions of unseen classes in test, they shall be highly discriminative to discern the incoming images do not belong to the classes they were trained for.

Contrary to the striking advantages for recognizing unseen classes, our method seems kind of "forgetful" and is overcome by many methods for the accuracy of seen classes. This is because during training, we constantly sample new ZSL tasks to train the weight generator to acquire the knowledge of handling new ZSL tasks. Unlike existing methods, which process the whole dataset altogether or are specially designed to keep the training memory, our method does not memorize the global class structure of the whole training set. Therefore, with the increase of the capability of handle new ZSL tasks, it is inevitably sacrifice some competence of classifying seen classes. Despite of this, our method surpasses the other ones by large margins for three our of the five datasets for the harmonic mean (H), while being very close to the feature synthesized based method.

Transductive zero-shot learning. When test data are available during training, better performance is often expected as we can utilize them to mitigate the bias of models towards seen classes. Table 4 verifies this and our transductive algorithm significantly outperform the inductive counterpart. This substantiates the effectiveness of our novel learning-without-forgetting self-training technique. Further, with generalized cross-entropy loss for unseen classes, Ours-trans (GXE) consistently performs better than that with con-

	SUN				CUB				AWA1				AWA2				aPY			
	ZSL	GZSL			ZSL	GZSL			ZSL	GZSL			ZSL	GZSL			ZSL	GZSL		
	T1	U	S	H	T1	U	S	H	T1	U	S	H	T1	U	S	H	T1	U	S	H
LATEM [40]	55.3	14.7	28.8	19.5	49.3	15.2	57.3	24.0	55.1	7.3	71.7	13.3	55.8	11.5	77.3	20.0	35.2	0.1	73.0	0.2
ALE [1]	58.1	21.8	33.1	26.3	54.9	23.7	62.8	34.4	59.9	16.8	76.1	27.5	62.5	14.0	81.8	23.9	39.7	4.6	73.7	8.7
DEVISE [10]	56.5	16.9	27.4	20.9	52.0	23.8	53.0	32.8	54.2	13.4	68.7	22.4	59.7	17.1	74.7	27.8	**39.8**	4.9	76.9	9.2
SJE [1]	53.7	14.7	30.5	19.8	53.9	23.5	59.2	33.6	65.6	11.3	74.6	19.6	61.9	8.0	73.9	14.4	32.9	3.7	55.7	6.9
ESZSL [30]	54.5	11.0	27.9	15.8	53.9	12.6	63.8	21.0	58.2	6.6	75.6	12.1	58.6	5.9	77.8	11.0	38.3	2.4	70.1	4.6
SYNC [3]	56.3	7.9	**43.3**	13.4	55.6	11.5	70.9	19.8	54.0	8.9	87.3	16.2	46.6	10.0	90.5	18.0	23.9	7.4	66.3	13.3
SAE ([14])	40.3	8.8	18.0	11.8	33.3	7.8	54.0	13.6	53.0	1.8	77.1	3.5	54.1	1.1	82.2	2.2	8.3	0.4	80.9	0.9
GFZSL [36]	60.6	0.0	39.6	0.0	49.3	0.0	45.7	0.0	68.3	1.8	80.3	3.5	63.8	2.5	80.1	4.8	38.4	0.0	83.3	0.0
DEM [43]	61.9	20.5	34.3	25.6	51.7	19.6	57.9	29.2	68.4	32.8	84.7	47.3	67.2	30.5	86.4	45.1	35.0	11.1	75.1	19.4
Relat. Net [35]	-	-	-	-	55.6	38.1	61.1	47.0	68.2	31.4	**91.3**	46.7	64.2	30.0	**93.4**	45.3	-	-	-	-
SP-AEN [5]	59.2	24.9	38.6	30.3	55.4	34.7	70.6	46.6	-	-	-	-	58.5	23.3	90.9	37.1	24.1	13.7	63.4	22.6
PSR [2]	61.4	20.8	37.2	26.7	56.0	24.6	54.3	33.9	-	-	-	-	63.8	20.7	73.8	32.3	38.4	13.5	51.4	21.4
f-CLSWGAN* [42]	60.8	**42.6**	36.6	**39.4**	**57.3**	**57.7**	43.7	**49.7**	68.2	43.7	57.7	49.7	-	-	-	-	-	-	-	-
Ours	**62.6**	36.3	42.8	39.3	54.4	47.4	47.6	47.5	**70.9**	**62.7**	77.0	**69.1**	**71.1**	**56.4**	81.4	**66.7**	38.0	**26.5**	74.0	**39.0**

Table 3. Zero-shot learning accuracy. The best results are in **bold**. The model with * (f-CLSWGAN) generates additional data for training while the remaining models do not.

