Transductive Unbiased Embedding for Zero-Shot Learning

Jie Song[1], Chengchao Shen[1], Yezhou Yang[2], Yang Liu[3], and Mingli Song[1]

[1]College of Computer Science and Technology, Zhejiang University, Hangzhou, China
[2]Arizona State University, Tempe, USA
[3]Alibaba Group, Hangzhou, China

Abstract

Most existing Zero-Shot Learning (ZSL) methods have the strong bias problem, in which instances of unseen (target) classes tend to be categorized as one of the seen (source) classes. So they yield poor performance after being deployed in the generalized ZSL settings. In this paper, we propose a straightforward yet effective method named Quasi-Fully Supervised Learning (QFSL) to alleviate the bias problem. Our method follows the way of transductive learning, which assumes that both the labeled source images and unlabeled target images are available for training. In the semantic embedding space, the labeled source images are mapped to several fixed points specified by the source categories, and the unlabeled target images are forced to be mapped to other points specified by the target categories. Experiments conducted on AwA2, CUB and SUN datasets demonstrate that our method outperforms existing state-of-the-art approaches by a huge margin of $9.3 \sim 24.5\%$ *following generalized ZSL settings, and by a large margin of* $0.2 \sim 16.2\%$ *following conventional ZSL settings.*

1. Introduction

With the availability of large-scale training data, the field of visual object recognition has made significant progress in the last several years [17, 35, 37, 13, 14]. However, collecting and labeling training data are laboriously difficult and costly. For example, in fine-grained classification, expert knowledge is required to discriminate between different categories. For rare categories, such as endangered species, it's an extremely difficult work to collect sufficient and statistically diverse training images. Even worse, the frequencies of observing objects follow a long-tailed distribution [33, 45], which indicates that the number of such unfrequent objects significantly surpasses that of common objects. Given limited or zero training images, existing visual recognition models (*e.g.*, deep CNN models) struggle to make correct predictions.

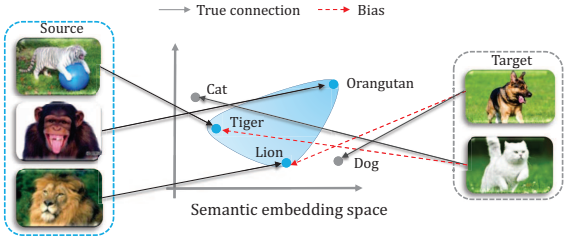

Figure 1. An illustrative diagram of the bias towards seen source classes in the semantic embedding space. The blue circles denote the anchor points specified by the source classes.

Zero-Shot Learning (ZSL) [8, 18, 1, 31, 2, 29, 41, 24] has emerged as a promising paradigm to alleviate the above problem. Unlike fully supervised classification which requires sufficient labeled training images for each category, ZSL distinguishes between two types of categories: *source* and *target*, where the labeled images are only available for the source categories. To facilitate the recognition of novel target categories, ZSL assumes the source and the target categories share a common semantic space to which both the images and class names can be projected. The semantic space can be defined by attributes [8, 1], word2vec [21] or WordNet [23]. Under this assumption, the recognition of images from novel target categories can be achieved by the nearest neighbor search in the shared space.

Depending on whether the unlabeled data of target classes are available for training, existing ZSL methods can be categorized into two schools: **inductive ZSL** [9, 25, 43, 2, 31, 4] and **transductive ZSL** [16, 10, 12]. For the inductive ZSL, only data of the source categories are available during the training phase. For the transductive ZSL methods, both the labeled source data and the unlabeled target data are available for training. The transductive ZSL aims to utilize the information from both the labeled source data and the unlabeled target data to accomplish the ZSL task.

During the test phase, most existing inductive and transductive ZSL methods [18, 1, 31, 2, 29, 24] assume the test images come solely from the target classes. Therefore, the search space for classifying the new test images is restricted to the target classes. We call this experimental settings **conventional settings**. However, in a more practical situation, the test images come not only from the target but also from the source classes. Hence, both the source and the target classes should be considered. This experimental settings are usually regarded as the *generalized* ZSL settings [41, 6], abbreviated to **generalized settings** in this paper.

Existing ZSL methods perform much worse in the generalized settings than in the conventional settings [41, 6]. One vital factor accounting for the poor performance can be explained as follows. ZSL achieves the recognition of new categories by establishing the connection between the visual embeddings and the semantic embeddings. However, during the phase of bridging the visual and the semantic embeddings, there exists a strong bias [6] (shown in Figure 1). During the training phase of most existing ZSL methods, the visual instances are usually projected to several fixed anchor points specified by the source classes in the semantic embedding space. This leads to a strong bias when these methods are used for testing: given images of novel classes in the target dataset, they tend to categorize them as one of the source classes.

To alleviate the mentioned problem above, we propose a novel transductive ZSL method in this paper. The proposed method assumes that both the labeled source and the unlabeled target data are available during the training phase. On the one hand, the labeled source data are used to learn the relationship between visual images and semantic embeddings. On the other hand, the unlabeled data of target classes are used to alleviate the strong bias towards source classes. More specifically, unlike other ZSL methods which always map input images to several fixed anchor points in the embedding space during training, our method allows the mapping from the inputs to other points, which significantly alleviates the strong bias problem.

