Zero-Shot Learning posed as a Missing Data Problem

Bo Zhao[1], Botong Wu[1], Tianfu Wu[2], Yizhou Wang[1]
[1]Nat'l Engineering Laboratory for Video Technology,
Key Laboratory of Machine Perception (MoE),
Cooperative Medianet Innovation Center, Shanghai,
Sch'l of EECS, Peking University, Beijing, 100871, China
[2]Department of ECE and the Visual Narrative Cluster, North Carolina State University
{bozhao, botongwu, Yizhou.Wang} @pku.edu.cn, tianfu_wu@ncsu.edu

Abstract

This paper presents a method of zero-shot learning (ZSL) which poses ZSL as the missing data problem, rather than the missing label problem. Specifically, most existing ZSL methods focus on learning mapping functions from the image feature space to the label embedding space. Whereas, the proposed method explores a simple yet effective transductive framework in the reverse way – our method estimates data distribution of unseen classes in the image feature space by transferring knowledge from the label embedding space. Following the transductive setting, we leverage unlabeled data to refine the initial estimation. In experiments, our method achieves the highest classification accuracies on two popular datasets, namely, ***96.00%*** *on AwA and* ***60.24%*** *on CUB.*

1. Introduction

The recent success of deep learning heavily relies on a large amount of labeled training data. Popular deep neural networks, e.g. VGG [19], GoogLeNet [21] and ResNet [6], require hundreds to thousands of labeled training data to learn a new concept. For some classes, e.g., rare wildlife and unusual diseases, it is expensive even impossible to collect training samples. Traditional supervised learning frameworks cannot work well in this situation. Zero-shot learning (ZSL) that aims to recognize instances from unseen classes is considered to be a promising solution.

In ZSL, data are split into labeled seen classes (source domain) and unlabeled unseen classes (target domain where labels are missing). The seen classes and unseen classes are disjointed. Therefore, "auxiliary information" is introduced to enable knowledge transfer from seen classes to unseen ones so that given a datum from the unseen classes, its label can be predicted. Often used auxiliary information includes attributes[9], textual description[10] and word vectors of labels[20], etc. In most practice, labels are embedded in "label embedding space" using auxiliary information. Data (e.g., images) are embedded in (e.g., image) feature space (using hand-craft or deep learning feature extractors). In the following of this paper, we introduce ZSL in the context of image recognition.

One popular type of ZSL is implemented in an inductive way, i.e. models are trained on seen classes then applied to unseen classes directly. Usually, inductive ZSL includes three steps: i) embedding images and labels in the image feature space and label embedding space respectively; ii) learning the mapping function from the image feature space to the label embedding space (F→E) using seen classes data; iii) mapping an unseen image to the label embedding space using the learned mapping function and predicting its label. In this way, ZSL is posed as a *missing label problem*. Many existing methods of this type (e.g., [20][2][16]) assume a global linear mapping F→E between the two spaces. Romera-Paredes *et al.*[17] present a very simple ZSL approach using this assumption, and extend the approach to a kernel version. However, the global linear mapping assumption can be over-simplified. Wang *et al.*[23] propose to utilize local relational knowledge to synthesize virtual unseen image data, but then back to the global linear assumption to learn the mapping F→E using both the seen data and synthesised unseen data. We observe that the synthesized data of unseen classes are not accurate, in addition, back to the global linear mapping assumption further damage the ZSL performance. Hence we propose to adjust the synthesized unseen data according to the manifold structure of real unseen data.