	SUN				CUB				AWA1				AWA2				aPY			
	ZSL	GZSL			ZSL	GZSL			ZSL	GZSL			ZSL	GZSL			ZSL	GZSL		
	T1	U	S	H	T1	U	S	H	T1	U	S	H	T1	U	S	H	T1	U	S	H
ALE-tran [1]	55.7	19.9	22.6	21.2	54.5	23.5	45.1	30.9	65.6	25.9	-	-	70.7	12.6	73.0	21.5	46.7	8.1	-	-
GFZSL-tran [36]	**64.0**	0.0	41.6	0.0	49.3	24.9	45.8	32.2	81.3	48.1	-	-	78.6	31.7	67.2	43.1	37.1	0.0	-	-
DSRL [27]	56.8	17.7	25.0	20.7	48.7	17.3	39.0	24.0	74.7	22.3	-	-	72.8	20.8	74.7	32.6	45.5	11.9	-	-
QFSL [34]	58.3	31.2	51.3	38.8	**72.1**	**71.5**	**74.9**	**73.2**	-	-	-	-	79.7	66.2	**93.1**	77.4	-	-	-	-
Ours-trans (XE)	61.9	44.5	57.6	50.2	59.2	54.4	67.9	60.4	87.4	84.2	84.3	84.2	81.4	77.7	88.3	82.7	52.7	50.4	86.3	63.7
Ours-trans (GXE)	63.5	**45.4**	**58.1**	**51.0**	61.3	57.0	68.7	62.3	**89.8**	**87.7**	**89.0**	**88.4**	**83.2**	**80.2**	90.0	**84.8**	**54.7**	**51.8**	**87.6**	**65.1**

Table 4. Transductive zero-shot learning accuracy. The best results are in **bold**.

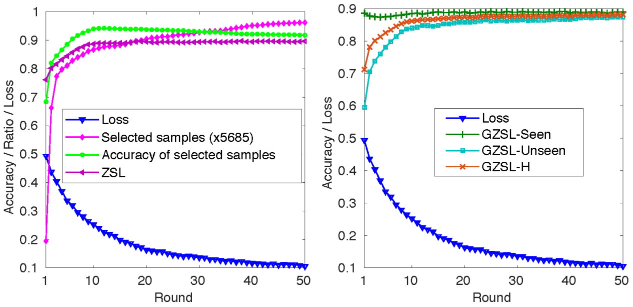


Figure 2. Analysis of the self-training process on the AwA1 dataset. "5685" stands for the total number of test images.

	c=4	c=8	c=16	c=32	c=40 (all)
ZSL	68.1	69.6	70.4	70.9	69.8

Table 5. ZSL accuracy w.r.t. training classes per batch.

