Few-Shot Zero-Shot Learning: Knowledge Transfer with Less Supervision

Nanyi Fei[1], Jiechao Guan[1], Zhiwu Lu[2] (✉), and Yizhao Gao[2]

[1] School of Information, Renmin University of China, Beijing, China
[2] Beijing Key Laboratory of Big Data Management and Analysis Methods, Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China
luzhiwu@ruc.edu.cn

Abstract. Existing zero-shot learning (ZSL) methods assume that there exist sufficient training samples from seen classes, each annotated with semantic descriptors such as attributes, for knowledge transfer to unseen classes without any training samples. However, this assumption is often invalid because collecting sufficient seen class samples can be difficult and attribute annotation is expensive; it thus severely limits the scalability of ZSL. In this paper, we define a new setting termed Few-Shot Zero-Shot Learning (FSZSL), where only a few annotated images are collected from each seen class (i.e., few-shot). This is clearly more challenging yet more realistic than the conventional ZSL setting. To overcome the resultant image-level attribute sparsity, we propose a novel inductive ZSL model termed sparse attribute propagation (SAP) by propagating attribute annotations to more unannotated images using sparse coding. This is followed by learning bidirectional projections between features and attributes for ZSL. An efficient solver is provided for such knowledge transfer with less supervision, together with rigorous theoretic analysis. With our SAP, we show that a ZSL training dataset can also be augmented by the abundant web images returned by image search engine, to further improve the model performance. Extensive experiments show that the proposed model achieves state-of-the-art results.

1 Introduction

Due to the difficulty in collecting sufficient training images for large-scale object recognition [1–4] where deep convolutional neural networks (CNNs) are often employed, zero-shot learning (ZSL) has become topical in computer vision [5–13]. To recognize unseen classes without any training images, existing ZSL models leverage a semantic space as the bridge for knowledge transfer from seen classes to unseen ones, and the semantic attribute space is the most commonly used [14]. Given a set of seen class images, the visual features are first extracted, typically using CNNs pretrained on ImageNet. With the feature representations of images and the semantic representations of class names, the next task is to learn a joint embedding space using seen class data. In such a space, both feature and semantic representations are projected to be directly compared. Once the

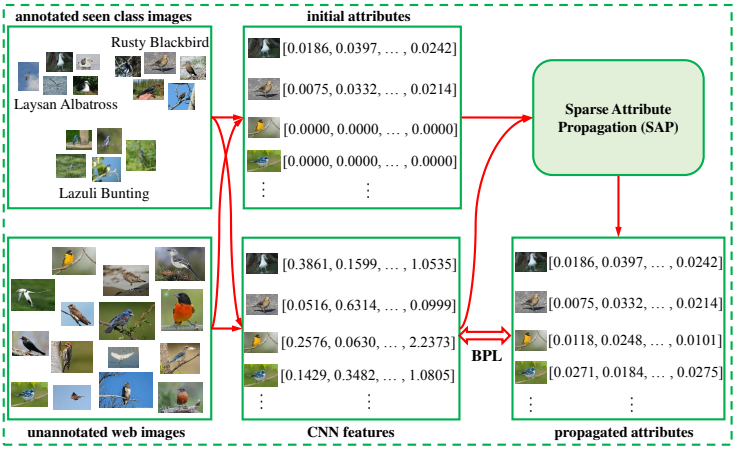


Fig. 1. Schematic illustration of the proposed ZSL model including SAP and BPL. The web images are obtained by Google with the query 'North American Bird'. The few annotated seen class images are augmented with these unannotated external data.

projection functions are learned, they are applied to test images and unseen class names, and the nearest neighbor class name is found for each test image.

Although ZSL can avoid the need of collecting unseen class images for training, it still requires a large number of attribute/label annotations per seen class: hundreds of class-level attribute annotations are often needed, along with hundreds of image-level class label annotations. This severely limits the scalability of ZSL. In this paper, to study how to overcome this limitation associated with existing ZSL models and make ZSL truly scalable, we define a new ZSL setting termed Few-Shot Zero-Shot Learning (FSZSL), where only a few annotated images are collected from each seen class. This is clearly more challenging yet *more realistic* than the conventional ZSL setting. Note that our new FSZSL setting is often encountered in real-world application scenarios such as fine-grained classification and medical image recognition. More specifically, in these scenarios, each image is hard to annotate with a class label even for an expert and thus only a few annotated images per seen class can be obtained; meanwhile, recognizing unseen classes is always needed because the new/rare classes will unavoidably occur when more data is accumulated.

To overcome the resultant image-level attribute sparsity, we propose a novel inductive ZSL model termed sparse attribute propagation (SAP) by propagating attribute annotations to more unannotated images using sparse coding [15, 16]. This is followed by learning bidirectional projections between features and attributes for ZSL. We formulate sparse attribute propagation (SAP) and bidirectional projection learning (BPL) within a unified ZSL framework: SAP aims to obtain more reliable attribute annotations, while BPL aims to learn more generalizable projections. We also give an efficient iterative solver, with rigorous theoretic algorithm analysis provided. Note that under the inductive ZSL

setting, only seen class images can be used for SAP. However, with SAP, our FSZSL becomes a semi-supervised learning problem. As a result, we are now able to exploit the abundant web images collected using image search engine to augment a ZSL dataset. These web images could even be used to replace the unannotated seen class images which are also exploited for training. In summary, we provide a flexible ZSL approach that can scale to real-world ZSL tasks. Our proposed ZSL model is illustrated in Fig. 1.

Our contributions are: (1) For the first time, we define a new setting termed FSZSL, which is more challenging yet more realistic than the conventional ZSL setting. (2) To overcome the attribute sparsity under our new setting, we propose a novel inductive ZSL model by integrating SAP and BPL into a unified framework. An efficient iterative solver is formulated, together with rigorous theoretic analysis. (3) Our model is highly flexible and can be generalized to other vision problems such as social image annotation (SIA) [17–19] (see the suppl. material). Extensive experiments show that our model achieves state-of-the-art results on both problems (i.e., ZSL and SIA).

2 Related Work

Knowledge Transfer for ZSL. Since both seen and unseen classes can be defined in a same semantic space, it is often leveraged as a bridge for knowledge transfer from seen classes to unseen ones. Existing ZSL methods typically learn a projection between the visual feature space and the semantic space, and can be divided into three groups depending on how the projection function is built: (1) The first group projects both visual and semantic spaces into a latent embedding space [20–23]. (2) Methods in the second group learn projections from the visual space to the semantic one [7, 6, 24]. (3) The third group projects semantic representations into the visual space [25, 26], which can reduce the hubness problem [27]. Moreover, several works [12, 28–31] first projects visual representations into the semantic space and then projects them back, which can help reduce the domain shift problem. Note that Semantic AutoEncoder (SAE) and our BPL are closely related. The main difference is that the weight hyperparameter for balancing the two projection directions is removed from our BPL. Moreover, our algorithm is given a theoretic analysis while such an analysis is missing for SAE. Notably, as we have stated our contributions above, BPL is not the focus of this work, and it can be replaced by any other embedding method (see Table 3).

ZSL with Less Human Annotation. A ZSL model typically exploits two types of human annotations for recognizing unseen classes without any training images: (1) the human-annotated class labels of training images from seen classes; (2) the human-defined semantic representations of seen/unseen classes. In the area of ZSL, much attention has been paid to reducing the annotation cost of generating human-defined semantic representations (e.g., the semantic space is formed using online textual documents [7, 8], human gaze [10], or visual similes [32, 11] instead of attributes), which leads to significantly less annotation cost. Different from these ZSL models, we focus on ZSL with less human annotation

by defining a new ZSL setting, i.e., only a few annotated images are collected from each seen class. Although our ZSL model is proposed based on attributes in this paper, it can be easily generalized to other forms of semantic space [7, 8, 10, 11] to further reduce the annotation cost. To our best knowledge, we are the first to define this new setting in the area of ZSL.