Accordingly, some transductive ZSL approaches are proposed for alleviating the domain shift problem[5]. In transductive ZSL, (unlabeled) real unseen data are utilized for refining the trained model, e.g., the label embedding space[11] and the mapping function. In [7], a dictionary for

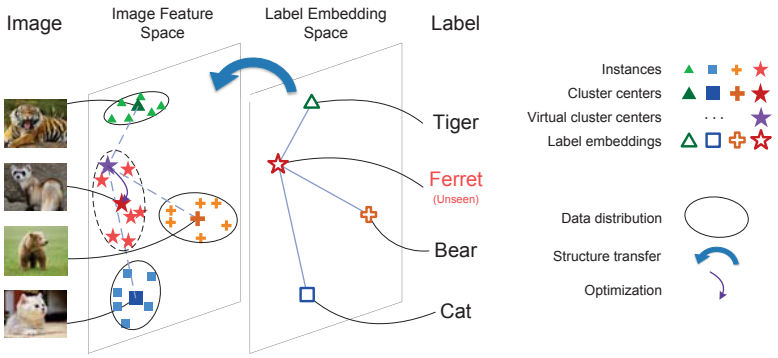

Figure 1. Illustration of the proposed method. The manifold structure (the straight lines) in the label embedding space is transferred to the image feature space for synthesizing the virtual cluster center (the purple star) of an unseen class. The purple arrow points to a refined cluster center (the red star), which demonstrates that the synthesized virtual cluster center is optimized after running the Expectation-Maximization algorithm so that unseen data are assigned to labels according to the data distribution.

the target domain is learned using regularised sparse coding, and the dictionary learned on the source domain serves as the regularizer. In [27], a structured prediction approach is proposed. Clusters on unseen data are generated using K-means, then a bipartite graph matching between these clusters and labels is optimized based on the learned similarity matrix on seen data.

Most aforementioned methods aim at learning a potentially complex mapping from F→E. Under circumstances such as the number of classes is large and there exists polysemy in text labels, such many-to-one "clean mapping" can be hard to learn. In this paper, we study a novel transductive zero-shot learning method (shown in Figure.1), which transfers the manifold structure in the label embedding space to the image feature space (E→F), and adapts the transferred structure according to the underlying data distribution of both seen and unseen data in the image feature space. As the proposed method associates data to the label, we categorize it as a *missing data method* in contrast to the conventional *missing label methods*.

Our method is based on two assumptions, i) data of each class in the image feature space approximately follow a Gaussian distribution, ii) the manifold structure of label embeddings is approximate to that of cluster centers in the image feature space. It is observed that data in each class form a tight cluster[27]. The cluster center serves as the representation of each class. Hence, the cluster center and corresponding label embedding of each class form a datum pair. In this paper, data distributions are modeled by Gaussians, and the cluster center is defined as the mean of a Gaussian. Our method consists of three main steps:

i) The cluster center of each seen class is estimated in the image feature space. ii) The manifold structure is estimated in the labeling embedding space, and is transferred to the image feature space so as to synthesize virtual cluster centers of the unseen classes. iii) The virtual cluster centers are refined, at the same time, each unseen instance is associated to an unseen label (label prediction) by the Expectation-Maximization (EM) algorithm.

We verify the effectiveness of our two assumptions in sufficient experiments. Experiments show that the proposed method outperforms the state-of-the-art on two popular datasets, namely, the Animals with Attributes (AwA) and the Caltech-UCSD Birds-200-2011 (CUB).