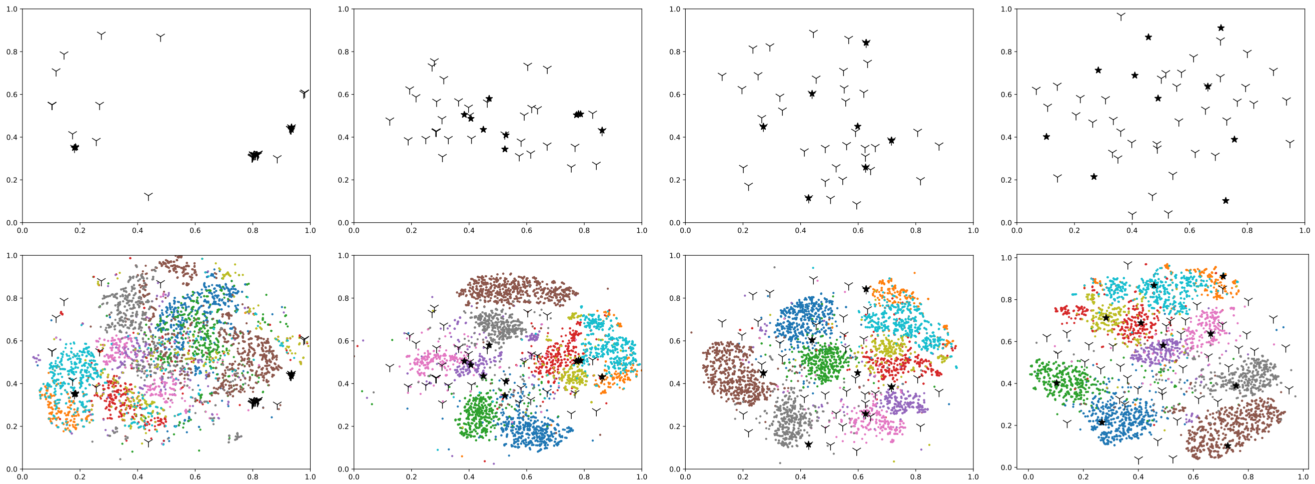
ventional cross-entropy loss (Ours-trans (XE)). This shows the effectiveness of using the generalized cross-entropy loss for avoiding the negative impact of incorrect pseudo labels. Comparative speaking, similar as we have observed in the inductive setting, our method significantly outperforms existing methods, especially for unseen classes in GZSL.

4.5. Further Analyses

Analyzing self-training process. In the transductive ZSL setting, we propose to calibrate weight generator f towards unseen classes using test data in a novel self-training fashion. We alternate between generating pseudo labels for unseen images using f and updating f using the pseudo labels of high confidence. By this self-training strategy, the bias of f towards seen classes can be progressively eliminated, with boost for unseen class recognition as the consequence.

To analyze how this self-training process works, we plot in Figure 2 the changes of training loss, classification accuracy, number of confident unseen samples (used for updating the model) and the portion of the correctly labeled ones among them. We can see that as the training round increases, the training loss keeps decreasing and the collection of confident samples is consistently enlarged. At the same time, the accuracy of pseudo label assignment is also promoted. This means with the increase of training round, the unlabeled images used for training are boosted in terms of both quantity and quality, which in return further improves the classifier generator.

Number of classes per episode. Table 5 shows that ZSL accuracy changes little w.r.t. sampled classes in each mini-batch, which contradicts the observations in [32], where episode-based training is used for few-shot learning. We speculate the reason is that sampling more classes per mini-batch in [32] helps boost discriminability of the feature extraction model. This does not apply to us as we use pre-trained features. Sampling more classes in each mini-batch can be approximated by sampling multiple mini-batches.



(a) V→A (b) DEM [43] (c) Ours (d) Ours-trans

Figure 3. t-SNE [24] visualization of visual feature embeddings and classifier weight vectors (or class prototypes) for the AwA1 dataset. **Top**: classifier weight vectors (or class prototypes) of both seen ("Y") and unseen (⋆) classes. **Bottom**: classifier weight vectors and visual feature embeddings for unseen classes. Different colors represent different classes. "V→A" represents projecting visual embeddings to attribute space.

Embedding visualization. Recall that we calculate the possibility of an image $\mathbf{x}$ of belonging to class y given class attribute $\mathbf{a}_y$ by calculating the Cosine similarity of $\mathbf{x}$ and the classifier weight $\mathbf{w}_y$ generating from $\mathbf{a}_y$ (Eq. (2)). As Cosine similarity of two vectors is equivalent to their dot product after being normalized, we can view $\frac{\mathbf{w}_y}{\|\mathbf{w}_y\|}$ as the prototype of class y. By this interpretation, the possibility of $\mathbf{x}$ of belonging to class y can be measured by the distance of the normalized feature $\frac{\mathbf{x}}{\|\mathbf{x}\|}$ and the normalized classifier weight vector $\frac{\mathbf{w}_y}{\|\mathbf{w}_y\|}$. Thus, we can visualize normalized classifier weight vectors and normalized visual feture vectors to qualitatively evaluate the discriminability of the classifiers.