Semi-Supervised ZSL. In this paper, attribute propagation is performed from a few annotated seen class images to more unannotated images so that more reliable attribute annotations can be obtained. This can be regarded as a form of semi-supervised ZSL. Note that the test images from unseen classes are not used for training our model, i.e., we take an inductive ZSL setting. However, in the area of ZSL, when semi-supervised learning is applied to ZSL, the unlabelled test images from unseen classes are typically used for training. This results in a transductive ZSL setting: either label propagation [5, 6, 33, 9] or self-training [34–38, 28] is employed for semi-supervised learning. Since these transductive ZSL models assume the access to the whole test set, they have limited applications in real-world scenarios. Note that although the test set is not involved in the training process, our model still exploits the unannotated seen class images for attribute propagation. Given that it is not easy to obtain the unannotated seen class images, we choose to perform attribute propagation with unannotated external data from image search engine, which thus provides a feasible/convenient approach to applying our model to real-world ZSL tasks.

ZSL with Web Images. In computer vision, web images have been widely used to promote the performance of existing recognition models as in [39–42]. However, there is less attention on exploiting web images for ZSL. Two exceptions are: the web images are utilized to augment the unseen class data in [43] and discover event composition knowledge for zero-shot event detection in [44]. In this work, although web images are also employed as external data, our model is quite different from [43] in that we do not search web images *directly with unseen class names* since this is against the zero-shot setting.

3 Methodology

3.1 Problem Definition

Let $\mathcal{C}_s = \{cs_1, \cdots, cs_p\}$ denote a set of seen classes and $\mathcal{C}_u = \{cu_1, \cdots, cu_q\}$ denote a set of unseen classes, where p and q are the numbers of seen and unseen classes, respectively. These two sets of classes are disjoint. Similarly, $\mathbf{Z}_s = [\mathbf{z}_1^{(s)}, \cdots, \mathbf{z}_p^{(s)}] \in \mathbb{R}^{k \times p}$ and $\mathbf{Z}_u = [\mathbf{z}_1^{(u)}, \cdots, \mathbf{z}_q^{(u)}] \in \mathbb{R}^{k \times q}$ denote the corresponding seen and unseen class semantic representations (e.g., k-dimensional attribute vectors). We are given a set of seen class training images $\mathcal{D}_s = \{(\mathbf{x}_i^{(s)}, l_i^{(s)}), \mathbf{y}_i^{(s)} : i = 1, \cdots, r, r+1, \cdots, N_s\}$, where $\mathbf{x}_i^{(s)} \in \mathbb{R}^{d \times 1}$ is the d-dimensional feature vector of the i-th training image, $l_i^{(s)} \in \{1, \cdots, p\}$ is the label of $\mathbf{x}_i^{(s)}$ according to $\mathcal{C}_s$, $\mathbf{y}_i^{(s)} = \mathbf{z}_{l_i^{(s)}}^{(s)}$ is the semantic representation of $\mathbf{x}_i^{(s)}$ (i.e., only class-level attributes are needed), and N_s is the number of training images. In this paper, only the first

r annotated training images $\mathbf{x}_i^{(s)}$ $(1 \leq i \leq r)$ have non-zero attribute vectors, i.e., $\mathbf{y}_i^{(s)} = \mathbf{0}$ $(r+1 \leq i \leq N_s)$. Moreover, let $\mathcal{D}_u = \{(\mathbf{x}_i^{(u)}, l_i^{(u)}), \mathbf{y}_i^{(u)} : i = 1, \cdots, N_u\}$ denote a set of unseen class test images, where $\mathbf{x}_i^{(u)} \in \mathbb{R}^{d\times 1}$ is the feature vector of the i-th test image, $l_i^{(u)} \in \{1, \cdots, q\}$ is the unknown label of $\mathbf{x}_i^{(u)}$ according to $\mathcal{C}_u$, $\mathbf{y}_i^{(u)}$ denotes the unknown semantic representation of $\mathbf{x}_i^{(u)}$, and N_u is the number of test images. The goal of FSZSL is to predict the labels of test images by learning a classifier $f : \mathcal{X}_u \to \mathcal{C}_u$, where $\mathcal{X}_u = \{\mathbf{x}_i^{(u)} : i = 1, \cdots, N_u\}$. Under the generalized FSZSL setting (following [45–47]), the test samples can come from both seen and unseen classes, so the classifier becomes $f : \mathcal{X} \to \mathcal{C}_s \cup \mathcal{C}_u$, where $\mathcal{X}$ denotes the set of all test samples.

Note that the above problem definition is consistent with the inductive ZSL setting, where the unannotated seen class training images are given along with the annotated seen class training ones. Their uniform notations make the model formulation more concise. As we have mentioned, the unannotated seen class images can be replaced by the unannotated web images from image search engine (see Fig. 1), resulting in a feasible approach for real-world ZSL tasks. The details of such FSZSL with external data are given at the end of Section 3.

3.2 Model Formulation

When learned with only a few annotated images per seen class under our new ZSL setting, the projection function between the feature and semantic spaces is not reliable. Therefore, we choose to propagate such sparse attribute annotations to more unannotated images using sparse coding [15, 16]: more attribute annotations enable us to learn a more reliable projection, but the noise caused by attribute propagation should also be suppressed by sparse coding, which is thus called sparse attribute propagation (SAP). Moreover, given all seen class training images (with ground truth/predicted attribute vectors), we integrate the forward and reverse projections for ZSL, since either projection suffers from the projection domain shift [5, 25]. By bidirectional projection learning (BPL), a visual feature vector is first projected into a semantic space and then back into visual feature space to reconstruct itself. Such self-reconstruction can improve the generalization ability of the model and help tackle the projection domain shift. Our unified framework including SAP and BPL is given below.

Concretely, with the whole seen class training set $\mathcal{X}_s = \{\mathbf{x}_i^{(s)} : i = 1, \cdots, N_s\}$, we construct first a graph $\mathcal{G} = \{\mathcal{V}, \mathbf{A}\}$ with its vertex set $\mathcal{V} = \mathcal{X}_s$ and affinity matrix $\mathbf{A} = [a_{ij}]_{N_s\times N_s}$, where a_{ij} denotes the similarity between training images $\mathbf{x}_i^{(s)}$ and $\mathbf{x}_j^{(s)}$. The affinity matrix $\mathbf{A}$ can be defined as: $a_{ij} = \exp(-\|\mathbf{x}_i^{(s)} - \mathbf{x}_j^{(s)}\|_2^2/(2\sigma^2))$, where the parameter σ can be determined empirically ($\sigma = 1$ in this paper). The normalized Laplacian matrix $\mathbf{L}$ is given by

$$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}} , \tag{1}$$

where $\mathbf{I}$ is an identity matrix, and $\mathbf{D}$ is a diagonal matrix with its i-th diagonal element being $\sum_j a_{ij}$. We derive a new matrix $\mathbf{B} \in \mathbb{R}^{N_s\times N_s}$ from $\mathbf{L}$: $\mathbf{B} = \mathbf{\Sigma}^{\frac{1}{2}}\mathbf{V}^T$,

where $\mathbf{V}$ is an orthonormal matrix with each column being an eigenvector of $\mathbf{L}$, and $\mathbf{\Sigma}$ is a diagonal matrix with its diagonal element Σ_{ii} being an eigenvalue of $\mathbf{L}$ (sorted as $0 \leq \Sigma_{11} \leq \cdots \leq \Sigma_{N_s N_s}$). Denoting the eigen-decomposition of $\mathbf{L}$ as $\mathbf{L} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T$, $\mathbf{L}$ can be represented as: $\mathbf{L} = (\mathbf{\Sigma}^{\frac{1}{2}}\mathbf{V}^T)^T\mathbf{\Sigma}^{\frac{1}{2}}\mathbf{V}^T = \mathbf{B}^T\mathbf{B}$.