2. The Proposed Method

N^s seen classes data are denoted as $(\boldsymbol{X}^s, \boldsymbol{Y}^s) = \{(\boldsymbol{x}_1^s, y_1^s), ..., (\boldsymbol{x}_{N^s}^s, y_{N^s}^s)\}$, and N^u unseen classes data are denoted as $(\boldsymbol{X}^u, \boldsymbol{Y}^u) = \{(\boldsymbol{x}_1^u, y_1^u), ..., (\boldsymbol{x}_{N^u}^u, y_{N^u}^u)\}$. Each datum $\boldsymbol{x}_i^s$ or $\boldsymbol{x}_i^u \in \Re^{d\times 1}$ is a d-dimensional feature vector in the image feature space. y_i^s or y_i^u denotes the labels. The label y_i^u of each unseen instance x_i^u is unknown. The label sets of the seen and unseen classes are disjointed, i.e. $\boldsymbol{Y}^s \cap \boldsymbol{Y}^u = \emptyset$. Attributes or/and word vectors (auxiliary information) are used as label embeddings denoted as $\boldsymbol{E}^s = \{\boldsymbol{e}_1^s, ..., \boldsymbol{e}_{K^s}^s\}$ and $\boldsymbol{E}^u = \{\boldsymbol{e}_1^u, ..., \boldsymbol{e}_{K^u}^u\}$ for seen and unseen classes respectively. $\boldsymbol{e}_k^s$ and $\boldsymbol{e}_k^u \in \Re^{d'\times 1}$. Using the seen data pairs $(\boldsymbol{x}_i^s, y_i^s)$, ZSL aims to predict labels y_i^u for each unseen instance $\boldsymbol{x}_i^u$ by leveraging the auxiliary information $\boldsymbol{E}^s$ and $\boldsymbol{E}^u$ for knowledge transfer.

2.1. Estimation of Seen Cluster Centers

In the image feature space, data from different classes are separable. By dimensionality reduction (using t-SNE[13]), it is observed that data in each class form a tight cluster (shown in Figure. 2). Similar to recent ZSL works, e.g., [27] [23], we assume that

Assumption 1 *Data of each class approximately follow a Gaussian distribution* $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ *in the image feature space.*

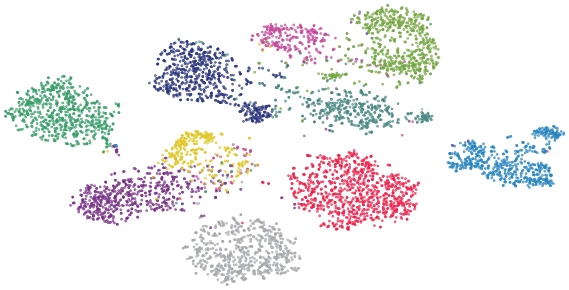

Figure 2. Visualization of the data (CNN features) in default 10 unseen classes of Animals with Attributes dataset using t-SNE.

Although the assumption of Gaussian distribution may not be very precise, it works well in our and others' experiments.

It is worth noting that, in the literature people used Nearest-Neighbor classifiers to assign labels to unseen data, e.g., [15] [7], the underlying assumption is that the distribution of data is isotropic Gaussian. Different from them, we estimate the parameters of the Gaussians.

The data (image features) in each class form a cluster. With our assumption, we formulate each cluster using Gaussian parameters $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, i.e. the mean and covariance. The cluster centers, i.e. the mean, of all classes in the image feature space are denoted as $\boldsymbol{M} = \{\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_K\}$. The cluster center and corresponding label embedding of each class form a datum pair $(\boldsymbol{\mu}_k, \boldsymbol{e}_k)$.

As the labels of seen classes data are provided, we can estimate cluster centers of seen classes directly, denoted as $\boldsymbol{M}^s$.

2.2. Synthesis of Virtual Unseen Cluster Centers

One of the key challenges in ZSL is to explore the relationship between the image feature space and the label embedding space. The label embedding is either pre-designed (e.g. by the annotated attribute vectors) or pre-trained on a large corpus (e.g. by word vectors). Although there may not be an accurate global linear mapping from the image feature space to the label embedding space, manifold structures in the two spaces may be similar. For instance, "cat" is more close to "dog" than "elephant" in the visual space. This closeness is kept in the (semantic) label embedding space, e.g. the attribute space or word vector space. In this paper we focus on exploiting the manifold structure rather than the global one. Hence we assume that

Assumption 2 *The manifold structure of label embeddings is approximate to that of cluster centers in the image feature space and can be transferred for synthesizing the virtual cluster centers of the unseen classes.*

This is formulated as

$$\boldsymbol{E}^u = R\left(\boldsymbol{E}^s\right) \Rightarrow \widehat{\boldsymbol{M}^u} = R\left(\boldsymbol{M}^s\right), \tag{1}$$

where $\widehat{\boldsymbol{M}^u} = \{\widehat{\boldsymbol{\mu}_1^u}, ..., \widehat{\boldsymbol{\mu}_{K^u}^u}\}$ denotes the synthesized (namely estimated) virtual cluster centers of the unseen classes. There are many choices of the synthesis function $R(\cdot)$ that can approximate the manifold structure of the label embeddings, such as Sparse Coding[14], Locally Linear Embedding[18] and so on.