We dub the proposed ZSL method as *Quasi-Fully Supervised Learning* (QFSL), as it works like the conventional fully supervised classification in which a multi-layer neural network and a classifier are integrated together (shown in Figure 2). The architecture of the multi-layer neural network is usually taken from AlexNet [17], GoogleNet [37] or other well-known deep networks. In the training phase, our model is trained in an end-to-end manner to recognize the data from both source and target classes even without labeled data for the target classes. This feature brings up a compelling advantage: when the labeled data of target classes are available in the future, it can be directly used to train our model. In the test phase, our trained model can be directly used to recognize new images from both the source and the target classes without any modifications.

To sum up, we made the following contributions: 1) A transductive learning (QFSL) method is proposed to learn unbiased embeddings for ZSL. To our knowledge, this is the first work to adopt transductive learning method in solving the ZSL problem in generalized settings. 2) Experiments reveal that our method significantly outperforms existing ZSL methods, in both generalized and conventional settings.

2. Related Work

Zero-Shot Learning ZSL relies on the semantic space to associate source and target classes. Various semantic spaces have been investigated, including attributes [8, 18, 1, 41, 24], word vector [9, 23], text description [29, 42] and human gaze [15]. The attribute has been shown to be an effective semantic space [2, 31, 24] for ZSL. However, its superior performance is obtained at the cost of much more expensive human labor. As an alternative, the word vectors are gaining more attention recently [22, 27] since they are learned from the large text corpus in an unsupervised way. Albeit their popularity, the word vectors often suffer from visual-semantic discrepancy problem [28, 5, 7]. In addition to the word vectors, human gaze [15] is recently proposed to replace the attributes, as its annotation can be performed by non-experts without domain knowledge.

In terms of the way how the visual space and the semantic space are related, existing ZSL methods can be mainly categorized into three groups: (1) from the visual space to the semantic space [9, 2, 29], (2) from the semantic space to the visual space [42, 34, 16] and (3) both the visual space and the semantic space are projected to a shared intermediate space [20, 44, 4]. As long as one of the above pathways is established, classification can be carried out via the nearest neighbor search in the embedding space which both the original visual inputs and the class labels can access. However, most existing ZSL methods share a common deficiency. During the training phase, regardless of how these two spaces are related, the existing ZSL usually project the visual inputs to several fixed points in the embedding space. It leads to the bias problem as discussed in Section 1. Our work aims at alleviating this problem to improve the performance of ZSL.

Transductive Zero-Shot Learning Transductive ZSL solves ZSL in a semi-supervised learning manner where both the labeled source data and the unlabeled target data are available. Propagated Semantic Transfer (PST) [30] exploits the manifold structure of novel classes to conduct label propagation. Transductive Multi-View ZSL (TMV) [10] and Unsupervised Domain Adaption (UDA) [16] associate cross-domain data by CCA and regularized sparse coding. In [12], a joint learning approach is proposed to learn the Shared Model Space (SMS) for transductive ZSL settings.

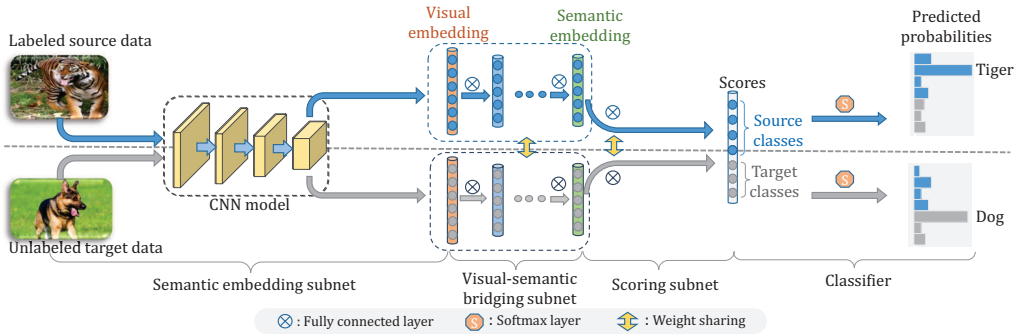

Figure 2. An overall architecture of the proposed QFSL model. Both the labeled and the unlabeled data are used to train the same model. Here for a better understanding, we depict them in two streams.

With the SMS, knowledge can be effectively transferred between classes using attributes. In this paper, we leverage both the labeled source data and the unlabeled target data to learn an unbiased embedding space for ZSL.

Zero-Shot Learning in Generalized Settings In performance evaluation, most existing ZSL methods usually assume that the test instances belong only to the unseen target classes. However, in practice, we are more often required to recognize instances from both the source and the target classes. The generalized settings relax the unrealistic assumption of the conventional settings with both the seen classes and the unseen classes at test time. In [9, 25], the source classes are considered when the classification is conducted, but only data from the unseen classes are tested. In [36], a two-stage approach is proposed to solve the ZSL problem in generalized settings. Before classification, it first determines whether a test instance is from a source or target class. In [6], an empirical study and analysis of ZSL in generalized settings are provided. Recently, [41] shows many ZSL methods behave much worse in the generalized settings than in the conventional settings.

