A General Transductive Regularizer for Zero-Shot Learning

Huaqi Mao[1]
maohuaqi@njust.edu.cn
Haofeng Zhang[1]
zhanghf@njust.edu.cn
Yang Long[2]
yang.long@ieee.org
Shidong Wang[3]
shidong.wang@uea.ac.uk
Longzhi Yang[4]
longzhi.yang@northumbria.ac.uk

[1] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China.
[2] The Openlab, School of Computing, Newcastle University, UK.
[3] School of Computing Sciences, University of East Anglia, UK.
[4] Department of Computer and Information Science, Northumbria University, UK.

Abstract

Zero Shot Learning (ZSL) has attracted much attention due to its ability to recognize objects of unseen classes, which is realized by transferring knowledge from seen classes through semantic embeddings. Since the seen classes and unseen classes usually have different distributions, conventional inductive ZSL often suffers from the domain shift problem. Transductive ZSL is a type of method for solving such a problem. However, the regularizers of conventional transductive methods are different from each other, and cannot be applied to other methods. In this paper, we propose a General Transductive Regularizer (GTR), which assigns each unlabeled sample to a fixed attribute by defining a Kullback-Leibler Divergence (KLD) objective. To this end, GTR can be easily applied to many compatible linear and deep inductive ZSL models. Extensive experiments on both linear and deep methods are conducted on four popular datasets, and the results show that GTR can significantly improve the performance comparing to its original inductive method, and also outperform some state-of-the-art methods, especially the extension on deep model.

1 Introduction

In recent years, image classification has achieved tremendous improvement, especially after the emergence of deep learning technology [15, 20]. Some methods on ImageNet [5] have achieved over 95% on top-5 accuracy [11], which is considered to exceed the level of human beings. However, the usage of these methods is limited by a very important constraint that the test data must come from the same classes as the training data. Due to the fact that there are billions of species over the world, and thousands of new objects emerge everyday, thus, it is impossible to include all the classes in a single training model. Fortunately, Zero Shot Learning (ZSL) [30, 31] is such a type of method proposed to solve this problem.

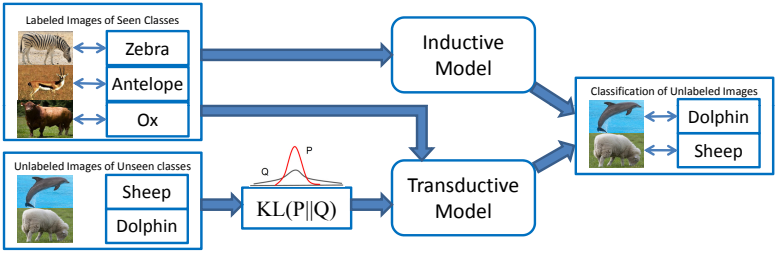


Figure 1: An illustration of difference between inductive ZSL and transductive ZSL (extended with the proposed GTR).

ZSL is inspired by the behavior of our human beings that when we meet new categories, we often utilize some auxiliary intermediate information, *e.g.*, predefined descriptions, to construct connections between seen and unseen categories. Therefore, in the field of ZSL, we similarly employ semantic vectors, *e.g.*, attributes annotated by experts, as intermediate information to achieve our purpose of recognizing novel categories. In the past decade, ZSL has made great success, but there is still a problem that has not been well solved. Since we all know that the distribution of seen data often differs from that of unseen data, thus just exploiting the model learned with only seen data will inevitably lead to the domain shift problem [7], *i.e.*, if the projection model from visual feature to semantic attribute is learned only from seen classes, the projection of unseen class image is likely to be shifted due to the bias distribution of the training on seen classes. Sometimes this bias might be far away from the correct unseen class prototype, and lead to error of the subsequent nearest neighbor search.

Many methods have been proposed to solve the domain shift problem. For example, Kodirov *et al.* [14] tried to exploit an additional decoder constraint to reconstruct the original visual feature. Jiang *et al.* proposed a Coupled Dictionary Learning (CDL) method [12] to align the visual-semantic structures using the class prototypes of both seen and unseen classes in a latent space. Many other inductive methods also make efforts to solve this problem [28, 29]. Although these methods can somewhat mitigate the domain shift problem, the performance is limited due to the true distribution of unseen samples is still unavailable. Therefore, the best way to solve this problem is to include unlabeled unseen data into training, which is called transductive ZSL.