We plot the t-SNE visualizations [24] of the classifier weights and their overlappings with the visual features of unseen classes in Figure 3. We can see that our class prototypes are more spatially dispersed than that of DEM [43] which does not consider the inter-class separation information for generating class prototypes. Besides, we can observe that by projecting visual features to attribute space, the corresponding class prototypes are extremely clustered. This substantiates the merits of formulating ZSL as a conditional visual classification problem, by which we can naturally benefit from the high discrimination of the visual features and the inter-class separation information to get discriminative classifiers for both seen and unseen classes. Moreover, we can also see that the distribution of the class prototypes in the transductive setting is even more dispersed than that for the inductive setting. This evidences the effectiveness of our transductive ZSL algorithm in exploiting unlabeled test data for enhancing the discriminability of the classifiers for both seen and unseen classes.

By overlapping the class prototypes with visual features of unseen classes, we can observe that visual features of unseen classes lie closely with their corresponding class prototypes, whiling being far away from those of seen classes. In contrast, this favorable distribution cannot be observed in the plots of DEM and the algorithm which projects visual features to the attribute space. This further substantiates the superiority of our method.

5. Conclusions

In this paper, we reformuate ZSL as a visual feature classification problem conditioned on the attributes. Under this reformulation, we develop algorithms for various ZSL settings. For the conventional setting, we propose to learn a deep neural network to generate visual feature classifiers directly from the attributes, and guide the process with a cosine similarity based cross-entropy loss and an episode-based training scheme. For the generalized setting, we propose to concatenate classifiers for both seen and unseen classes to recognize objects from all classes. For the transductive setting, we develop a novel learning-without-forgetting self-training mechanism to calibrate the classifier genereator towards unseen classes while maintaining good performance for seen classes. Experiments on widely used datasets verify the effectiveness of the proposed methods and demonstrate that the proposed methods obtain remarkable advantages over the state-of-the-art methods, especially for unseen classes in the generalized ZSL setting.

Acknowledgement: This research is supported in part by the NSF IIS award 1651902, U.S. Army Research Office Award W911NF-17-1-0367, and NEC labs America.

References

[1] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, 2015. 1, 2, 7

[2] Yashas Annadani and Soma Biswas. Preserving semantic relations for zero-shot learning. In *CVPR*, 2018. 2, 7

[3] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *CVPR*, 2016. 7

[4] Soravit Changpinyo, Wei-Lun Chao, and Fei Sha. Predicting visual exemplars of unseen classes for zero-shot learning. In *ICCV*, 2017. 2

[5] Long Chen, Hanwang Zhang, Jun Xiao, Wei Liu, and Shih-Fu Chang. Zero-shot visual recognition using semantics-preserving adversarial embedding network. In *CVPR*, 2018. 2, 7

[6] Mohamed Elhoseiny, Yizhe Zhu, Han Zhang, and Ahmed Elgammal. Link the head to the "beak": Zero shot learning from noisy text description at part precision. In *CVPR*, 2017. 2

[7] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *CVPR*, 2009. 5

[8] Rafael Felix, BG Vijay Kumar, Ian Reid, and Gustavo Carneiro. Multi-modal cycle-consistent generalized zero-shot learning. In *ECCV*, 2018. 2

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*, 2017. 3

[10] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Tomas Mikolov, et al. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013. 2, 7

[11] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018. 3

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[13] Elyor Kodirov, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Unsupervised domain adaptation for zero-shot learning. In *ICCV*, 2015. 1, 2

[14] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. In *CVPR*, 2017. 7

[15] 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. 2, 5

[16] Jimmy Lei Ba, Kevin Swersky, Sanja Fidler, et al. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *CVPR*, 2015. 2

[17] Kai Li, Zhengming Ding, Kunpeng Li, Yulun Zhang, and Yun Fu. Support neighbor loss for person re-identification. In *ACM MM*, 2018. 1

[18] Kai Li, Martin Renqiang Min, Bing Bai, Yun Fu, and Hans Peter Graf. On novel object recognition: A unified framework for discriminability and adaptability. In *CIKM*, 2019. 3

[19] Kunpeng Li, Ziyan Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. Guided attention inference network. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019. 1