3. Quasi-Fully Supervised Learning

3.1. Problem Formulation

Assume that there is a source dataset $\mathcal{D}^s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ consisting of N_s images. Each image x_i^s is associated with a corresponding label y_i^s, $y_i^s \in \mathcal{Y}^s = \{y_i\}_{i=1}^{S}$, and S is the number of the source classes. Similarly, there is a target dataset $\mathcal{D}^t = \{(x_i^t, y_i^t)\}_{i=1}^{N_t}$ consisting of N_t images. Each image x_i^t is associated with a corresponding label y_i^t, $y_i^t \in \mathcal{Y}^t = \{y_{S+i}\}_{i=1}^{T}$, and T is the number of the target classes. $\mathcal{Y}^s \cup \mathcal{Y}^t = \mathcal{Y}$, $\mathcal{Y}^s \cap \mathcal{Y}^t = \emptyset$. The goal of ZSL in conventional settings is to learn a prediction function f as below from the source data

$$f(x; W) = \arg\max_{y \in \mathcal{Y}} F(x, y; W), \tag{1}$$

so that its performance on the target data is maximized. F is a score function, which ranks the correct label higher than the incorrect labels, and W is the parameters of F. F usually takes the following bilinear form [1, 9, 2]:

$$F(x, y; W) = \theta(x)^T W \phi(y), \tag{2}$$

where $\theta(x)$ and $\phi(y)$ are the visual and the semantic embeddings, respectively. The score function is usually optimized by minimizing the regularized loss:

$$L = \frac{1}{N_s} \sum_{i=1}^{N_s} L_p(y_i, f(x_i; W)) + \gamma \Omega(W), \tag{3}$$

where L_p is the classification loss (such as entropy loss and structured SVM [38]) to learn the mapping between the visual and the semantic embeddings. Ω is the regularization term used to constrain the complexity of the model.

In this paper, we assume the labeled source data $\mathcal{D}^s$, the unlabeled target data $\mathcal{D}_u^t = \{x_i^t\}_{i=1}^{N_t}$, and the semantic embeddings ϕ are available for training in our approach. The aim of our method is to achieve good performance in not only the conventional but also the generalized settings.

3.2. The QFSL Model

Different from the bilinear form described above, the scoring function F in our method is designed as a nonlinear one. The whole model is implemented by a deep neural network (shown in Figure 2). It consists of four modules: the visual embedding subnet, the visual-semantic bridging subnet, the scoring subnet, and the classifier. The visual embedding subnet maps the raw images into visual embedding space. The visual-semantic bridging subnet projects the visual embeddings to semantic embeddings. The scoring subnet produces scores of every class in the semantic embedding space. And the classifier makes the final predictions based on the scores. All modules are differentiable and implemented by widely used layers including the convolutional layer, the fully connected layer, the ReLU [17] layer and the softmax layer. Hence, our model can be trained in

an end-to-end manner. Now we describe each module in detail in the following sections.

3.2.1 Visual Embedding Subnet

Most existing ZSL models [11, 2, 3, 31, 43, 19] adopt deep CNN features for visual embeddings. The visual embedding function θ is fixed in these methods. So they do not fully exploit the power of deep CNN models. Here, we also adopt a pre-trained CNN model to perform visual embedding. The major difference is that our visual embedding function can be optimized together with other modules[1]. The parameters of the visual embedding subnet are denoted by W_θ. Unless otherwise specified, we use the output of the first fully connected layer as the visual embeddings.

3.2.2 Visual-Semantic Bridging Subnet

It is vital to build the connections between the image and the semantic embeddings. The connection can be built by either a linear [1, 9, 2] or a nonlinear [40, 36] function. In this paper, we adopt a non-linear function φ to project the visual embeddings to the semantic embeddings. φ is implemented by several fully connected layers, each of which is followed by a ReLU non-linear activation layer. The design of bridging function depends on the CNN architecture from the visual embedding subnet. Specifically, our design follows the fully connected layers of the selected CNN model. The visual-semantic bridging subnet is optimized together with the visual embedding subnet. The parameters of the visual-semantic bridging subnet are denoted by W_φ.

3.2.3 Scoring Subnet

After bridging the visual and the semantic embeddings, recognition task can be carried out by the nearest neighbor search in the semantic embedding space. Given an image, we firstly obtain its visual embedding by the visual embedding subnet. Then the visual embedding is mapped to the semantic embedding by the visual-semantic bridging subnet. Finally, we use the inner product between the projected embedding and the normalized semantic embeddings as the scores. Therefore, the score function is

$$F(x, y; W) = \varphi(\theta(x; W_\theta); W_\varphi)\phi^*(y) \tag{4}$$

where W_θ and W_φ are the weights of the visual embedding function and the visual-semantic bridging function respectively, and $\phi^*(y)$ is the normalized semantic embedding of y: $\phi^*(y) = \frac{\phi(y)}{\|\phi(y)\|_2}$.

[1]In some situations, keeping the visual embedding subnet fixed produces better performance. We conduct further discussions in Section 4.2.1.

The scoring subnet is implemented as a single fully connected layer. The weights are initialized with the normalized semantic vectors of both the source and the target classes: $[\phi^*(y_1), \phi^*(y_2), ..., \phi^*(y_{S+T})]$. Unlike the visual embedding subnet and the visual-semantic bridging subnet, the weights of the scoring subnet are frozen and will not be updated during the training phase. In this way, for a labeled source image (x_i^s, y_i^s), our model is trained to project the image x_i^s to an embedding which has the most similar direction with the semantic embedding $\phi(y_i^s)$.