Transductive ZSL is first defined by Fu *et al.* [8], who learned a multi-label regression to generalize a model to unseen classes by utilizing both labeled seen and unlabeled unseen data, which is illustrated in the bottom part of Fig. 1. Guo *et al.* proposed a joint learning approach which learns the Shared Model Space (SMS) [10] for models such that the knowledge can be transferred between classes using the attributes. Unsupervised Domain Adaptation (UDA) [13] formulates a regularized sparse coding framework which exploits the unseen class labels' projection in the semantic space to regularize the learned unseen class projection. Song *et al.* proposed an end-to-end transductive deep model, called Quasi-Fully Supervised Learning (QFSL) [21] to solve the bias problem. In QFSL, the labeled seen samples are projected to several fixed points specified by the source categories, and the unlabeled unseen samples are compelled to be projected to other points specified by the target categories. Although these tranductive methods have make great success in solving the domain shift problem, but there still exists a problem that these transductive regularizers are designed for themselves and cannot be extended to other methods, which seriously limits

the usage of them.

In this paper, we propose a General Transductive Regularizer (GTR), which defines a Kullback-Leibler Divergence (KLD) objective to force the soft assignment of the data of unseen classes to be similar as an auxiliary target distribution, *i.e.*, the predefined attributes or the word embeddings. Since our GTR does not employ any dedicated strategy, it is very convenient to be extended to other inductive methods, especially the compatible models. To verify the effectiveness of our GTR, we extend it on two popular linear models, including SAE [14], Label-Embedding with Attribute (ALE) [2], and a deep model. Extensive experiments are conducted on four popular dataset, and the results show that our GTR can be easily integrated into the inductive models and significantly improve the performance of classification. In addition, the experimental results also show that the extension on deep model can outperform the state-of-the-art methods. The contributions of our method are mainly in the following three aspects,

- In order to solve the domain shift problem, we propose a General Transductive Regularizer (GTR), which defines a KLD objective to force the soft assignment of the unlabeled unseen data to be similar as an auxiliary target distribution;
- The proposed GTR is independent of original inductive methods, so it can be easily extended to many compatible ZSL methods, such as SAE and ALE;
- Extensive experiments on four popular datasets prove that GTR can significantly improve the classification accuracy of the original inductive ZSL methods. Especially, the extended deep model can outperform the state-of-the-art methods by a large margin in most circumstances.

2 Related Work

Since our method only focuses on solving the domain shift problem by designing tansductive regularizer, we only briefly review some researches on domain shift and its corresponding methods with inductive and transductive settings.

Domain shift The domain shift problem is first identified by Fu *et al.* [8]. This problem is explained as following: since the seen classes and the unseen classes are disjoint, the underlying data distribution of them are usually different. The learned projection functions only on the dataset of seen classes from visual space to attribute space without any adaption to the unseen classes, will inevitably lead to bad generalization on them.

Inductive ZSL Since the domain shift problem is the main reason for the performance degradation in inductive ZSL, many researchers have make great efforts to mitigate it. SAE [14] is one of the representative inductive method, which tries to add an additional decoder to constrain the reconstruction from attributes to original visual features. Zhang *et al.* [28] built a deep network with triple verifications to crossly reconstruct both attributes and original features, and add an orthogonal constraint in third space to make the latent embeddings more discriminative. In recent year, Generative Adversarial Network (GAN) [9] has made great success in data synthesis, thence an increasing number of GAN based ZSL methods have been proposed [29]. These methods utilize GAN to synthesize new samples of unseen classes from the corresponding attributes, and then train a fully supervised classifier with both the synthesized data of unseen classes and the labeled data of seen classes. However, these inductive methods cannot capture the real distribution of the unseen classes, they are doomed to be unable to solve this problem completely.