We further collect the feature and attribute vectors of the training set as $\mathbf{X}^{(s)} = [\mathbf{x}_1^{(s)}, \cdots, \mathbf{x}_{N_s}^{(s)}] \in \mathbb{R}^{d \times N_s}$ and $\mathbf{Y}^{(s)} = [\mathbf{y}_1^{(s)}, \cdots, \mathbf{y}_{N_s}^{(s)}] \in \mathbb{R}^{k \times N_s}$. Our ZSL model solves the following optimization problem:

$$\min_{\mathbf{Y},\tilde{\mathbf{Y}},\mathbf{W}} \{\|\mathbf{Y} - \tilde{\mathbf{Y}}\|_F^2 + \lambda_1\|\mathbf{B}\tilde{\mathbf{Y}}^T\|_1 + \lambda_2\|\mathbf{Y} - \mathbf{Y}^{(s)}\|_1 + \lambda_3(\|\mathbf{W}\mathbf{X}^{(s)} - \mathbf{Y}\|_F^2 + \|\mathbf{X}^{(s)} - \mathbf{W}^T\mathbf{Y}\|_F^2 + \lambda_4\|\mathbf{W}\|_F^2)\} , \quad (2)$$

where $\mathbf{W} \in \mathbb{R}^{k \times d}$ is a projection matrix from the visual feature space to the semantic space, $\mathbf{Y} \in \mathbb{R}^{k \times N_s}$ collects the optimal attribute vectors of all seen class training images, $\tilde{\mathbf{Y}} \in \mathbb{R}^{k \times N_s}$ denotes an intermediate matrix that approaches $\mathbf{Y}$, and λ_1, λ_2, λ_3, λ_4 are free parameters.

The first and third terms of Eq. (2) are the L_2-norm and L_1-norm fitting constraints, respectively. Particularly, the third term enforces the noise sparsity in $\mathbf{Y}$, which is a commonly used constraint for data noise and has been proven to be effective. Also, by adding this term, the reliable entries of $\mathbf{Y}$ (with large values) will remain large, while the unreliable entries (with small values) are forced to be close to zero, leading to noise reduction. Fig. 3(b) shows that removing the third term of Eq. (2) leads to significant performance degradation (see 'Single L_1' vs. 'No L_1' in Fig. 3(b)). The second term is a graph smoothness constraint, different from the conventional graph smoothness constraint as a trace norm term. Here, L_1-norm is used to promote the sparsity on the inferred attribute vectors and thus noise reduction (see Fig. 3(b)). Additionally, the last three terms denote the loss function of projection learning for ZSL. The two projection matrices are transpose of each other, similar to those in an auto-encoder [48, 49].

Note that introducing both $\mathbf{Y}$ and $\tilde{\mathbf{Y}}$ makes Eq. (2) much easier to solve. If we do not introduce the intermediate attribute matrix $\tilde{\mathbf{Y}}$, the SAP part of Eq. (2) becomes $\lambda_1\|\mathbf{B}\mathbf{Y}^T\|_1 + \lambda_2\|\mathbf{Y} - \mathbf{Y}^{(s)}\|_1$. That is, the objective function would have two L_1-norm terms, and solving an optimization problem with such an objective function is notoriously hard. We thus replace $\mathbf{Y}$ in $\lambda_1\|\mathbf{B}\mathbf{Y}^T\|_1$ with $\tilde{\mathbf{Y}}$ and add a term $\|\mathbf{Y} - \tilde{\mathbf{Y}}\|_F^2$ to ensure that $\tilde{\mathbf{Y}}$ and $\mathbf{Y}$ are close.

3.3 Optimization Algorithm

Let $\mathcal{F}(\mathbf{Y}, \tilde{\mathbf{Y}}, \mathbf{W})$ denote the objective function in Eq. (2). The optimization problem in Eq. (2) can be solved in two alternating steps as follows:

$$\textbf{SAP}: \ \mathbf{Y}^*, \tilde{\mathbf{Y}}^* = \arg\min_{\mathbf{Y},\tilde{\mathbf{Y}}} \mathcal{F}(\mathbf{Y}, \tilde{\mathbf{Y}}, \mathbf{W}^*) , \quad (3)$$

$$\textbf{BPL}: \ \mathbf{W}^* = \arg\min_{\mathbf{W}} \mathcal{F}(\mathbf{Y}^*, \tilde{\mathbf{Y}}^*, \mathbf{W}) , \quad (4)$$

where $\mathbf{Y}^*$ is initialized with $\mathbf{Y}^{(s)}$, and $\mathbf{W}^*$ is initialized by solving the BPL problem in Eq. (4) with $\mathbf{Y}^* = \tilde{\mathbf{Y}}^* = \mathbf{Y}^{(s)}$.

Sparse Attribute Propagation (SAP). The SAP subproblem in Eq. (3) is solved with the alternating optimization technique as follows: 1) SAP-I: fix $\mathbf{Y} = \mathbf{Y}^*$, and update $\tilde{\mathbf{Y}}$ by $\tilde{\mathbf{Y}}^* = \arg\min_{\tilde{\mathbf{Y}}} \mathcal{F}(\mathbf{Y}^*, \tilde{\mathbf{Y}}, \mathbf{W}^*)$; 2) SAP-II: fix $\tilde{\mathbf{Y}} = \tilde{\mathbf{Y}}^*$, and update $\mathbf{Y}$ by $\mathbf{Y}^* = \arg\min_{\mathbf{Y}} \mathcal{F}(\mathbf{Y}, \tilde{\mathbf{Y}}^*, \mathbf{W}^*)$.

1) SAP-I. Directly solving the SAP-I subproblem is of high computational cost mainly due to the dimension of $\mathbf{B}$ ($N_s \times N_s$). Fortunately, we find a way to dramatically reduce this dimension by using only a small subset of eigenvectors of $\mathbf{L}$. Specifically, we decompose $\tilde{\mathbf{Y}}$ to $\tilde{\mathbf{Y}} = (\mathbf{V}_m \alpha)^T$, where $\alpha = \{\alpha_{ij}\}_{m \times k}$ is an $m \times k$ matrix that collects the reconstruction coefficients and $\mathbf{V}_m$ is an $N_s \times m$ matrix whose columns are the m smallest eigenvectors of $\mathbf{L}$ (i.e., the first m columns of $\mathbf{V}$). The SAP-I subproblem can be reformulated as follows:

$$\begin{aligned}\alpha^* &= \arg\min_{\alpha} \|\mathbf{V}_m \alpha - \mathbf{Y}^{*T}\|_F^2 + \lambda_1 \|\mathbf{B}\mathbf{V}_m \alpha\|_1 \\ &= \arg\min_{\alpha} \sum_{j=1}^{k} (\|\mathbf{V}_m \alpha_{\cdot j} - \mathbf{Y}_{\cdot j}^{*T}\|_2^2 + \lambda_1 \|\mathbf{B}\mathbf{V}_m \alpha_{\cdot j}\|_1) \ , \end{aligned} \tag{5}$$

where $\alpha_{\cdot j}$ and $\mathbf{Y}_{\cdot j}^{*T}$ denote the j-th column of α and $\mathbf{Y}^{*T}$, respectively. The above problem can be decomposed into k independent subproblems:

$$\begin{aligned}&\arg\min_{\alpha_{\cdot j}} \|\mathbf{V}_m \alpha_{\cdot j} - \mathbf{Y}_{\cdot j}^{*T}\|_2^2 + \lambda_1 \|\mathbf{B}\mathbf{V}_m \alpha_{\cdot j}\|_1 \\ &= \arg\min_{\alpha_{\cdot j}} \|\mathbf{V}_m \alpha_{\cdot j} - \mathbf{Y}_{\cdot j}^{*T}\|_2^2 + \lambda_1 \|\sum_{i=1}^{m} \mathbf{\Sigma}^{\frac{1}{2}} \mathbf{V}^T \mathbf{V}_{\cdot i} \alpha_{ij}\|_1 \\ &= \arg\min_{\alpha_{\cdot j}} \|\mathbf{V}_m \alpha_{\cdot j} - \mathbf{Y}_{\cdot j}^{*T}\|_2^2 + \lambda_1 \sum_{i=1}^{m} \Sigma_{ii}^{\frac{1}{2}} |\alpha_{ij}| \ , \end{aligned} \tag{6}$$

where the orthonormality of $\mathbf{V}$ is used to simplify $\|\mathbf{B}\mathbf{V}_m \alpha_{\cdot j}\|_1$. Many off-the-shelf solvers exist for solving L_1-optimization problems like Eq. (6). L1General[3] is employed here, which can solve Eq. (6) at a linear time cost.

To further improve the efficiency, we compute the affinity matrix $\mathbf{A}$ over a k_g-nearest neighbor graph with $k_g \ll N_s$. The time complexity for finding m eigenvectors with the smallest eigenvalues of the sparse matrix $\mathbf{L}$ is $O(m^3 + m^2 N_s + k_g m N_s)$, which scales well to the data.