Note that though we don't have the labeled data of target classes, the target classes will also be involved in the training in our approach. Hence during the training phase, our method produces $S + T$ scores for a given image.

3.2.4 Classifier

After the scoring subnet, we apply a traditional $(S + T)$-way softmax classifier to produce the predicted probability vector for all the classes. The predicted class of the input image is just the one with the highest probability.

3.3. Optimization of the QFSL Model

As described above, the architecture of our method is like the conventional fully supervised classification model with a $(S + T)$-way classifier for both the target and the source classes. Unfortunately, only the data for source classes are labeled while the data from target classes is unlabeled. In order to train the proposed model, we define a *Quasi-Fully Supervised Learning (QFSL)* loss:

$$L = \frac{1}{N_s}\sum_{i=1}^{N_s} L_p(x_i^s) + \frac{1}{N_t}\sum_{i=1}^{N_t} \lambda L_b(x_i^t) + \gamma\Omega(W). \tag{5}$$

It is known that the loss of conventional fully supervised classification is usually composed by the classification loss L_p and regulation loss Ω. Different from such conventional definition, our proposed QFSL incorporates an additional bias loss L_b to alleviate the bias towards source classes:

$$L_b(x_i^t) = -\ln \sum_{i \in \mathcal{Y}^t} p_i, \tag{6}$$

where p_i is the predicted probability of class i. Given unlabeled instances from the target classes, this loss encourages our model to increase the sum of probabilities of being any target class. And consequently the model will prevent the instances of target classes from being mapped to the source classes.

For the classification loss L_p, we adopt the entropy loss in our method. For the regularization loss Ω, ℓ^2-norm is used for all the trainable parameters $W = \{W_\theta, W_\varphi\}$. λ and γ are trade-off weights among different losses, and they are set via cross-validation.

During the training phase, all the labeled and unlabeled data are mixed for training. Our model is optimized by the stochastic gradient descent algorithm. Each batch of training images is randomly drawn from the mixed dataset. Although our method is straightforward without bells and whistles, experiments show that it not only significantly alleviates the bias problem but also facilitates the building of connections between visual and semantic embeddings.

4. Experiments

In this section, extensive experiments are carried out to evaluate the performance of the proposed QFSL method. Firstly, we introduce some basic experimental settings. Then we discuss two implementation details of our method. Finally, we compare our proposed QFSL with existing state-of-the-art ZSL methods, in both the conventional and the generalized settings.

4.1. Experimental Settings

Datasets Three datasets are considered: Animals with Attributes 2 (AwA2) [41], Caltech-UCSD Birds-200-2011 (CUB) [39] and SUN Attribute Database (SUN) [26]. AwA2 is a coarse-grained dataset. It contains 37,322 images of 50 animals classes, in which 40 classes are used for training and the rest 10 classes for testing. For each class, there are about 750 labeled images. CUB is a fine-grained dataset containing 11,788 images of 200 bird species. We use 150 classes for training and the rest 50 for testing. In this dataset, each class has about 60 labeled images. SUN is another fine-grained dataset. There are 14,340 images coming from 717 types of scenes, of which 645 types are used for training, and the rest 72 for testing. Note that there are only about 20 images for every class on SUN, which is relatively scarce. In our experiments, we adopt either the standard train/test splits (SS) or the splits proposed (PS) in [41] in some experiments for fair comparisons.

Class-level attributes are used in our experiments. For AwA2, we use the provided continuous 85-dimension class-level attributes [41]. For CUB, continuous 312-dimension class-level attributes are provided in [39]. For SUN, there are continuous 102-dimension attributes provided in [26].

Model Selection and Training Four popularly used deep CNN models are involved in our following experiments: AlexNet [17], GoogLeNet [37], VGG19 [35] and ResNet101 [41]. They are all pre-trained on ImageNet [32] with 1K classes. Among these models, GoogLeNet is one of the most popular models used in the ZSL field, so we adopt GoogLeNet when we make comparisons between our and existing methods.

Unless otherwise specified, the learning rate is fixed to be 0.001, and the minibatch size is 64. The scaling weights of bias loss (λ) and weights decay (γ) are 1 and 0.0005, respectively. The training process stops after 5,000 iterations. These hyper-parameters are selected based on class-wise cross validation [43, 4, 6].

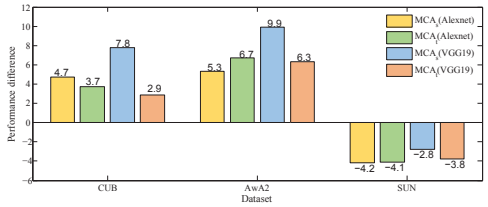

Figure 3. Comparisons between optimizing the visual embedding subnet and keeping it fixed. Performance difference = MCA(unfixed) $-$ MCA(fixed).

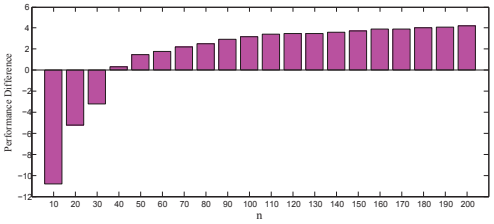

Figure 4. Performance difference with different number of training images per class on AwA2.