Transductive ZSL Due to the fact that inductive ZSL cannot fully solve the domains shift problem, transductive setting is first proposed by Fu *et al.* [7], who trained a multi-label regression to generalize a model to unseen classes by using both labeled seen data and unlabeled unseen data. Yu *et al.* [27] formulated the class prediction problem in an iterative refining process, where the object classification capacity is progressively reinforced through bootstrapping-based model updating over highly reliable instances. Quasi-Fully Supervised Learning (QFSL) [21] is another type of transductive method, it trains an end-to-end deep network to assign each unlabeled data with a fixed target category. Although these methods can well target the domain shift problem, the transductive regularizer of them are designed specially, and cannot be extended to other inductive models.

3 Methodology

3.1 Problem Definition

Given a dataset $\mathcal{D}$, which is consisted of two separate parts, the seen classes $\mathcal{S}$ and the unseen classes $\mathcal{U}$, where, $\mathcal{S} = \{1, \cdots, s\}$, $\mathcal{U} = \{s+1, \cdots, s+u\}$, and $\mathcal{S} \cap \mathcal{U} = \varnothing$. In addition, each class of $\mathcal{S}$ and $\mathcal{U}$ is associated with an auxiliary attribute, and they are $\boldsymbol{A}_s \subset \mathcal{R}^{d_a \times s}$ and $\boldsymbol{A}_u \subset \mathcal{R}^{d_a \times u}$ respectively, where d_a is the dimension of the attribute. Let $\boldsymbol{X}^s = \{\boldsymbol{x}_1^s, \cdots, \boldsymbol{x}_i^s, \cdots, \boldsymbol{x}_{N_s}^s\} \subset \mathbb{R}^{d_x \times N_s}$ denotes the samples from the seen classes $\mathcal{S}$, where N_s is the number of samples, and each sample $\boldsymbol{x}_i^s$ is labeled with a single class in $\mathcal{S}$. Similarly, let $\boldsymbol{X}^u = \{\boldsymbol{x}_1^u, \cdots, \boldsymbol{x}_j^u, \cdots, \boldsymbol{x}_{N_u}^u\} \subset \mathbb{R}^{d_x \times N_u}$ denotes the samples from the unseen classes $\mathcal{U}$, where N_u is the number of samples, but the labels of $\boldsymbol{X}^u$ are totally unknown. In transductive setting, the goal is to find a classifier $F(\boldsymbol{x}_j^u)$ to assign each sample $\boldsymbol{x}_j^u$ with a category from the unseen classes $\mathcal{U}$ by assuming $\boldsymbol{X}^s$, $\boldsymbol{A}^s$, $\boldsymbol{X}^u$, and $\boldsymbol{A}^u$ are all accessible during training.

3.2 General Tansductive Regularizer (GTR)

Since the data of unseen classes is unlabeled, it is unable to directly use a supervised strategy to learn its projection function from visual space to attribute space. Xie *et al.* [26] proposed to use an iterative method for data clustering, which inspires us to define an iterative assignment for unlabeled unseen data too. Here, we propose to iteratively refine the projection function by learning from the high confidence assignment with the assist of an auxiliary target distribution. Concretely, our model is learned by matching the soft assignment to the target distribution. To this end, we define our GTR as a KLD loss between the soft assignments Q and the auxiliary distribution P,

$$\mathcal{L}_G = KL(P||Q) = \sum_{i}^{N_u}\sum_{j}^{u} p_{ij} \log \frac{p_{ij}}{q_{ij}}, \tag{1}$$

where q_{ij} is the similarity between projected data point $\boldsymbol{z}_i^u$ and the attribute $\boldsymbol{a}_j^u$ measured by Student's t-distribution [18]:

$$q_{ij} = \frac{(\varepsilon + \|\boldsymbol{z}_i^u - \boldsymbol{a}_j^u\|^2)^{-1}}{\sum_j (\varepsilon + \|\boldsymbol{z}_i^u - \boldsymbol{a}_j^u\|^2)^{-1}}, \tag{2}$$

where, ε is a small value to guarantee the denominator should not be zero, and $\boldsymbol{z}_i^u = g(\boldsymbol{x}_i^u)$ is the projected data point in attribute space from $\boldsymbol{x}_i^u$. In linear model, $g(\boldsymbol{x}_i^u) = \boldsymbol{W}\boldsymbol{x}_i^u$, where $\boldsymbol{W}$ is the linear projection matrix, and in deep model, $g(\boldsymbol{x}_i^u)$ is a multi-layer neural network.