2) SAP-II. Let $\bar{\mathbf{Y}} = \mathbf{Y} - \mathbf{Y}^{(s)}$. The SAP-II subproblem can be reformulated as

$$\begin{aligned}\bar{\mathbf{Y}}^* &= \arg\min_{\bar{\mathbf{Y}}} \{\|\bar{\mathbf{Y}} + \mathbf{Y}^{(s)} - \tilde{\mathbf{Y}}^*\|_F^2 + \lambda_2 \|\bar{\mathbf{Y}}\|_1 \\ &\quad + \lambda_3 (\|\mathbf{W}^* \mathbf{X}^{(s)} - (\bar{\mathbf{Y}} + \mathbf{Y}^{(s)})\|_F^2 + \|\mathbf{X}^{(s)} - \mathbf{W}^{*T} (\bar{\mathbf{Y}} + \mathbf{Y}^{(s)})\|_F^2)\} \\ &= \arg\min_{\bar{\mathbf{Y}}} \{\text{loss}(\bar{\mathbf{Y}}) + \lambda_2 \|\bar{\mathbf{Y}}\|_1\} \ , \end{aligned} \tag{7}$$

where $\text{loss}(\bar{\mathbf{Y}}) = \|\bar{\mathbf{Y}} + \mathbf{Y}^{(s)} - \tilde{\mathbf{Y}}^*\|_F^2 + \lambda_3 (\|\mathbf{W}^* \mathbf{X}^{(s)} - (\bar{\mathbf{Y}} + \mathbf{Y}^{(s)})\|_F^2 + \|\mathbf{X}^{(s)} - \mathbf{W}^{*T} (\bar{\mathbf{Y}} + \mathbf{Y}^{(s)})\|_F^2)$. Since $\text{loss}(\bar{\mathbf{Y}})$ is a quadratic function w.r.t. $\bar{\mathbf{Y}}$, the above L_1-optimization problem can also be solved efficiently with L1General.

[3] https://www.cs.ubc.ca/ schmidtm/Software/L1General.html

Algorithm 1 Inductive FSZSL with Joint SAP and BPL

Input: Feature representation of the training set $\mathbf{X}^{(s)}$
Initial semantic representation $\mathbf{Y}^{(s)}$
Parameters $k_g, m, \lambda_1, \lambda_2, \lambda_3$
Output: $\mathbf{W}^*$
1: Construct a k_g-NN graph with its affinity matrix $\mathbf{A}$ being defined over $\mathbf{X}^{(s)}$;
2: Find m smallest eigenvectors of the Laplacian matrix $\mathbf{L}$ and store them in $\mathbf{V}_m$;
3: Initialize $\mathbf{W}^*$ by solving Eq. (8) with $\mathbf{Y}^* = \mathbf{Y}^{(s)}$;
4: **for all** iteration = 1, ..., MaxIteration **do**
5: SAP-I: find α^* with Eq. (5) and compute $\tilde{\mathbf{Y}}^*$ as $\tilde{\mathbf{Y}}^* = (\mathbf{V}_m\alpha^*)^T$;
6: SAP-II: find $\bar{\mathbf{Y}}^*$ with Eq. (7) and compute $\mathbf{Y}^*$ as $\mathbf{Y}^* = \bar{\mathbf{Y}}^* + \mathbf{Y}^{(s)}$;
7: BPL: find $\mathbf{W}^*$ by solving Eq. (8);
8: **end for**
9: return $\mathbf{W}^*$.

Bidirectional Projection Learning (BPL). By setting $\frac{\partial \mathcal{F}(\mathbf{Y}^*, \tilde{\mathbf{Y}}^*, \mathbf{W})}{\partial \mathbf{W}} = 0$, the BPL subproblem in Eq. (4) can be solved using a Sylvester equation:

$$(\mathbf{Y}^*\mathbf{Y}^{*T} + \lambda_4\mathbf{I})\mathbf{W} + \mathbf{W}(\mathbf{X}^{(s)}\mathbf{X}^{(s)T}) = 2\mathbf{Y}^*\mathbf{X}^{(s)T} \ , \tag{8}$$

which is solved (using Matlab built-in function) with a time complexity of $O((k^2 + d^2 + kd)N_s + k^3 + d^3)$. We empirically set $\lambda_4 = 0.01$.

By joint SAP and BPL for inductive FSZSL, our algorithm is given in Algorithm 1. Once learned, given the optimal projection matrix $\mathbf{W}^*$ found by our algorithm, we predict the label of a test image $\mathbf{x}_i^{(u)}$ as

$$l_i^{(u)} = \arg\min_j \|\mathbf{x}_i^{(u)} - \mathbf{W}^{*T}\mathbf{z}_j^{(u)}\|_2^2 \ . \tag{9}$$

Since each of iteration steps 5–7 in Algorithm 1 has an efficient solver and our algorithm is shown to converge very quickly (≤ 5 iterations) in the experiments, it has a linear time complexity with respect to the data size.

3.4 Algorithm Analysis

We provide a rigorous analysis on the properties and behaviors of Algorithm 1 as follows. Without loss of generality, we first normalize all of $\|\mathbf{x}_i^{(s)}\|_2$, $\|\mathbf{y}_j^{(s)}\|_1$ to 1, and thus have: $\|\mathbf{Y}^{(s)}\|_F \leq \|\mathbf{Y}^{(s)}\|_1 \leq \sqrt{r}$.

Proposition 1 *The solutions ($\mathbf{Y}^*$ and $\mathbf{W}^*$) found by Algorithm 1 are bounded.*

Proof. (a) Eq. (7) is equivalent to: $\bar{\mathbf{Y}}^* = \arg\min_{\bar{\mathbf{Y}}} \text{loss}(\bar{\mathbf{Y}})$, s.t. $\|\bar{\mathbf{Y}}\|_1 \leq M(\lambda_2)$, where $M(\lambda_2)$ is a constant depended on λ_2. Since $\mathbf{Y}^* = \bar{\mathbf{Y}}^* + \mathbf{Y}^{(s)}$, we have $\|\mathbf{Y}^*\|_F \leq \|\bar{\mathbf{Y}}^*\|_F + \|\mathbf{Y}^{(s)}\|_F \leq C_1$, where $C_1 = M(\lambda_2) + \sqrt{r}$.
(b) Given that $\mathbf{Y}^*\mathbf{Y}^{*T} + \lambda_4\mathbf{I}$ and $\mathbf{X}^{(s)}\mathbf{X}^{(s)T}$ in Eq. (8) are non-negative definite, there exist orthogonal matrices $\mathbf{P}$, $\mathbf{Q}$ s.t. $\mathbf{\Sigma_1}\mathbf{P}^T\mathbf{W}\mathbf{Q} + \mathbf{P}^T\mathbf{W}\mathbf{Q}\mathbf{\Sigma_2} =$

$2\mathbf{P}^T\mathbf{Y}^*\mathbf{X}^{(s)T}\mathbf{Q}$, where $\mathbf{\Sigma_1} = \text{diag}(\theta_1^1, ...\theta_k^1)$ and $\mathbf{\Sigma_2} = \text{diag}(\theta_1^2, ...\theta_d^2)$ collect the eigenvalues of $\mathbf{Y}^*\mathbf{Y}^{*T} + \lambda_4\mathbf{I}$ and $\mathbf{X}^{(s)}\mathbf{X}^{(s)T}$, respectively. Obviously, $\theta_i^1 \geq \lambda_4$ $(i = 1, ..., k), \theta_j^2 \geq 0$ $(j = 1, ..., d)$. Let $\tilde{\mathbf{W}} = \mathbf{P}^T\mathbf{W}\mathbf{Q}$ and $\tilde{\mathbf{R}} = \mathbf{P}^T\mathbf{Y}^*\mathbf{X}^{(s)T}\mathbf{Q}$. We have $\mathbf{\Sigma_1}\tilde{\mathbf{W}} + \tilde{\mathbf{W}}\mathbf{\Sigma_2} = 2\tilde{\mathbf{R}}$. Since $\tilde{w}_{ij} = 2\tilde{r}_{ij}/(\theta_i^1 + \theta_j^2)$, $\|\mathbf{W}^*\|_F = \|\tilde{\mathbf{W}}\|_F \leq 2\|\mathbf{Y}^*\mathbf{X}^{(s)T}\|_F/\lambda_4$. Given that $\|\mathbf{Y}^*\|_F \leq C_1$, we further obtain: $\|\mathbf{W}^*\|_F \leq 2\|\mathbf{Y}^*\|_F\|\mathbf{X}^{(s)T}\|_F/\lambda_4 \leq C_2$, where $C_2 = 2C_1\sqrt{N_s}/\lambda_4$. □