Evaluation Metrics To compare the performances, we adopt the Mean Class Accuracy (MCA) as the evaluation metric in our experiments:

$$MCA = \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} acc_y, \tag{7}$$

where acc_y is the top-1 accuracy on the test data from class y. In the conventional settings, MCA on only the target test data (MCA_t) is considered ($\mathcal{Y} = \mathcal{Y}_t$ in Eqn. 7). In the generalized settings, the search space at evaluation time is not restricted to the target classes, instead the the source classes are also included. Meanwhile, the test instances come from not only the target dataset, but also the source dataset ($\mathcal{Y} = \mathcal{Y}_s + \mathcal{Y}_t$ in Eqn. 7). Therefore, we adopt MCA_t, MCA_s (MCA on the source test data) and their harmonic mean (H) as the evaluation metrics:

$$H = \frac{2 * MCA_s * MCA_t}{MCA_s + MCA_t}. \tag{8}$$

4.2. Implementation Discussions

4.2.1 Optimization of the Visual Embedding Subnet

Many existing ZSL methods adopt pre-trained deep ConvNets as the visual embedding function. Most of them keep the trained CNN models fixed and do not optimize them during the training phase. In contrast, in our method, the visual embedding subnet can be optimized together with other parts. In this experiment, we compare the performance

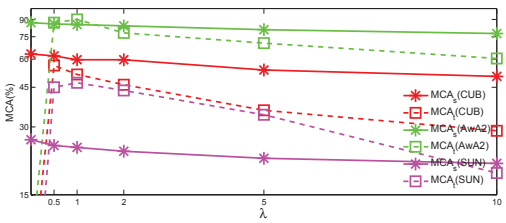

Figure 5. Performance of QFSL with varying λ.

of our method between with and without the visual embedding subnet fixed. All four models (AlexNet, GoogLeNet, VGG19, and ResNet101) are adopted to implement our methods. Experiments in the generalized settings are conducted on CUB, AwA2 and SUN datasets. The results of AlexNet and VGG19 are shown in Figure 3 (GoogLeNet and ResNet101 produce similar results). It can be seen that with the visual embedding subnet optimized, QFSL achieves much better performance on CUB and AwA2 than that with visual embedding function fixed. However, on the SUN dataset, training the visual embedding subnet produces a worse performance. We speculate that the scarce training data for source classes account for that. On AwA2 and CUB, there are about 750 and 60 training images for each category, respectively. However, there are only 20 images for each category on SUN. To validate our speculation, we conduct another experiment on the AwA2 dataset, as there are much more images per class in this dataset. In this experiment, our model is trained with different numbers (denoted by n) of labeled source images per class. Results are depicted in Figure 4. It can be concluded that with fewer training images per class, training the visual embedding subnet indeed leads to worse performance, which verifies our speculation.

4.2.2 Classification Loss and Bias Loss

As aforementioned in Section 3.3, there are three components in our loss function: the classification loss, the bias loss, and the regularization loss. The classification loss is used to build the connection between the visual embeddings and the semantic embeddings, and the bias loss is designed to alleviate the bias towards source classes. In this section, we explore how the trade-off between the classification loss and the bias loss impacts the performance of QFSL in the generalized settings.

We test QFSL with several different λ values $\{0.0, 0.5, 1.0, 2.0, 5.0, 10.0\}$ on all the three datasets. In the experiment, we adopt the AlexNet as the visual embedding function. Figure 5 shows the results of QFSL with different λ. Consistently, on all the three datasets, MCA_s decreases steadily as we increase λ. It is reasonable because putting more attention to alleviating the bias will distract the model from building the connection between image and semantic

Table 1. Comparisons in conventional settings (in %). For each dataset, the best result is marked in **bold** font and the second best in blue. We report results averaged over 5 random trails.

	Method	CUB SS	CUB PS	SUN SS	SUN PS	AwA2 SS	AwA2 PS
§	DAP [19]	37.5	40.0	38.9	39.9	58.7	46.1
§	CONSE [25]	36.7	34.3	44.2	38.8	67.9	44.5
§	SSE [43]	43.7	43.9	25.4	54.5	67.5	61.0
§	ALE [1]	53.2	54.9	59.1	58.1	80.3	62.5
§	DEVISE [9]	53.2	52.0	57.5	56.5	68.6	59.7
§	SJE [2]	55.3	53.9	57.1	53.7	69.5	61.9
§	ESZSL [31]	55.1	53.9	57.3	54.5	75.6	58.6
§	SYNC [4]	54.1	55.6	59.1	56.3	71.2	46.6
£	UDA [16]	39.5	–	–	–	–	–
£	TMV [10]	51.2	–	61.4	–	–	–
£	SMS [12]	59.2	–	60.5	–	–	–
	QFSL$^-$	58.5	58.8	58.9	56.2	72.6	63.5
	QFSL	**69.7** ↑10.5	**72.1** ↑13.3	**61.7** ↑0.3	**58.3** ↑0.2	**84.8** ↑4.5	**79.7** ↑16.2

§ : inductive ZSL methods.
£ : transductive ZSL methods.
↑ : performance boost compared with the best existing ZSL methods (including the baseline QFSL$^-$).