The selection of target distributions P is critical for the final performance. According to [26], the target distribution should have the following characteristics: (1) it should be able to strengthen prediction; (2) it can put more emphasis on the data assignment with high confidence; and (3) normalize loss contribution of each class assignment to prevent large one from distorting the attribute space. Therefore, p_i is computed by first raising q_i to the second power and then normalized per class,

$$p_{ij} = \frac{q_{ij}^2 / \sum_i q_{ij}}{\sum_{j'} (q_{ij'}^2 / \sum_i q_{ij'})}. \tag{3}$$

In our method, we try to solve the projection function with iterative mode, so we should first compute the gradient of $\mathcal{L}_G$. For linear model, the gradient of $\mathcal{L}_G$ with respect to the projection matrix $\boldsymbol{W}$ can be realized by the chain rule,

$$\frac{\partial \mathcal{L}_G}{\partial \boldsymbol{W}} = \frac{\partial \mathcal{L}_G}{\partial \boldsymbol{z}_i^u} \frac{\boldsymbol{z}_i^u}{\boldsymbol{W}} = 2 \sum_{i=1}^{N_u} \sum_{j=1}^{u} (\varepsilon + \|\boldsymbol{W}\boldsymbol{x}_i^u - \boldsymbol{a}_j^u\|^2)^{-1} (p_{ij} - q_{ij}) (\boldsymbol{W}\boldsymbol{x}_i^u - \boldsymbol{a}_j^u) \boldsymbol{x}_i^{uT}. \tag{4}$$

3.3 Transductive extensions on linear models

Since we have claimed that our GTR can be easily extended to compatible inductive ZSL methods, we select two models, SAE and ALE, to explain how to employ our GTR on them, and turn them into tranductive setting.

3.3.1 SAE

SAE is a representative inductive model, it exploits an encoder to project visual feature into attribute space and a decoder to reconstruct visual feature from its corresponding attribute, which can be represent as,

$$\mathcal{L}_{SI} = \|\boldsymbol{W}\boldsymbol{X}^s - \boldsymbol{A}^s\|_F^2 + \alpha_1 \|\boldsymbol{W}^T\boldsymbol{A}^s - \boldsymbol{X}^s\|_F^2, \tag{5}$$

where, α_1 is the balancing coefficient for the two items. Therefore, the transductive setting of SAE can be easily realized by adding the GTR,

$$\mathcal{L}_{ST} = \|\boldsymbol{W}\boldsymbol{X}^s - \boldsymbol{A}^s\|_F^2 + \alpha_1 \|\boldsymbol{W}^T\boldsymbol{A}^s - \boldsymbol{X}^s\|_F^2 + \beta_1 KL(P||Q). \tag{6}$$

where, β_1 is the balancing coefficient for $\mathcal{L}_{SI}$ and $\mathcal{L}_G$. The gradient of $\mathcal{L}_{ST}$ with respect to $\boldsymbol{W}$ can be represented as,

$$\begin{aligned} \frac{\partial \mathcal{L}_{ST}}{\partial \boldsymbol{W}} =& \boldsymbol{W}\boldsymbol{X}^s\boldsymbol{X}^{sT} - \boldsymbol{A}^s\boldsymbol{X}^{sT} + \alpha_1 (\boldsymbol{A}^s\boldsymbol{A}^{sT}\boldsymbol{W} - \boldsymbol{A}^s\boldsymbol{X}^{sT}) \\ &+ \beta_1 \sum_{i=1}^{N_u} \sum_{j=1}^{u} (\varepsilon + \|\boldsymbol{W}\boldsymbol{x}_i^u - \boldsymbol{a}_j^u\|^2)^{-1} (p_{ij} - q_{ij}) (\boldsymbol{W}\boldsymbol{x}_i^u - \boldsymbol{a}_j^u) \boldsymbol{x}_i^{uT}. \end{aligned} \tag{7}$$

Therefore, the iterative process can be obtained by using the following Gradient Descent (GD) strategy,

$$\boldsymbol{W}^{(t)} = \boldsymbol{W}^{(t-1)} - \lambda_1 \frac{\partial \mathcal{L}_{ST}}{\partial \boldsymbol{W}}, \tag{8}$$

where, λ_1 is the learning rate, and the $\boldsymbol{W}^{(t)}$ is the t^{th} iterative result.