Proposition 2 *The optimal projection matrix* $\mathbf{W}^*$ *found by Algorithm 1 is insensitive to the perturbation of* $\mathbf{Y}^*$*, i.e.,* $\|\triangle\mathbf{W}^*\|_F \to 0$*, if* $\|\triangle\mathbf{Y}^*\|_F \to 0$*.*

Proof. Given $\mathbf{W}^*$ found by Algorithm 1, we have

$$(\mathbf{Y}^*\mathbf{Y}^{*T} + \lambda_4\mathbf{I})\mathbf{W}^* + \mathbf{W}^*(\mathbf{X}^{(s)}\mathbf{X}^{(s)T}) = 2\mathbf{Y}^*\mathbf{X}^{(s)T} \ . \tag{10}$$

When a perturbation $\triangle\mathbf{Y}^*$ is added to $\mathbf{Y}^*$, the optimal projection matrix found by Algorithm 1 is $\hat{\mathbf{W}}^*$:

$$\mathbf{H}\hat{\mathbf{W}}^* + \hat{\mathbf{W}}^*(\mathbf{X}^{(s)}\mathbf{X}^{(s)T}) = 2(\mathbf{Y}^* + \triangle\mathbf{Y}^*)\mathbf{X}^{(s)T} \ , \tag{11}$$

where $\mathbf{H} = (\mathbf{Y}^* + \triangle\mathbf{Y}^*)(\mathbf{Y}^* + \triangle\mathbf{Y}^*)^T + \lambda_4\mathbf{I}$. Let $\triangle\mathbf{W}^* = \hat{\mathbf{W}}^* - \mathbf{W}^*$. Subtracting Eq. (10) from Eq. (11), we obtain $\mathbf{H}\triangle\mathbf{W}^* + \triangle\mathbf{W}^*(\mathbf{X}^{(s)}\mathbf{X}^{(s)T}) = \mathbf{K}$, where $\mathbf{K} = 2\triangle\mathbf{Y}^*\mathbf{X}^{(s)T} - (\triangle\mathbf{Y}^*\triangle\mathbf{Y}^{*T} + \mathbf{Y}^*\triangle\mathbf{Y}^{*T} + \triangle\mathbf{Y}^*\mathbf{Y}^{*T})\mathbf{W}^*$. According to the proof of Prop. 1, we similarly obtain $\|\triangle\mathbf{W}^*\|_F \leq \|\mathbf{K}\|_F/\lambda_4$. We further have:

$$\begin{aligned}\|\triangle\mathbf{W}^*\|_F &\leq [2\sqrt{N_s}\|\triangle\mathbf{Y}^*\|_F + C_2\|\triangle\mathbf{Y}^*\|_F(\|\triangle\mathbf{Y}^*\|_F + 2\|\mathbf{Y}^*\|_F)]/\lambda_4 \\ &\leq \|\triangle\mathbf{Y}^*\|_F[2\sqrt{N_s} + C_2(\|\triangle\mathbf{Y}^*\|_F + 2C_1)]/\lambda_4 \ ,\end{aligned} \tag{12}$$

which means that $\|\triangle\mathbf{W}^*\|_F \to 0$, if $\|\triangle\mathbf{Y}^*\|_F \to 0$. □

Note that Prop. 1 is used in the proof of Prop. 2 as a preliminary proposition. Importantly, from Prop. 2, the optimal projection matrix $\mathbf{W}^*$ used for final recognition is insensitive to the perturbation of $\mathbf{Y}^*$. This thus provides guarantee that Algorithm 1 is robust under our new ZSL setting.

3.5 FSZSL with External Data

Although the test images from unseen classes are not involved in the training process (see Algorithm 1), the proposed algorithm still exploits the unannotated seen class images for SAP. Sometimes, even collecting unannotated seen class images becomes a burden. To address this issue, we thus choose to perform SAP with the unannotated external data from image search engine. By searching relevant images with high-level semantic abstraction (i.e., query) of seen classes, we obtain many free web images to augment the few annotated seen class images at hand. These unannotated web images can be readily exploited for SAP, instead of the unannotated seen class images used in Algorithm 1. When the few annotated seen class images are fused with the unannotated external data, the proposed algorithm can be implemented without any modifications.

Note that the unannotated web images are obtained at a low cost (search key words on a image search engine), and thus unavoidably contain some images that do not belong to the seen classes. For example, given the benchmark seen/unseen class split (i.e., 150/50) of the CUB-200-2011 Birds (CUB) dataset [50], we collect the external data by Google with the query 'North American Bird' (i.e., high-level semantic abstraction of seen classes). With this high-level query, it is very likely that a returned image comes from either seen or unseen classes, and beyond (see Fig. 2). In this paper, we choose to classify the obtained web images using the CNN model proposed in [51], and then discard the images that are classified to unseen classes. Given that [51] has reported a very high accuracy in fine-grained classification, *the effect of possible unseen class images can be suppressed dramatically* during training our SAP model. Therefore, the achieved improvements (if any) are mainly contributed to our SAP model itself.

4 Experiments

4.1 FSZSL on Benchmark Datasets

Datasets and Settings. 1) Datasets. Four widely-used benchmark datasets are selected: (a) Animals with Attributes (AwA) [14] has 30,475 images, 85 attributes, and the seen/unseen class split of 40/10; (b) CUB-200-2011 Birds (CUB) [50] has 11,788 images, 312 attributes, and the seen/unseen split of 150/50; (c) aPascal&Yahoo (aPY) [52] has 15,339 images, 64 attributes, and the class split of 20/12; (d) SUN Attribute (SUN) [53] has 14,340 images, 102 attributes, and the split of 707/10.
2) Semantic and Feature Spaces. First, we establish the semantic space with attributes for the four benchmark datasets, all of which provide the attribute annotations for seen/unseen classes. Second, we extract the ResNet101 [4] features to form the visual feature space as in [54–56].
3) Evaluation Metrics. For the standard FSZSL setting, we compute the multi-way top-1 accuracy as in previous works. For the generalized FSZSL setting (following [45–47]), we compute the harmonic mean of the following two accuracies: acc_u – the top-1 accuracy of classifying the test samples from unseen classes to all seen/unseen classes, and acc_s – the top-1 accuracy of classifying the test samples from seen classes to all seen/unseen classes.
4) Parameter Settings. Our algorithm has five hyperparameters: k_g, m, λ_1, λ_2, λ_3. Given only a few annotated seen class images, it is impossible to select the parameters by cross-validation. Fortunately, our algorithm is shown to be insensitive to these parameters (see the suppl. material). We thus uniformly set $k_g = 300$, $m = 50$, $\lambda_1 = 0.01$, $\lambda_2 = 1e-4$, and $\lambda_3 = 1e-6$. Note that λ_1, λ_2 and λ_3 are small but such small values are needed. For λ_1 and λ_3, they need to be small because the associated terms have much larger values than others. For λ_2, it controls the strength of noise reduction and it needs to be small since the entries of the associated matrix are mostly small (otherwise most entries are forced to be zeros). However, having such small values does not mean that

Table 1. Comparative results (%) of standard FSZSL. Average top-1 accuracy is reported (with standard deviation in bracket).