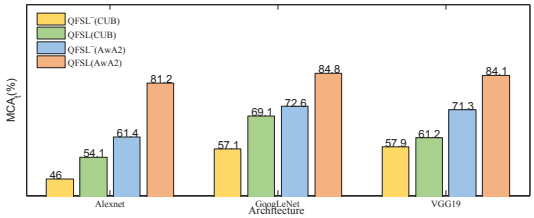

Figure 6. Comparisons between QFSL$^-$ and QFSL on different CNN architectures.

embeddings. For MCA_t, the overall best results are obtained when $\lambda \in [0.5, 2]$. Smaller λ (< 0.5) leaves the bias problem unsolved. On the other side, larger λ (> 2) yields negative effects on the building of the relationship between image and semantic embeddings, thus decreasing MCA_t in return.

4.3. Comparisons in Conventional Settings

We firstly compare our method with existing state-of-the-art ZSL methods in the conventional settings. The compared methods include: 1) **inductive methods** DAP [19], CONSE [25], SSE [43], ALE [1], DEVISE [9], SJE [2], ESZSL [31], SYNC [4], and 2) **transductive methods** UDA [16], TMV [10] and SMS [12]. In addition to these existing ZSL methods, there exists a latent baseline: training our proposed model with only labeled source data, *i.e.*, the inductive version of our model. In this case, QFSL loss degrades to conventional fully supervised classification loss. We denote this baseline by QFSL$^-$ and also compare our method with it.

Experiments are conducted on AwA2, CUB, and SUN. We use both the standard split (SS) and the proposed split (PS) [41] for more convincing results. The visual embedding subnet is optimized for AwA2 and CUB, but fixed for SUN. Table 1 shows the experimental results. It can be seen

Table 2. Comparisons in the generalized settings (in %). Previously published results are given in normal font, and results of our implementations are given in *italics* font. For *QFSL*^{G} and *QFSL*^{R}, the visual embedding function is implemented with GoogLeNet and ResNet101, respectively. For each dataset, the best result is marked in **bold** font and the second best in blue. We report results averaged over 5 random trails (CMT*: CMT with novelty detection).

		AwA2			CUB			SUN		
	Method	**MCA$_s$**	**MCA$_t$**	**H**	**MCA$_s$**	**MCA$_t$**	**H**	**MCA$_s$**	**MCA$_t$**	**H**
†	DAP [19]	84.7	0.0	0.0	67.9	1.7	3.3	25.1	4.2	7.2
†	CONSE [25]	90.6	0.5	1.0	72.2	1.6	3.1	**39.9**	6.8	11.6
†	SSE [43]	82.5	8.1	14.8	46.9	8.5	14.4	36.4	2.1	4.0
†	ALE [1]	81.8	14.0	23.9	62.8	23.7	34.4	33.1	21.8	26.3
†	DEVISE [9]	74.7	17.1	27.8	53.0	23.8	32.8	30.5	14.7	19.8
†	SJE [2]	73.9	8.0	14.4	59.2	23.5	33.6	30.5	14.7	19.8
†	ESZSL [31]	77.8	5.9	11.0	63.8	12.6	21.0	27.9	11.0	15.8
†	SYNC [4]	90.5	10.0	18.0	70.9	11.5	19.8	43.3	7.9	13.4
†	CMT [36]	90.0	0.5	1.0	49.8	7.2	12.6	21.8	8.1	11.8
‡	CMT* [36]	89.0	8.7	15.9	60.1	4.7	8.7	28.0	8.7	13.3
‡	*CS* [6]	*77.6*	*45.3*	*57.2*	*49.4*	*48.1*	*48.7*	*22.0*	*44.9*	*29.5*
‡	*baseline*	*72.8*	*52.1*	*60.7*	*48.1*	*33.3*	*39.4*	*18.5*	*30.9*	*23.1*
‡	*QFSL*^{G}	*92.4*$^{\uparrow 1.8}$	*64.3*$^{\uparrow 12.2}$	*75.8*$^{\uparrow 15.1}$	*74.2*$^{\uparrow 2.0}$	***71.6***$^{\uparrow 23.5}$	*72.9*$^{\uparrow 24.2}$	*33.6*$^{\downarrow 6.3}$	***54.8***$^{\uparrow 9.9}$	***41.7***$^{\uparrow 12.2}$
‡	*QFSL*^{R}	***93.1***$^{\uparrow 2.5}$	***66.2***$^{\uparrow 14.1}$	***77.4***$^{\uparrow 16.7}$	***74.9***$^{\uparrow 2.7}$	*71.5*$^{\uparrow 23.4}$	***73.2***$^{\uparrow 24.5}$	*31.2*$^{\downarrow 8.7}$	*51.3*$^{\uparrow 6.4}$	*38.8*$^{\uparrow 9.3}$

† : ZSL methods which *do not* takes generalized settings into consideration.
‡ : ZSL methods which takes generalized settings into consideration.
↑ : Performance boost compared with the best existing ZSL methods (including the baseline).
↓ : Performance drop compared with the best existing ZSL methods (including the baseline).

that 1) the baseline of our method (QFSL$^-$) yields comparable performance with existing ZSL methods, and 2) the proposed method outperforms the baseline and existing approaches on all datasets. Notably, on CUB and AwA2, our method outperforms other state-of-the-art ZSL methods (including QFSL$^-$) by a large margin of $4.5 \sim 16.2\%$. The experimental results indicate that our approach effectively utilizes the valuable information contained in the unlabeled target data to facilitate the building of connections between the visual and the semantic embeddings.