3.3.2 ALE

Label Embedding with Attribute (ALE) [2, 3] is another typical compatible inductive model, which tries to learn the projection with a max margin loss. The loss function of ALE is represented as,

$$\mathcal{L}_{AI} = \sum_{i=1}^{N_s} \sum_{j=1\&j\neq\ell(\boldsymbol{x}_i^s)}^{s+u} \max\{0, \alpha_2 + \boldsymbol{x}_i^{sT}\boldsymbol{W}^T(\boldsymbol{a}_j^s - \boldsymbol{a}_{\ell(\boldsymbol{x}_i^s)}^s)\} + \frac{\gamma_2}{2}\|\boldsymbol{W}\|_F^2, \tag{9}$$

where, $\ell(\boldsymbol{x}_i^s)$ means the label of $\boldsymbol{x}_i^s$, α_2 represents for the value of max margin, and γ_2 stands for the balancing coefficient. Therefore, the transductive setting of ALE can be easily realized by adding the GTR to $\mathcal{L}_{AI}$,

$$\mathcal{L}_{AT} = \sum_{i=1}^{N_s} \sum_{j=1\&j\neq\ell(\boldsymbol{x}_i^s)}^{s+u} \max\{0, \alpha_2 + \boldsymbol{x}_i^{sT}\boldsymbol{W}^T(\boldsymbol{a}_j^s - \boldsymbol{a}_{\ell(\boldsymbol{x}_i^s)}^s)\} + \frac{\beta_2}{2}KL(P||Q) + \frac{\gamma_2}{2}\|\boldsymbol{W}\|_F^2, \tag{10}$$

where, β_2 is the balancing coefficient for the two items. The gradient of $\mathcal{L}_{AT}$ with respect to $\boldsymbol{W}$ can be represented as,

$$\begin{aligned}\frac{\partial \mathcal{L}_{AT}}{\partial \boldsymbol{W}} =& \sum_{i=1}^{N_s} \sum_{j=1\&j\neq\ell(\boldsymbol{x}_i^s)}^{s+u} \mathbb{1}(\alpha_2 + \boldsymbol{x}_i^{sT}\boldsymbol{W}^T(\boldsymbol{a}_j^s - \boldsymbol{a}_{\ell(\boldsymbol{x}_i^s)}^s) > 0)(\boldsymbol{a}_j^s - \boldsymbol{a}_{\ell(\boldsymbol{x}_i^s)}^s)\boldsymbol{x}_i^{sT} \\ &+ \beta_2 \sum_{i=1}^{N_u}\sum_{j=1}^{u} (\varepsilon + \|\boldsymbol{W}\boldsymbol{x}_i^u - \boldsymbol{a}_j^u\|^2)^{-1}(p_{ij} - q_{ij})(\boldsymbol{W}\boldsymbol{x}_i^u - \boldsymbol{a}_j^u)\boldsymbol{x}_i^{uT} + \gamma_2\boldsymbol{W}\end{aligned} \tag{11}$$

where, $\mathbb{1}(\cdot)$ is an indicator function, when the condition is satisfied, the result is 1, otherwise 0. Similar as Eq. 8, Eq. 10 can also be optimized with an iterative GD,

$$\boldsymbol{W}^{(t)} = \boldsymbol{W}^{(t-1)} - \lambda_2 \frac{\partial \mathcal{L}_{AT}}{\partial \boldsymbol{W}}, \tag{12}$$

where, λ_2 is the learning rate.