Dataset	K	RPL [27]	ESZSL [24]	SSE [20]	SAE [12]	ZSKL [57]	RN [55]	PQZSL [23]	AREN [58]	Ours
CUB	5	29.4(1.4)	24.1(1.4)	15.6(2.8)	29.7(1.5)	32.2(1.6)	27.5(7.9)	26.5(1.1)	29.9(0.9)	**40.9(1.5)**
	10	34.3(1.0)	27.7(0.8)	16.3(1.4)	35.8(1.9)	37.8(0.8)	28.0(3.2)	31.0(1.5)	30.9(0.9)	**45.8(0.8)**
	15	38.2(0.9)	30.3(0.6)	18.2(0.9)	39.1(0.8)	40.5(0.5)	31.3(6.5)	36.9(0.5)	31.5(0.7)	**47.5(0.8)**
	20	40.0(0.5)	32.9(0.4)	19.2(1.2)	41.0(0.2)	41.5(0.7)	32.6(4.0)	38.7(1.0)	31.9(0.6)	**48.4(0.3)**
	25	41.4(1.0)	34.9(0.6)	21.0(0.6)	43.0(0.4)	42.3(0.3)	35.4(1.8)	40.3(0.3)	32.0(0.3)	**49.3(0.7)**
AwA	5	50.3(2.0)	26.4(4.0)	39.9(3.2)	58.1(5.3)	54.5(2.7)	28.7(3.2)	40.9(2.9)	60.0(1.1)	**71.0(3.1)**
	10	53.6(2.2)	26.8(5.7)	41.4(5.8)	62.6(4.1)	61.0(2.2)	29.7(1.8)	43.3(7.2)	60.8(0.4)	**74.5(1.8)**
	15	53.8(2.3)	34.4(1.8)	42.2(4.4)	63.2(2.0)	62.6(2.4)	31.1(4.0)	51.2(1.8)	61.1(0.8)	**75.7(1.7)**
	20	55.2(1.7)	40.8(1.7)	42.6(4.5)	65.4(1.7)	65.6(1.5)	32.4(5.2)	54.2(1.9)	61.3(0.6)	**76.0(1.4)**
	25	55.7(1.4)	41.5(2.0)	42.9(3.5)	66.6(1.5)	66.7(1.9)	35.2(3.3)	56.5(2.9)	64.2(5.3)	**77.5(0.9)**
aPY	5	21.4(5.7)	19.2(4.4)	13.1(3.0)	25.5(5.2)	33.6(3.4)	8.0(4.3)	24.8(1.7)	34.1(2.5)	**42.5(8.3)**
	10	22.3(3.6)	19.8(4.0)	14.0(3.4)	31.6(6.8)	35.0(6.1)	27.4(4.6)	26.6(5.4)	36.6(2.5)	**44.1(4.8)**
	15	23.0(2.6)	20.8(5.3)	14.1(4.0)	35.5(6.2)	37.1(4.7)	32.2(2.8)	29.1(2.9)	37.3(3.6)	**44.8(5.0)**
	20	24.9(2.8)	20.6(3.4)	15.3(2.3)	39.2(5.4)	38.7(6.5)	32.7(2.8)	30.7(2.2)	37.5(3.3)	**45.7(4.7)**
	25	25.5(2.8)	21.8(2.1)	17.6(2.1)	40.6(5.0)	40.4(4.6)	35.1(2.5)	32.0(2.2)	37.7(2.6)	**47.4(4.7)**
SUN	1	57.2(3.2)	58.0(4.4)	58.1(3.1)	53.4(2.1)	58.9(5.5)	54.2(3.8)	55.0(3.3)	57.3(1.1)	**81.7(1.9)**
	2	62.4(3.3)	62.5(4.7)	60.2(3.3)	64.4(1.5)	67.8(1.6)	58.7(4.8)	57.8(3.0)	59.4(1.7)	**83.0(2.2)**
	3	64.0(4.1)	65.8(5.2)	60.8(3.2)	70.1(3.1)	70.3(2.4)	60.4(4.0)	66.1(2.6)	61.0(1.0)	**83.3(1.4)**
	4	66.5(3.2)	68.9(4.5)	62.1(3.4)	74.5(2.6)	71.4(2.6)	62.1(5.5)	71.5(2.1)	61.0(1.0)	**83.9(1.4)**
	5	69.1(1.9)	70.2(2.1)	62.6(2.2)	76.8(2.0)	73.4(2.2)	64.6(4.4)	75.9(3.3)	61.5(0.3)	**84.3(1.3)**

the corresponding terms can be dropped. Concretely, when the second term of Eq. (2) is removed (i.e., $\lambda_1 = 0$), the performance drops (see SAP-I+BPL vs. BPL in Fig. 3(a)). When the third term of Eq. (2) is removed (i.e., $\lambda_2 = 0$), the performance drops significantly (see SAP-I+SAP-II+BPL vs. SAP-I+BPL in Fig. 3(a)). As for λ_3, it surely cannot be zero since BPL is needed for ZSL.

5) Compared Methods. We select eight representative/state-of-the-art ZSL models as the baselines: RPL [27], ESZSL [24], SSE [20], SAE [12], ZSKL [57], RN [55], PQZSL [23], and AREN [58]. Note that for the comparison in Table 3, some baselines are selected because they can utilize the propagated attributes (with continuous, rather than binary values) as inputs for ZSL.

Results of Standard FSZSL. The comparative results under the standard FSZSL setting are shown in Table 1. Note that all seen class images from each dataset are provided for training, but only K images per seen class are annotated (the others are unannotated). Since there are only 20 images in each class of SUN, we take $K \in \{1, 2, 3, 4, 5\}$. For fair comparison, all eight ZSL alternatives apply the nearest neighbor classifier over a few annotated seen class images to classify each unannotated image to a seen class (thus its pseudo label and the corresponding attribute vector can be obtained)[4]. We have the following observations: (1) Our model achieves the best results on all four datasets, and the improvements over the second-best range from 6% to 23%. This clearly validates the effectiveness of our model in overcoming the attribute sparsity problem. (2) The performance margin between our model and eight ZSL alternatives

[4] Due to the insufficient initial supervision, stronger label propagation models often induce too much noise. In contrast, with only one-step propagation, such nearest neighbor based label propagation (NN-LP) induces much less noise. Experiments in the suppl. material also show that NN-LP is comparable to MixMatch [59] (one of the strongest, but without denoising) under our FSZSL setting. Therefore, it is reasonable to use NN-LP for all compared ZSL models.

Table 2. Comparative results (%) of generalized FSZSL. Harmonic mean is reported (with standard deviation in bracket).

Dataset	K	RPL [27]	ESZSL [24]	SSE [20]	SAE [12]	ZSKL [57]	RN [55]	PQZSL [23]	AREN [58]	Ours
CUB	5	16.8(1.5)	9.4(0.6)	6.9(1.4)	18.5(1.1)	18.6(0.9)	12.4(2.5)	19.9(1.3)	23.1(2.4)	**27.7(0.5)**
	10	21.3(0.8)	9.9(0.5)	8.7(1.4)	24.9(0.9)	22.2(0.5)	14.3(1.4)	22.5(0.9)	25.9(1.2)	**33.9(0.4)**
	15	23.7(0.3)	10.6(0.9)	9.0(0.6)	28.4(0.6)	24.1(0.3)	15.3(2.6)	24.8(0.7)	26.7(0.6)	**36.1(0.9)**
	20	25.3(0.2)	11.8(0.4)	9.9(0.3)	30.2(0.5)	25.3(0.3)	17.3(1.4)	26.6(0.5)	27.1(0.8)	**37.9(0.7)**
	25	26.1(0.3)	12.7(1.4)	10.5(1.2)	31.8(0.6)	26.0(0.3)	19.9(4.6)	28.1(0.3)	28.1(0.6)	**39.0(0.3)**
AwA	5	40.6(1.6)	27.7(4.1)	19.4(2.4)	33.5(3.9)	45.6(1.7)	36.8(2.4)	37.0(3.1)	37.6(2.0)	**55.1(2.8)**
	10	41.3(1.7)	29.2(4.0)	21.7(2.2)	44.0(2.4)	46.9(1.7)	37.8(4.8)	38.1(2.2)	41.1(2.6)	**59.0(1.7)**
	15	41.4(0.9)	33.2(2.6)	24.2(2.6)	47.8(2.1)	47.4(1.8)	40.3(1.8)	47.3(1.8)	42.4(2.7)	**61.2(1.3)**
	20	41.5(1.6)	37.1(1.5)	28.6(0.6)	50.4(1.4)	48.1(1.5)	41.7(0.6)	50.3(1.1)	42.5(4.3)	**62.7(1.3)**
	25	41.6(1.4)	39.7(2.4)	32.5(0.6)	51.8(0.6)	48.5(1.5)	42.7(3.5)	51.1(1.6)	42.7(1.6)	**63.0(2.0)**

Table 3. Comparative accuracies (%) of FSZSL with external data on the CUB+Web dataset. Note that RPL, ESZSL, SAE, and ZSKL also exploit the propagated attributes obtained by our SAP method for fair comparison.