Our assumption is more general than the traditional global linear mapping assumption. It can be proven that, when the synthesis function $R(\cdot)$ is a linear mapping, this synthesis process is equivalent to learning a linear mapping from the label embedding space to the image feature space using balanced samples in each class. The more interesting property is that we can choose $R(\cdot)$ with locality to explore the manifold structure rather than the global structure in the two spaces.

In the literature, many works assume the two spaces observe a global linear transformation so that the structure of the image features can be transferred to the label embeddings via a global linear mapping, e.g., [2][16]. We observe that such an assumption is over-simplified. There are works assuming that a global non-linear mapping may exist between the two spaces[17], e.g., using kernel methods. However, it is prone to get overfitting on the seen data and obtain bad performance on the unseen data. In contrast, our manifold preserving assumption works well empirically in the experiments.

2.2.1 Synthesis via Sparse Coding

We choose Sparse Coding[14] (inspired by [23]) to approximate the manifold structures of the image features and label embeddings. In our implementation, label embeddings of the seen classes serve as the dictionary. Then we compute the sparse linear reconstruction coefficients of the bases for unseen label embeddings. According to the Sparse Coding theory, we minimize the following loss function to obtain the coefficients $\boldsymbol{\alpha}$.

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{e}_k^u - \boldsymbol{E}^s\boldsymbol{\alpha}\|^2 + \lambda|\boldsymbol{\alpha}|_1, \tag{2}$$

where $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_{K^s}]^T$. This loss function is convex and easy to optimize.

Then, we transfer such local structure from the label embedding space to the image feature space and synthesize the virtual cluster center of each unseen class using the same set of coefficients, i.e. $\widehat{\boldsymbol{\mu}_k^u} = \boldsymbol{M}^s\boldsymbol{\alpha}$, where the components in $\boldsymbol{E}^s$ and $\boldsymbol{M}^s$ correspond to each other. This transferring is valid because the distribution of an unseen class in the image space is assumed to be Gaussian and the components either in $\boldsymbol{E}^s$ or $\boldsymbol{M}^s$ are assumed to be independent.

After synthesizing all unseen cluster centers (say K^u of them), the distribution of all unseen instances $\{\boldsymbol{x}_n^u\}$ in the image feature space is a Gaussian Mixture Model (GMM),

$$p\left(\boldsymbol{x}_n^u\right) = \sum_{k=1}^{K^u} \pi_k \mathcal{N}\left(\boldsymbol{x}_n^u | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) \tag{3}$$

π_k denotes the kth mixing coefficient and its initial value is assumed to be $1/K^u$. The value of $\boldsymbol{\mu}_k = \widehat{\boldsymbol{\mu}_k^u}$. We initialize each $\boldsymbol{\Sigma}_k$ using an identity matrix. $\boldsymbol{x}_n^u$ denotes the nth image in $\boldsymbol{X}^u$.

The synthesized virtual cluster centers approximate the distribution of the unseen data in the image feature space. However, they may not be accurate. Next, we optimize/refine the cluster centers, at the same time, associate each unseen image to an unseen label. This is the reason we pose our ZSL as a missing data problem.

2.3. Solving the Missing Data Problem

We impute unseen image labels and update the GMM parameters using the Expectation-Maximization (EM) algorithm.