3.4 Transductive extension on deep model

To show the powerful generalization ability of our GTR, we also extend it to a deep model, and the architecture is illustrated in Fig. 2. Concretely, the input contains both the visual features and the attributes, where the features include both labeled seen data and unlabeled unseen data, and the attributes contain both seen and unseen classes. In addition, to be more convenient, we take the visual features extracted with ResNet-101 as input, which is followed by two fully connected layers, and each of them is attached with a nonlinear activation operation, *i.e.*, Rectified Linear Unit (ReLU). The final output of the network is the Softmax probability (replace Eq. 2) of the inner product of the visual branch result and the attributes,

$$q_{ij} = \frac{g(\boldsymbol{x}_i)^T\boldsymbol{a}_j}{\sum_{j=1}^{s+u} g(\boldsymbol{x}_i)^T\boldsymbol{a}_j}, \tag{13}$$

where, $\boldsymbol{a}_j \in \boldsymbol{A}^s \cup \boldsymbol{A}^u$, and $\boldsymbol{x}_i \in \boldsymbol{X}^s \cup \boldsymbol{X}^u$.

The output is divided into two parts, the upper part is designed for the seen classes, and the bottom part is built for the unseen classes. For the data points of seen classes, we exploit

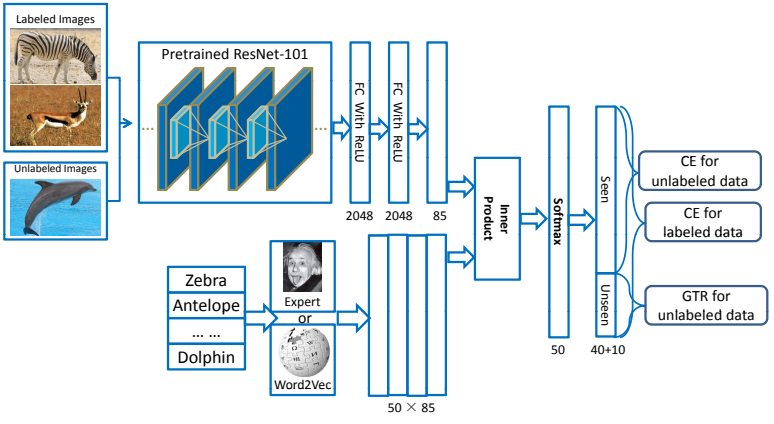


Figure 2: Illustration of the architecture of our deep network with GTR.

Table 1: The summarization of the four popular dataset used in our experiments.

Datasets	Attribute Dim	Samples	Seen/Unseen Class	Train	Test
SUN [19]	102	14,340	645/102	10,320	1,440
CUB [19]	312	11,788	150/50	7,057	2,967
AWA [16]	85	30,475	40/10	19,832	5,685
aPY [6]	64	15,339	20/12	5,932	7,924

the Cross Entropy (CE) loss to assign each one with a ground truth label specified with the seen classes. For the data samples of unseen classes, it is also assigned a label softly with CE, and constrained by GTR loss, which can be represented as,

$$\mathcal{L}_{DT} = -\sum_{i=1}^{N_s}\sum_{j=1}^{s+u}(\ell_{ij} log q_{ij} + (1-\ell_{ij}) log(1-q_{ij})) - \sum_{i=1}^{N_u}\sum_{j=1}^{s} log(1-q_{ij}) + \beta_3 KL(P||Q) \tag{14}$$

where, ℓ_{ij} is the j^{th} entry of one-hot label of i^{th} sample, β_3 is the balancing coefficient. Since the three items in Eq. 14 are all differentiable, it can be easily optimized with Stochastic Gradient Descent (SGD) in Tensorflow [1].

4 Experiments

4.1 Datasets and settings

To verify the effectiveness of our approach, we conduct experiments on four popular datasets, including SUN attribute (SUN) [19], Caltech-UCSD Birds-200-2011 (CUB-200) [24], Animal with Attribute (AwA) [16], and a Pascal & Yahoo attribute (aPY) [6]. The dataset splits for training and testing follow that used in [25], and the details of the four datasets are recorded in Tab. 1.

During our experiments, we use the 2048 dimensional features extracted with ResNet-101 as the input, and the same attributes employed in the evaluation in [25]. In addition, there are six hyper-parameters α_1, β_1, α_2, β_2, γ_2, and β_3 for the three extended methods. Since γ_2 is only for the regularization of $\boldsymbol{W}$, we set it with a small value 1×10^{-3}. The other

Table 2: Comparison with state-of-the-art ZSL baselines on both inductive setting and transductive settings. '$\mathcal{I}$' stands for inductive setting, and '$\mathcal{T}$' represents for transductive setting.