[20] Kunpeng Li, Yulun Zhang, Kai Li, Yuanyuan Li, and Yun Fu. Attention bridging network for knowledge transfer. In *ICCV*, 2019. 1

[21] Kunpeng Li, Yulun Zhang, Kai Li, Yuanyuan Li, and Yun Fu. Visual semantic reasoning for image-text matching. In *ICCV*, 2019. 1

[22] Yan Li, Junge Zhang, Jianguo Zhang, and Kaiqi Huang. Discriminative learning of latent features for zero-shot recognition. In *CVPR*, 2018. 1

[23] Chunjie Luo, Jianfeng Zhan, Lei Wang, and Qiang Yang. Cosine normalization: Using cosine similarity instead of dot product in neural networks. *arXiv preprint arXiv:1702.05870*, 2017. 3, 6

[24] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008. 8

[25] Ashish Mishra, Shiva Krishna Reddy, Anurag Mittal, and Hema A Murthy. A generative model for zero shot learning using conditional variational autoencoders. In *CVPR Workshops*, 2018. 2

[26] Pedro Morgado and Nuno Vasconcelos. Semantically consistent regularization for zero-shot recognition. In *CVPR*, 2017. 2

[27] 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. In *ICLR*, 2014. 7

[28] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*, 2012. 5

[29] Marcus Rohrbach, Sandra Ebert, and Bernt Schiele. Transfer learning in a transductive setting. In *NIPS*, 2013. 2

[30] Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *ICML*, 2015. 2, 7

[31] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Rogerio Feris, Abhishek Kumar, Raja Giryes, and Alex M Bronstein. Delta-encoder: an effective sample synthesis method for few-shot object recognition. *arXiv preprint arXiv:1806.04734*, 2018. 1

[32] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NIPS*, 2017. 7

[33] Jie Song, Chengchao Shen, Jie Lei, An-Xiang Zeng, Kairi Ou, Dacheng Tao, and Mingli Song. Selective zero-shot classification with augmented attributes. In *ECCV*, 2018. 1

[34] Jie Song, Chengchao Shen, Yezhou Yang, Yang Liu, and Mingli Song. Transductive unbiased embedding for zero-shot learning. In *CVPR*, 2018. 2, 7

[35] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, 2018. 2, 7

[36] Vinay Kumar Verma and Piyush Rai. A simple exponential family framework for zero-shot learning. In *ECML-PKDD*, 2017. 7

[37] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NIPS*, 2016. 3

[38] Xiaolong Wang, Yufei Ye, and Abhinav Gupta. Zero-shot recognition via semantic embeddings and knowledge graphs. In *CVPR*, 2018. 2, 3

[39] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010. 5

[40] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *CVPR*, 2016. 1, 7

[41] 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. 1, 5

[42] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *CVPR*, 2018. 2, 3, 7

[43] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *CVPR*, 2017. 1, 2, 7, 8

[44] Yulun Zhang, Chen Fang, Yilin Wang, Zhaowen Wang, Zhe Lin, Yun Fu, and Jimei Yang. Multimodal style transfer via graph cuts. In *ICCV*, 2019. 1

[45] Yulun Zhang, Kunpeng Li, Kai Li, Bineng Zhong, and Yun Fu. Residual non-local attention networks for image restoration. In *ICLR*, 2019. 1

[46] Zhilu Zhang and Mert R Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. *arXiv preprint arXiv:1805.07836*, 2018. 4

[47] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via semantic similarity embedding. In *ICCV*, 2015. 2

[48] Yizhe Zhu, Mohamed Elhoseiny, Bingchen Liu, Xi Peng, and Ahmed Elgammal. A generative adversarial approach for zero-shot learning from noisy texts. In *CVPR*, 2018. 2

[49] Yizhe Zhu, Mohamed Elhoseiny, Bingchen Liu, Xi Peng, and Ahmed Elgammal. A generative adversarial approach for zero-shot learning from noisy texts. In *CVPR*, 2018. 3

[50] Yizhe Zhu, Jianwen Xie, Bingchen Liu, and Ahmed Elgammal. Learning feature-to-feature translator by alternating back-propagation for generative zero-shot learning. In *ICCV*, 2019. 2