The objective function is defined as the log of the likelihood function,

$$\ln p\left(\boldsymbol{X}^u | \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\right) = \sum_{n=1}^{N^u} \ln \sum_{k=1}^{K^u} \pi_k \mathcal{N}\left(\boldsymbol{x}_n^u | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) \tag{4}$$

In the Expectation step, the conditional probability of the latent variable $y_n^u = k$ given $\boldsymbol{x}_n^u$ under the current parameter is

$$p(y_n^u = k | \boldsymbol{x}_n^u) = \frac{\pi_k \mathcal{N}\left(\boldsymbol{x}_n^u | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K^u} \pi_j \mathcal{N}\left(\boldsymbol{x}_n^u | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right)}. \tag{5}$$

This is the posterior probability of an unseen image $\boldsymbol{x}_n^u$ belonging to label k.

In the Maximization step, the model updates the parameters using the posterior probability.

$$\boldsymbol{\mu}_k^{new} = \frac{1}{N^u} \sum_{n=1}^{N^u} p(y_n^u = k | \boldsymbol{x}_n^u) \boldsymbol{x}_n^u \tag{6}$$

$$\boldsymbol{\Sigma}_k^{new} = \frac{1}{N^u} \sum_{n=1}^{N^u} p(y_n^u = k | \boldsymbol{x}_n^u) (\boldsymbol{x}_n^u - \boldsymbol{\mu}_k^{new})^T (\boldsymbol{x}_n^u - \boldsymbol{\mu}_k^{new}) \tag{7}$$

$$\pi_k^{new} = \frac{N_k^u}{N^u} \tag{8}$$

where

$$N_k^u = \sum_{n=1}^{N^u} p(y_n^u = k | \boldsymbol{x}_n^u) \tag{9}$$

K^u and N^u denote the number of all unseen classes and instances respectively. We iterate the E-step and M-step until convergence. After the convergence, the parameters of the data distribution are refined and the unseen instances are assigned with labels.

2.3.1 Regularization

During the EM process when estimating the GMM, each covariance matrix $\boldsymbol{\Sigma}_k$ should be nonsingular, i.e. invertible. For a reliable computation, empirically, the number of data in each class N_k should be greater than the square of feature dimension, i.e. $\forall_k, N_k \geq \lambda d^2, s.t.\ \lambda \geq 1$. λ is a coefficient. However, this may not be satisfied in some situations when feature dimension is high but only a small number of data are provided per class.

We employ two tricks to solve this problem, namely, dimensionality reduction and regularization of $\boldsymbol{\Sigma}_k$. For dimensionality reduction, we choose to use linear dimension reduction methods, e.g. principal components analysis (PCA), to reduce the image feature representation to d dimensional, which is much smaller than the original one.

If we only choose to stabilize the computation by reducing the image feature dimension, the label prediction accuracy will degrade quickly. Hence, we also resort to another solution, i.e., regularizing $\boldsymbol{\Sigma}_k$. Here, we present two regularization methods of $\boldsymbol{\Sigma}_k$, namely, diagonal $\boldsymbol{\Sigma}_k$, $s.t.\ N_k \geq \lambda d$ and unit $\boldsymbol{\Sigma}_k$, $s.t.\ N_k \geq 1$. Diagonal $\boldsymbol{\Sigma}_k$ means that $\boldsymbol{\Sigma}_k$ is assumed to be a diagonal matrix. Unit $\boldsymbol{\Sigma}_k$ means that $\boldsymbol{\Sigma}_k$ is an identity matrix. These two regularization methods simplify $\boldsymbol{\Sigma}_k$ in an increasing order. We choose to use a simpler one if the number of the data is smaller.

3. Experiments

3.1. Datasets & Settings

We evaluate the proposed method by conducting experiments on two popular datasets, i.e., the Animals with Attributes (AwA) [8] and the Caltech-UCSD Birds-200-2011 (CUB) [22]. i) AwA contains 50 animal classes (coarse-grained) and 85 manual attributes (both binary and continuous). Ten classes serve as the unseen classes and the remaining forty are utilized as the seen classes. ii) CUB is a fine-grained image dataset which contains 200 species of birds annotated with 312 binary attributes. Commonly, 50 species are chosen as the unseen classes, and the rest are the seen classes.