Method	Setting	SUN	CUB	AWA	aPY	Average
QFSL [21]	$\mathcal{T}$	63.7	56.2	60.4	38.6	54.7
GFZSL-Trans [22]	$\mathcal{T}$	59.4	45.2	**74.7**	35.9	53.8
VZSL-Trans [23]	$\mathcal{T}$	57.6	49.3	69.1	35.7	52.9
SAE [14]	$\mathcal{I}$	53.4	36.0	58.1	32.9	45.1
SAE+GTR	$\mathcal{T}$	59.0	39.9	71.4	41.5	53.0
ALE [2]	$\mathcal{I}$	58.1	54.9	59.9	39.7	53.2
ALE+GTR	$\mathcal{T}$	59.7	55.7	66.8	43.0	56.3
Deep-CE	$\mathcal{I}$	61.0	55.5	63.9	39.2	54.9
Deep-CE+GTR	$\mathcal{T}$	**66.3**	**60.3**	72.5	**44.0**	**60.8**

five hyper-parameters are fine-tuned in the set of $\{0.001, 0.01, 0.1, 1, 10, 100\}$ with a cross validation. In this experiment, the cross validation means splitting 20% of the seen classes as the validation unseen classes, and both the labeled seen data and the unlabeled validation data are employed in training to find the optimal parameters. To be more robust, the optimal parameters are the selections from highest value of average on 5 executions, and they are $\alpha_1 = 0.01$, $\beta_1 = 1$, $\alpha_2 = 0.1$, $\beta_2 = 1$, and $\beta_3 = 1$ for all four datasets. In addition, there are three learning rates for the three iterative methods, including SAE+GTR, ALE+GTR, and Deep-CE+GTR, and they are set with the same value $\lambda_1 = \lambda_2 = \lambda_3 = 1 \times 10^{-3}$, where, λ_3 is the learning rate of the deep model.

4.2 Comparison with baselines

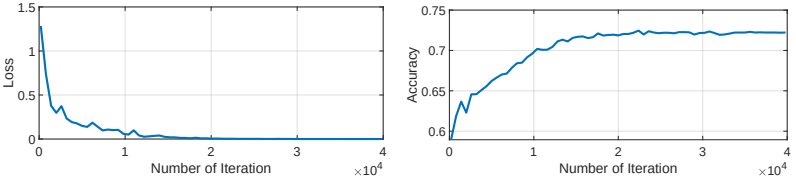
We compare our method with 6 recently proposed inductive and transductive methods. The inductive methods contain SAE [14], ALE [2], and Deep-Cross Entropy (Deep-CE) [17], which are extended by our GTR. The transductive methods include QFSL, Transdutive General Framework for ZSL (GFZSL-Trans) [22], and Transductive Variational ZSL (VZSL-Trans) [23]. Since Generalized ZSL (GZSL) setting assumes that the ascription of test data is not known for seen classes or unseen classes, while transductive ZSL knows that in advance. Therefore, the transductive setting is not suitable for GZSL and we only focus our method on conventional ZSL here. All these transductive methods are implemented by us according to the description in their original papers, and the results are recorded in Tab. 2. The inductive results in Tab. 2 are directly cited from [25].

From this table, it can be obviously found that the extensions with our GTR can be significantly improve the original inductive methods. Concretely, for SAE, we can obtain 5.6%, 3.9%, 13.3%, and 8.6% improvement on SUN, CUB, AWA, and APY respectively; for ALE, 1.6%, 0.8%, 6.9%, and 3.3% improvement on SUN, CUB, AWA, and APY respectively are achieved. For the deep model, we implement the cross entropy based classification loss with deep architecture, and extend it with our GTR. The results show that it can get 4.7%, 4.8%, 8.4%, and 4.8% improvement on SUN, CUB, AWA, and APY respectively. In addition, we also compare our method with the state-of-the-art transductive methods QFSL, GFZSL-Trans and VZSL-Trans. The results show that our Deep-CE+GTR can outperform the best methods on SUN, CUB and aPY respectively, except a little worse than GFZSL-Trans on AWA. However, we argue that a good method should perform well on all datasets rather than just on a single one, thus we also calculate the average performance on all four datasets, and record the results in the last column in Tab. 2. According to the average values, we can

clearly discover that our method can outperform the state-of-the-art transductive methods by a large margin, which shows the superiority of our method.