To further verify that our method is not only effective to a specific CNN model, we implement our method with AlexNet, GoogleNet, and VGG respectively. In this experiment, as QFSL$^-$ is shown to achieve comparable performance with other ZSL methods in Table 1, we compare our method only with QFSL$^-$. The comparison result is provided in Figure 6. It can be noticed that our method outperforms the baseline consistently on all the three CNN models, which validates the effectiveness of our method.

4.4. Comparisons in Generalized Settings

Our method is designed to alleviate the strong bias problem. Therefore, we verify its effectiveness in the generalized settings, in which the strong bias problem often leads to poor performance. Before evaluating the performance of our method, there remains one issue to address. When evaluating the performance in the test phase, most of the existing transductive ZSL methods use the same target data used in the training phase. However, if our method adopts the same policy, it will be problematic because our method has already used the supervisory information that the unlabeled data are coming from the target classes. To solve this problem, we split the unlabeled target data into two halves and train two QFSL models. One half of the unlabeled data is used for training and the other one for testing when training our first model, and vice versa when training our second model. The final performance of our method is the average performance of these two models. To our knowledge, this is the first study on applying the transductive method to solve the ZSL problem in generalized settings.

We compare our method with several state-of-the-art ZSL methods [19, 25, 43, 1, 9, 2, 31, 4]. However, these methods do not take the generalized settings into consideration. In addition to these methods, we also compare our methods with two other ZSL methods *Calibrated Stacking (CS)* [6] and *Cross Model Transfer (CMT)* [36], which take the generalized settings into consideration. CS maximizes the performance in the generalized settings by trading off between MCA_s and MCA_t. CMT first utilizes novelty detection methods [36] to differentiate between source and target classes and then accordingly applies the corresponding classifiers. As our method utilizes the unlabeled target data, we introduce another baseline (called *baseline* here) for a clearer comparison. The *baseline* trains a deep binary classifier (GoogLeNet) on the available source data and unlabeled target data to discriminate between the source and the target data, then classifies the test instances in the corresponding search space.

The original data split and other experimental settings are kept the same as that used in [41], where the visual embedding function is implemented with ResNet101. For a fair comparison, we also adopt ResNet101 to implement the visual embedding function (denoted by QFSL^R). In addition, as GoogLeNet is widely used in ZSL, the performance of our method with GoogLeNet is also provided (denoted by QFSL^G). Experimental results are given in Table 2. It

can be seen that generally our method improves the overall performance (harmonic mean H) by an obvious margin ($9.3 \sim 24.5\%$ on the three datasets). The performance boost mainly comes from the improvement of mean class accuracy on the target classes (MCA_t), meanwhile without much performance drop on the source classes (MCA_s). These compelling results verify that our method can significantly alleviate the strong bias towards source classes by using the unlabeled instances from the target classes.

Another noticeable result from Table 2 is that the results of QFSL^R are generally better that of QFSL^G on CUB and AwA2 datasets. However, on SUN, QFSL^G achieves better performance. We observe the fact that only scarce (about 20) training images are available for each source category in the SUN dataset accounts for that. Using such scarce data to train deep CNN models like ResNet101 usually leads to over-fitted models.

5. Further Study and Discussions

In real-world scenarios, the number of the target classes usually greatly surpasses that of source ones. However, most datasets for ZSL benchmark violate that. For examples, for AwA2, only 10 of 50 classes are treated as target ones. On CUB, only 50 out of 150 classes are used as the target. More severely, on the SUN dataset, only 72 out of 717 classes are put into the target classes. In this section, we empirically study how the imbalance between the source and the target classes affects the proposed QFSL method.

Experiments are conducted on the SUN dataset, as there are much more classes in it. The visual embedding function is implemented with GoogLeNet. We adopt the standard split used by the most of other works. 72 classes are treated as the target categories. For the source categories, we randomly select seven subsets from the rest categories. The number of source categories is $\{100, 200, 300, 450, 550, 600, 645\}$. We use these 7 different source data and the fixed target data to test out method. For a better understanding of our method, we also depict the performance of the baseline QFSL$^-$, in which only the labeled source data are available.

Results in generalized settings are demonstrated in Figure 7. On the one hand, as the number of source classes increases, the classification task of source data becomes more difficult, which results in the performance drop in MCA_s. On the other hand, the increasing source classes provide more knowledge to build the mapping between the visual and the semantic embeddings, which results in the performance boost in terms of MCA_t.

Note that albeit with taking additional consideration of addressing the bias problem, our proposed method produces a comparable performance with the baseline QFSL$^-$ in MCA_s. Furthermore, with more imbalanced source and target classes, the new test instances from target classes are more likely to be classified into source classes (*i.e.*, the bias problem is more severe). Because our method alleviates the bias problem, it yields much better performance in this case. Consequently, as the number of source classes increases (*i.e.*, the imbalance ratio between source and target classes becomes larger), the superiority of our method over the baseline QFSL$^-$ becomes larger.