For AwA, we use i) VGG-fc7, ii) GoogLeNet, iii) ResNet features. For CUB, we use iv) GoogLeNet and v) ResNet. i) is provided along with the dataset[8]. Image features (ii, iii, iv) and label embeddings (attributes and word vectors) are provided by [23]. The parameter λ used in the loss function of Sparse Coding is set as a fixed value (0.5) in all experiments for speeding up training process.

In the analysis of our assumptions, we evaluate our method using "Many random splits". We report the average results of 300 random splits for the high reliability. When compared to the state-of-the-art (in Sec. 3.5), we follow the default splits [23] on AwA and CUB that are widely used.

	Setting	VGG (%)	GoogLeNet (%)	ResNet (%)	GoogLeNet+ResNet (%)
AwA	Random	94.26	92.55	88.80	90.71
CUB		-	81.73	82.63	86.06
AwA	Default	*96.52*	94.58	90.74	92.64
CUB		-	81.24	80.03	*85.03*

Table 1. Evaluation of data distribution assumption. Our data distribution assumption is effective in different kinds of feature spaces. The experimental upper bound performance on the default splits of AwA and CUB are *96.52%* and *85.03%* respectively.

3.2. Evaluation of Data Distribution Assumption

First, we examine if **Assumption 1** is a reasonable assumption, i.e. the data of each class approximately subject to a Gaussian distribution in the image feature space. The idea is to show that under this assumption the upper bound performance of the proposed method exceeds the state-of-the-art performance by a considerable margin.

To obtain the upper bound performance of the proposed method under **Assumption 1**, we conduct an upper-bound experiment (traditional supervised learning), in which the labels of all data (both seen and unseen) are given. Data are separated into training and testing parts. We estimate the Gaussian distribution for each unseen class according to the ground-truth labeled training data. Then the label of each testing datum is predicted as the one with the maximum likelihood of the Gaussians/classes. The mean classification accuracy consequently can be computed.

Table1 shows the upper-bound classification performances of the proposed method based on **Assumption 1** in different image feature spaces. The result on CUB using VGG features is not reported due to the lack of data. We implement experiments on both random and default splits.

The experimental upper bound performance under **Assumption 1** on the default splits of AwA and CUB are *96.52%* and *85.03%* using VGG and GoogLeNet+ResNet features respectively. According to Table3, the proposed upper-bound performance is much higher than the corresponding state-of-the-art performance – 81.41% on AwA and 55.59% on CUB, which is achieved by repeating the experiment in [23] using the same data. Therefore, the Gaussian assumption of the distribution of data is reasonably good currently when comparing the proposed method with the other state-of-the-arts.

3.3. Effectiveness of Manifold Transfer

To justify **Assumption 2**, we evaluate the classification performance using synthesized virtual cluster centers directly (without EM optimization). This strategy can be viewed as the inductive version of our method (denoted as Ours_I). We run 300 random trials on AwA and CUB respectively. Features extracted from VGG-fc7 (4096-dim) for AwA and GoogLeNet+ResNet (3072-dim) for CUB are utilized. We use the same label embeddings as those in [23]. According to our analysis in Sec.2.3.1, the image feature dimension is reduced to 80-dim on AwA, because the minimum number of images of each class is 92. We also reduce the feature dimension of CUB data to 400-dim for speeding up the computation. Three types of label embedding are tested, namely, attributes(A), word vectors(W) and attributes with word vectors(A+W). Results using different settings are shown in Table2.

We also implement a baseline experiment to illustrate the priority of our manifold transfer assumption. In the baseline experiment, we use the global linear mapping to synthesize virtual cluster centers. Then we use these virtual cluster centers to classify unseen data directly, which is denoted as Base.-Syn.-Cen. in Table2.