K	RPL0 [27]	RPL [27]	ESZSL0 [24]	ESZSL [24]	SAE0 [12]	SAE [12]	ZSKL0 [57]	ZSKL [57]	Ours
1	21.5	26.9	10.4	12.8	22.0	27.8	23.0	26.5	**29.1**
2	23.5	29.4	20.6	23.5	24.7	30.6	26.2	30.3	**32.9**
3	26.2	31.4	22.1	27.6	25.1	33.3	28.5	32.7	**36.4**
4	28.9	35.0	23.5	30.8	27.9	35.7	30.5	34.5	**39.1**
5	29.4	37.3	24.1	32.2	29.7	37.6	32.2	35.8	**41.2**

generally becomes bigger as fewer annotated seen class images are provided for training. Our explanation is that: other than eight ZSL alternatives, our model exploits more accurately propagated attribute annotations (obtained by SAP) for BPL. (3) Our model significantly outperforms the state-of-the-art deep ZSL models RN [55] and AREN [58], suggesting that deep models tend to suffer from the annotation sparsity and thus may not be suitable for our new setting.

Results of Generalized FSZSL. We take the generalized FSZSL setting by following [45]. K annotated images per seen class are given for model training as in the standard setting. The comparative results are presented in Table 2. Our model is still shown to achieve the best results under this more challenging setting. Importantly, the obtained even bigger margins suggest that both SAP and BPL can promote the generalization ability of our model.

4.2 FSZSL with External Data

Dataset and Settings. We construct a new dataset called CUB+Web[5] as follows: 1) The training set has 750 annotated images (5 per class) from the 150 seen classes of standard CUB [50], along with 1,205 unannotated web images obtained by Google with the query 'North American Bird'; 2) The test set has 2,946 unannotated images from the 50 unseen classes of CUB. Particularly, we first download top-2,000 web images from Google and discard the images with bird objects from multiple classes (see Fig. 2). Furthermore, we classify the obtained web images using the CNN model proposed in [51], and discard the images that are classified to unseen classes, resulting in 1,205 unannotated web images left. Since [51] has reported a very high accuracy in fine-grained classification,

[5] https://github.com/anonymous04321/cub-web

Fig. 2. Examples of the top images returned by Google with the query 'North American Bird'. We discard the images containing multiple bird classes (marked with red boxes).

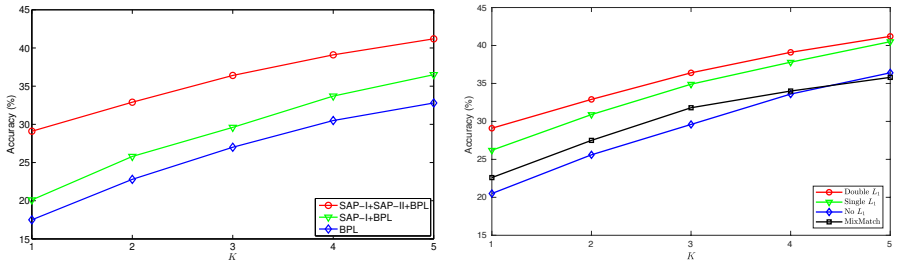
the effect of possible unseen class images can be suppressed dramatically during training our SAP model. Additionally, for CUB+Web, the semantic and feature spaces are formed exactly the same as in Section 4.1.

Comparative Results. We compare with eight closely-related ZSL models: 1) RPL0 – the reverse projection learning model [27] trained with only a few annotated seen class images; 2) RPL – the RPL model trained with not only a few annotated seen class images but also the web images with the propagated attributes obtained by our model; 3) ESZSL0 – the ESZSL model [24] trained like RPL0; 4) ESZSL – ESZSL trained like RPL; 5) SAE0 – the SAE model [12] trained like RPL0; 6) SAE – SAE trained like RPL; 7) ZSKL0 – the ZSKL model [57] trained like RPL0; 8) ZSKL – ZSKL trained like RPL.

Note that four baselines (i.e., SSE, RN, PQZSL, and AREN) require class labels of training samples as inputs. Applying the nearest neighbor classifier like Table 1 makes no sense here, because the external images are not even guaranteed to belong to seen classes (there may be outliers although the unseen class images have been mostly removed). These baselines thus are inapplicable, but others can still benefit from external data (with propagated attributes).

The comparative results in Table 3 (with K annotated images per seen class) show that: (1) The five models (i.e., RPL, ESZSL, SAE, ZSKL and ours) trained using extra web images with propagated attributes lead to significant improvements over those without using extra web images (i.e., RPL0, ESZSL0, SAE0 and ZSKL0), validating the effectiveness of our SAP method. (2) Due to iterative optimization between BPL and SAP, our model still outperforms RPL, ESZSL, SAE, and ZSKL (although they also utilize the web images with propagated attributes obtained by our SAP method).

Further Evaluation. 1) Ablation Study. To evaluate the contribution of each component (SAP-I, SAP-II, or BPL) of our full model, we conduct experiments by adding more components to the BPL model. The ablative results in Fig. 3(a) show that: (1) The SAP-I step by solving Eq. (5) yields better results (see SAP-I+BPL vs. BPL). (2) The SAP-II step by solving Eq. (7) obtains further improvements (see SAP-I+SAP-II+BPL vs. SAP-I+BPL), which become more significant with fewer annotated seen class images (i.e., smaller K).



(a) Ablation study (b) Alternative attribute propagation

Fig. 3. (a) Ablation study results for our full model on the CUB+Web dataset. (b) Comparative results obtained by different attribute propagation models (the same BPL is used for ZSL) on the CUB+Web dataset.

2) Alternative Attribute Propagation. We compare four alternative attribute propagation models: 1) 'Double L_1': our model formulated in Eq. (2); 2) 'Single L_1': $\|\mathbf{B}\tilde{\mathbf{Y}}^T\|_1$ used in our model is replaced by $\|\mathbf{B}\tilde{\mathbf{Y}}^T\|_F^2$; 3) 'No L_1': $\|\mathbf{Y} - \mathbf{Y}^{(s)}\|_1$ is removed from the second model 'Single L_1'; 4) 'MixMatch' [59]: our SAP is replaced by the state-of-the-art semi-supervised learning (SSL) method to perform attribute propagation. The comparative results are presented in Fig. 3(b). As expected, more L_1-norm regularization terms used for attribute propagation lead to better results, due to the stronger noise reduction ability. Interestingly, under our FSZSL setting where only few labelled seen class samples can be used as initial supervision, MixMatch (one of the strongest, but without denoising) performs even worse than 'Double L_1' and 'Single L_1'. This shows the importance of noise reduction during performing attribute propagation for FSZSL (which is also our main motivation of developing SAP).

5 Conclusion

In this paper, we have investigated the challenging problem of ZSL with less human annotation. For the first time, we define the new FSZSL setting where only a few annotated seen class images are given for training. To overcome the annotation sparsity, we propose a novel inductive ZSL model by formulating SAP and BPL within a unified framework, with rigorous theoretic analysis provided. Moreover, we generalize the proposed model to FSZSL with external data as well as social image annotation. Extensive experiments show that the proposed model achieves state-of-the-art results.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China (61976220 and 61832017), and Beijing Outstanding Young Scientist Program (BJJWZYJH012019100020098).