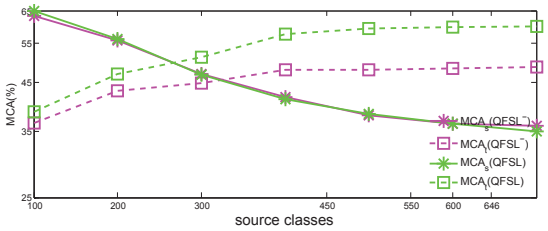

Figure 7. Performance of QFSL with different numbers of source classes on SUN.

6. Conclusions and Future Work

In this work, we have proposed a straightforward yet effective method to learn the unbiased embedding for ZSL. This method assumes both the labeled source data and the unlabeled target data are available at the training time. On the one hand, the labeled source data are projected to the points specified by the source classes in the semantic space, which builds the relationship between the visual embeddings and the semantic embeddings. On the other hand, the unlabeled target data are forced to be projected to other points specified by the target classes, which alleviates the bias towards source classes significantly. Various experiments conducted on different benchmarks demonstrate that our method outperforms other state-of-the-art ZSL methods by a large margin in both the conventional and the generalized settings.

There are many different research lines which are worthy of further study following this work. For example, in this work, semantically meaningful attributes are adopted as the semantic space. In our future work, we will exploit other semantic space such as word vectors. Another example is that this work addresses the bias problem by transductive learning, in our future work we will consider solving the same problem following the way of inductive learning.

Acknowledgments. This work was supported in part by the National Basic Research Program (973 Program, No. 2015CB352400), National Natural Science Foundation of China (61572428, U1509206), National Key Research and Development Program (2016YFB1200203), Fundamental Research Funds for the Central Universities (2017FZA5014), Alibaba-Zhejiang University Joint Institute of Frontier Technologies.

References

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

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

[3] M. Bucher, S. Herbin, and F. Jurie. Improving semantic embedding consistency by metric learning for zero-shot classiffication. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V*, pages 730–746, 2016.

[4] S. Changpinyo, W.-L. Chao, B. Gong, and F. Sha. Synthesized classifiers for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5327–5336, 2016.

[5] S. Changpinyo, W.-L. Chao, and F. Sha. Predicting visual exemplars of unseen classes for zero-shot learning. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017.

[6] W.-L. Chao, S. Changpinyo, B. Gong, and F. Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *European Conference on Computer Vision*, pages 52–68. Springer, 2016.

[7] B. Demirel, R. G. Cinbis, and N. I. Cinbis. Attributes2classname: A discriminative model for attribute-based unsupervised zero-shot learning. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017.

[8] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1778–1785. IEEE, 2009.

[9] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov, et al. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pages 2121–2129, 2013.

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

[11] Y. Fu and L. Sigal. Semi-supervised vocabulary-informed learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

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

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

[14] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017.

[15] N. Karessli, Z. Akata, B. Schiele, and A. Bulling. Gaze embeddings for zero-shot image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017.

[16] E. Kodirov, T. Xiang, Z. Fu, and S. Gong. Unsupervised domain adaptation for zero-shot learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2452–2460, 2015.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

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

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

[20] Y. Lu. Unsupervised learning on neural network outputs: with application in zero-shot learning. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 3432–3438. AAAI Press, 2016.

[21] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

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

[23] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[24] P. Morgado and N. Vasconcelos. Semantically consistent regularization for zero-shot recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017.

[25] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean. Zeroshot learning by convex combination of semantic embeddings. In *In Proceedings of ICLR*. Citeseer, 2014.

[26] G. Patterson and J. Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *Proceeding of the 25th Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[27] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[28] R. Qiao, L. Liu, C. Shen, and A. van den Hengel. Visually aligned word embeddings for improving zero-shot learning. In *British Machine Vision Conference (BMVC'17)*, 2017.

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

[30] M. Rohrbach, S. Ebert, and B. Schiele. Transfer learning in a transductive setting. In *Advances in neural information processing systems*, pages 46–54, 2013.

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

[32] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[33] R. Salakhutdinov, A. Torralba, and J. Tenenbaum. Learning to share visual appearance for multiclass object detection. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1481–1488. IEEE, 2011.

[34] Y. Shigeto, I. Suzuki, K. Hara, M. Shimbo, and Y. Matsumoto. Ridge regression, hubness, and zero-shot learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 135–151. Springer, 2015.

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

[36] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng. Zero-shot learning through cross-modal transfer. In *Advances in neural information processing systems*, pages 935–943, 2013.

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[38] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *J. Mach. Learn. Res.*, 6:1453–1484, Dec. 2005.

[39] 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.

[40] Y. Xian, Z. Akata, G. Sharma, Q. Nguyen, M. Hein, and B. Schiele. Latent embeddings for zero-shot classification. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 69–77, June 2016.

[41] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *arXiv preprint arXiv:1707.00600*, 2017.

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

[43] Z. Zhang and V. Saligrama. Zero-shot learning via semantic similarity embedding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4166–4174, 2015.

[44] Z. Zhang and V. Saligrama. Zero-shot learning via joint latent similarity embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6034–6042, 2016.

[45] X. Zhu, D. Anguelov, and D. Ramanan. Capturing long-tail distributions of object subcategories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 915–922, 2014.