As shown in Table2, the classification accuracies using synthesized cluster centers without EM step (denoted as Syn.-Cen.) are 73.39% on AwA and 59.94% on CUB (using A+W label embeddings). This result outperforms that of the global linear mapping (Base.-Syn.-Cen.) with the improvement of 1.15% on AwA and 10.88% on CUB. It is observed that the gap between Syn.-Cen. and Base.-Syn.-Cen. is wider on the larger scale dataset (CUB). The reason is that the estimated manifold structure is more reliable on larger datasets which have denser data.

According to Table3, the inductive version of our method (Ours_I) outperforms state-of-the-art method (RKT) on both AwA and CUB. This result also verifies the effectiveness of our manifold transfer assumption.

3.4. Evaluation of the EM Optimization

Here, we evaluate the gain brought by the EM optimization (shown in Table2). All data (features, label embeddings, random splits) are consistent with those in the previous subsection. GMM with diagonal $\mathbf{\Sigma}_k$ (GMM-EM-D) and unit $\mathbf{\Sigma}_k$ (GMM-EM-U) are tested. For AwA, GMM-EM-U brings about 13% improvement of classification accuracy using the three label embeddings on average. Using GMM-EM-D increases nearly 1% classification accuracy over the GMM-EM-U. For CUB, nearly 6% improvement is brought by using GMM-EM-U. The experiment using GMM-EM-D on CUB is not reported due to the lack of training data (about 60 data in each class, which is explained in Sec.2.3.1). These results show that the transductive EM optimization improves classification performances in different settings.

	Label Embedding	Acc. % of Syn.-Cen.	Acc. % of GMM-EM-U	Acc. % of GMM-EM-D	Acc. % of Base.-Syn.-Cen.	Acc. % of Base.-GMM-EM-U
AwA	A	74.60	85.23	86.21	59.05	75.92
	W	60.99	75.31	76.31	61.02	73.61
	A+W	73.39	86.14	87.11	72.24	84.92
CUB	A	56.21	61.27	-	37.18	46.40
	W	47.31	55.62		37.38	42.66
	A+W	59.94	63.37		49.06	55.73

Table 2. Evaluate the synthesized virtual cluster centers with and without the EM optimization algorithm under the 300-random-split setting. Syn.-Cen. denotes classification directly using the synthesized virtual cluster centers. GMM-EM-D and GMM-EM-U are two regularization methods that use diagonal and unit $\mathbf{\Sigma}_k$ in the EM step. Base.-Syn.-Cen. and Base.-GMM-EM-U are two baseline experiments.

	ESZSL[17]	RKT[23]	Ours_I	Ours
AwA	79.53	81.41	83.24	96.00
CUB	51.90	55.59	57.31	60.24

Table 3. Performance (%) compared to state-of-the-art methods on the default splits by repeating their experiments. Ours_I is the inductive version of our method.

In the baseline experiment (Base.-GMM-EM-U), EM optimization is initialized by the virtual cluster centers synthesized by the global linear mapping. The classification accuracies of GMM-EM-U are 1.22% and 7.64% higher than those of Base.-GMM-EM-U on AwA and CUB respectively. The reason is that the performance of local optimization relies heavily on the initialization and our method can estimate more accurate virtual cluster centers. This result illustrates that the performance improvement of our method relay on both the estimation of virtual cluster centers and the transductive EM optimization.

3.5. Comparison to the State-of-the-Art

We repeat experiments of two state-of-the-art inductive methods, namely ESZSL [17] and RKT [23], using provided codes and the same data (including image features, label embeddings) as the aforementioned in Sec.3.3. Although we have to reduce image feature dimensions in our method, we use the original image features for their methods. We compare to these two methods using the widely used default split of each dataset.

As shown in Table3, our inductive method (Ours_I) outperforms RKT with 1.83% and 1.72% improvement on AwA and CUB. By leveraging unlabeled data, our transductive method (Ours) outperforms RKT with 14.59% and 4.65% improvement on AwA and CUB respectively.