References

1. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems. (2012) 1097–1105
2. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
3. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. IJCV **115** (2015) 211–252
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. (2016) 770–778
5. Fu, Y., Hospedales, T.M., Xiang, T., Gong, S.: Transductive multi-view zero-shot learning. TPAMI **37** (2015) 2332–2345
6. Fu, Z., Xiang, T., Kodirov, E., Gong, S.: Zero-shot object recognition by semantic manifold distance. In: CVPR. (2015) 2635–2644
7. Ba, L.J., Swersky, K., Fidler, S., Salakhutdinov, R.: Predicting deep zero-shot convolutional neural networks using textual descriptions. In: ICCV. (2015) 4247–4255
8. Qiao, R., Liu, L., Shen, C., van den Hengel, A.: Less is more: zero-shot learning from online textual documents with noise suppression. In: CVPR. (2016) 2249–2257
9. Ye, M., Guo, Y.: Zero-shot classification with discriminative semantic representation learning. In: CVPR. (2017) 7140–7148
10. Karessli, N., Akata, Z., Schiele, B., Bulling, A., et al.: Gaze embeddings for zero-shot image classification. In: CVPR. (2017) 4525–4534
11. Long, Y., Shao, L.: Learning to recognise unseen classes by a few similes. In: ACM-MM. (2017) 636–644
12. Kodirov, E., Xiang, T., Gong, S.: Semantic autoencoder for zero-shot learning. In: CVPR. (2017) 3174–3183
13. Wang, W., Pu, Y., Verma, V., et al.: Zero-shot learning via class-conditioned deep generative models. In: AAAI. (2018) 4211–4218
14. Lampert, C.H., Nickisch, H., Harmeling, S.: Attribute-based classification for zero-shot visual object categorization. TPAMI **36** (2014) 453–465
15. Lee, H., Battle, A., Raina, R., Ng, A.: Efficient sparse coding algorithms. In: Advances in Neural Information Processing Systems. (2007) 801–808
16. Sinha, K., Belkin, M.: Semi-supervised learning using sparse eigenfunction bases. In: Advances in Neural Information Processing Systems. (2010) 1687–1695
17. Johnson, J., Ballan, L., Fei-Fei, L.: Love thy neighbors: Image annotation by exploiting image metadata. In: ICCV. (2015) 4624–4632
18. Hu, H., Zhou, G.T., Deng, Z., Liao, Z., Mori, G.: Learning structured inference neural networks with label relations. In: CVPR. (2016) 2960–2968
19. Liu, F., Xiang, T., Hospedales, T.M., Yang, W., Sun, C.: Semantic regularisation for recurrent image annotation. In: CVPR. (2017) 4160–4168
20. Zhang, Z., Saligrama, V.: Zero-shot learning via semantic similarity embedding. In: ICCV. (2015) 4166–4174
21. Changpinyo, S., Chao, W., Gong, B., Sha, F.: Synthesized classifiers for zero-shot learning. In: CVPR. (2016) 5327–5336
22. Liu, G., Guan, J., Zhang, M., Zhang, J., Wang, Z., Lu, Z.: Joint projection and subspace learning for zero-shot recognition. In: ICME. (2019) 1228–1233

23. Li, J., Lan, X., Liu, Y., Wang, L., Zheng, N.: Compressing unknown images with product quantizer for efficient zero-shot classification. In: CVPR. (2019) 5463–5472
24. Romera-Paredes, B., Torr, P.H.S.: An embarrassingly simple approach to zero-shot learning. In: ICML. (2015) 2152–2161
25. Kodirov, E., Xiang, T., Fu, Z., Gong, S.: Unsupervised domain adaptation for zero-shot learning. In: ICCV. (2015) 2452–2460
26. Zhang, F., Shi, G.: Co-representation network for generalized zero-shot learning. In: ICML. (2019) 7434–7443
27. Shigeto, Y., Suzuki, I., Hara, K., Shimbo, M., Matsumoto, Y.: Ridge regression, hubness, and zero-shot learning. In: ECML-PKDD. (2015) 135–151
28. Zhao, A., Ding, M., Guan, J., Lu, Z., Xiang, T., Wen, J.R.: Domain-invariant projection learning for zero-shot recognition. In: Advances in Neural Information Processing Systems. (2018) 1027–1038
29. Li, A., Lu, Z., Guan, J., Xiang, T., Wang, L., Wen, J.R.: Transferrable feature and projection learning with class hierarchy for zero-shot learning. IJCV (2020) 1–18
30. Huo, Y., Guan, J., Zhang, J., Zhang, M., Wen, J.R., Lu, Z.: Zero-shot learning with few seen class samples. In: ICME. (2019) 1336–1341
31. Guan, J., Lu, Z., Xiang, T., Li, A., Zhao, A., Wen, J.R.: Zero and few shot learning with semantic feature synthesis and competitive learning. TPAMI (2020) 1–14
32. Long, Y., Shao, L.: Describing unseen classes by exemplars: Zero-shot learning using grouped simile ensemble. In: WACV. (2017) 907–915
33. Li, A., Lu, Z., Wang, L., Xiang, T., Wen, J.R.: Zero-shot scene classification for high spatial resolution remote sensing images. IEEE Trans. Geoscience and Remote Sensing **55** (2017) 4157–4167
34. Li, X., Guo, Y., Schuurmans, D.: Semi-supervised zero-shot classification with label representation learning. In: ICCV. (2015) 4211–4219
35. Guo, Y., Ding, G., Jin, X., Wang, J.: Transductive zero-shot recognition via shared model space learning. In: AAAI. (2016) 3494–3500
36. Shojaee, S.M., Baghshah, M.S.: Semi-supervised zero-shot learning by a clustering-based approach. arXiv preprint arXiv:1605.09016 (2016)
37. Wang, Q., Chen, K.: Zero-shot visual recognition via bidirectional latent embedding. IJCV **124** (2017) 356–383
38. Yu, Y., Ji, Z., Li, X., Guo, J., Zhang, Z., Ling, H., Wu, F.: Transductive zero-shot learning with a self-training dictionary approach. arXiv preprint arXiv:1703.08893 (2017)
39. Bergamo, A., Torresani, L.: Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach. In: Advances in Neural Information Processing Systems. (2010) 181–189
40. Duan, L., Xu, D., Chang, S.F.: Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach. In: CVPR. (2012) 1338–1345
41. Zhang, H., Liu, S., Zhang, C., Ren, W., Wang, R., Cao, X.: Sketchnet: Sketch classification with web images. In: CVPR. (2016) 1105–1113
42. Niu, L., Tang, Q., Veeraraghavan, A., Sabharwal, A.: Learning from noisy web data with category-level supervision. In: CVPR. (2018) 7689–7698
43. Niu, L., Veeraraghavan, A., Sabharwal, A.: Webly supervised learning meets zero-shot learning: A hybrid approach for fine-grained classification. In: CVPR. (2018) 7171–7180
44. Gan, C., Sun, C., Nevatia, R.: Deck: Discovering event composition knowledge from web images for zero-shot event detection and recounting in videos. In: AAAI. (2017) 4032–4038

45. Chao, W.L., Changpinyo, S., Gong, B., Sha, F.: An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In: ECCV. (2016) 52–68
46. Rahman, S., Khan, S.H., Porikli, F.: A unified approach for conventional zero-shot, generalized zero-shot and few-shot learning. arXiv preprint arXiv:1706.08653 (2017)
47. Xian, Y., Schiele, B., Akata, Z.: Zero-shot learning - the good, the bad and the ugly. In: CVPR. (2017) 4582–4591
48. Lu, X., Tsao, Y., Matsuda, S., Hori, C.: Speech enhancement based on deep denoising autoencoder. In: Interspeech. (2013) 436–440
49. Feng, F., Wang, X., Li, R.: Cross-modal retrieval with correspondence autoencoder. In: ACM-MM. (2014) 7–16
50. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology (2011)
51. Yu, C., Zhao, X., Zheng, Q., Zhang, P., You, X.: Hierarchical bilinear pooling for fine-grained visual recognition. In: ECCV. (2018) 595–610
52. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: CVPR. (2009) 1778–1785
53. Patterson, G., Xu, C., Su, H., Hays, J.: The sun attribute database: Beyond categories for deeper scene understanding. IJCV **108** (2014) 59–81
54. Chen, L., Zhang, H., Xiao, J., Liu, W., Chang, S.F.: Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In: CVPR. (2018) 1043–1052
55. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. In: CVPR. (2018) 1199–1208
56. Xian, Y., Lorenz, T., Schiele, B., Akata, Z.: Feature generating networks for zero-shot learning. In: CVPR. (2018) 5542–5551
57. Zhang, H., Koniusz, P.: Zero-shot kernel learning. In: CVPR. (2018) 7670–7679
58. Xie, G.S., Liu, L., Jin, X., Zhu, F., Zhang, Z., Qin, J., Yao, Y., Shao, L.: Attentive region embedding network for zero-shot learning. In: CVPR. (2019) 9384–9393
59. Berthelot, D., Carlini, N., Goodfellow, I.J., Papernot, N., Oliver, A., Raffel, C.: Mixmatch: A holistic approach to semi-supervised learning. In: Advances in Neural Information Processing Systems. (2019) 5050–5060