We also compare to the reported results of state-of-the-art methods including both inductive and transductive methods. Inductive methods include DAP/IAP [9], VSAR [12], SJE [1], SC_struct [3], LatEm [24] and JLSE [26]. Transductive methods include UDA [7], SSE [25], TMV-HLP [4] and SP-ZSR [27].

From Table4, it can be seen that our method (Ours_I) achieves the best performance on both AwA and CUB datasets compared to those inductive methods. The performance (**83.24%** on AwA and **57.31%** on CUB) achieved by Ours_I is 2.78% and 7.21% higher than that of the runner-up methods (JLSE and SJE) respectively.

	Methods	Split	AwA (%)	CUB (%)
Inductive	DAP/IAP[9]	Default	41.4/42.2	-
	VSAR[12]		51.75	-
	SJE[26]		66.7	50.1
	SC_struct[3]		72.9	-
	LatEm[24]		76.1	47.4
	JLSE[26]		80.46	42.11
	Ours_I		**83.24**	**57.31**
Transductive	UDA[7]		75.6	40.6
	SSE[25]		76.33	30.41
	TMV-HLP[4]		80.5	47.9
	SP-ZSR[27]		92.08	55.34
	Ours		**96.00**	**60.24**

Table 4. Comparison to reported results of the state-of-the-art methods on the default splits.

In the transductive setting, our method (Ours) achieves the highest classification accuracies on two datasets, namely, **96.00%** on AwA and **60.24%** on CUB. This result is 3.92% and 4.90% higher than that of the runner-up method SP-ZSR (92.08% on AwA and 55.34% on CUB). Overall, by leveraging unlabeled data, transductive setting brings significant performance improvement based on inductive setting. The improvement on AwA is more remarkable than that on CUB. The reason is that the unlabeled data are used only for local optimization and the average inductive performance on AwA is much higher than that on CUB.

4. Conclusion

In this paper, we propose a transductive ZSL method based on the estimation of data distribution by posing ZSL as a missing data problem. We focus on exploiting the manifold structure in two spaces rather than the global mapping. Experiments show that our method outperforms other state-of-the-art methods on two popular datasets.

References

[1] 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.

[2] Z. Al-Halah, M. Tapaswi, and R. Stiefelhagen. Recovering the missing link: Predicting class-attribute associations for unsupervised zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5975–5984, 2016.

[3] 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.

[4] Y. Fu, T. M. Hospedales, T. Xiang, Z. Fu, and S. Gong. Transductive multi-view embedding for zero-shot recognition and annotation. In *European Conference on Computer Vision*, pages 584–599. Springer, 2014.

[5] 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.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[7] 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.

[8] 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.

[9] 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.

[10] J. Lei Ba, K. Swersky, S. Fidler, et al. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4247–4255, 2015.

[11] X. Li, Y. Guo, and D. Schuurmans. Semi-supervised zero-shot classification with label representation learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4211–4219, 2015.

[12] Y. Long, L. Liu, and L. Shao. Attribute embedding with visual-semantic ambiguity removal for zero-shot learning. In *Proc. Brit. Mach. Vis. Conf.*, 2016.

[13] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[14] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

[15] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In *Advances in neural information processing systems*, pages 1410–1418, 2009.

[16] R. Qiao, L. Liu, C. Shen, and A. van den Hengel. Less is more: zero-shot learning from online textual documents with noise suppression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2249–2257, 2016.

[17] B. Romera-Paredes and P. Torr. An embarrassingly simple approach to zero-shot learning. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 2152–2161, 2015.

[18] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.

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

[20] 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.

[21] 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.

[22] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[23] D. Wang, Y. Li, Y. Lin, and Y. Zhuang. Relational knowledge transfer for zero-shot learning. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

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

[25] 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.

[26] 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.

[27] Z. Zhang and V. Saligrama. Zero-shot recognition via structured prediction. In *European Conference on Computer Vision*, pages 533–548. Springer, 2016.