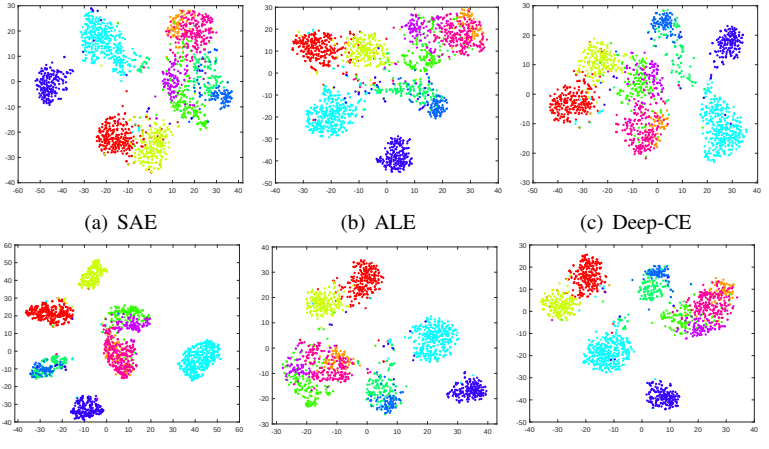
4.3 Detailed analysis

Convergence analysis Since the optimization of the transductive extension with GTR is an iterative process, it is necessary to analyze the convergence of it. In this experiment, we take AWA as an example, and draw the curves of loss function and classification accuracy of Deep-CE+GTR. The results are shown in Fig. 3, from which we can clearly find that when the iteration number equals 2×10^4, the loss function converges, and the accuracy achieve the highest. This phenomenon indicates that the iterative optimization strategy for the transductive extension with GTR is convergent and feasible.



(a) Loss (b) Accuracy

Figure 3: Illustrating the convergence curves of loss function and accuracy of Deep-CE+GTR on AWA.



(a) SAE (b) ALE (c) Deep-CE

(d) SAE+GTR (e) ALE+GTR (f) Deep-CE+GTR

Figure 4: Illustrations of the projected data point of unseen classes in attribute space on AWA with t-SNE [13]. Best viewed in color.

Distribution in attribute space The difference between the distributions of seen classes and unseen classes causes the domain shift problem in inductive ZSL methods, so the projection function trained with only labeled data of seen classes is surely not suitable for the unseen classes. Since we have claimed that GTR can well solve the domain shift problem, the projection function learned with extended transductive models should be appropriate for the unseen classes. We illustrate the projected samples of unseen classes in attribute space in Fig. 4, from which it is clearly observed that the distributions with GTR are more discriminative than those without. Therefore, we can say that the projection function learned from tranductive extension with GTR is effective.

Table 3: Time consumption of Deep-CE+GTR for training and testing on four datasets.

Datasets	Train	Test
SUN [19]	698.3s	0.01s
CUB [19]	422.8s	0.02s
AWA [16]	396.1s	0.01s
aPY [6]	397.1s	0.01s

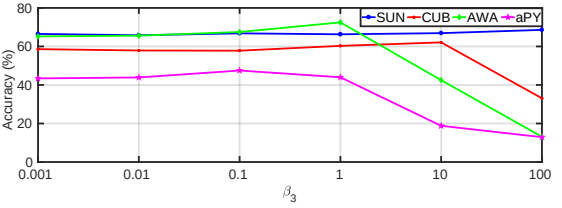


Figure 5: Accuracy curves of Deep-CE+GTR on four datasets with respect to different β_3.

Time consumption analysis Since efficiency is very important for the availability of our model in real application, we record the time consumptions of Deep-CE+GTR on four datasets in Tab. 3. It is obviously discovered that the training time is on the level of $10^2 s$, and the test time is on the level of $10^{-2} s$, which indicates that our model can be conducted in real time and is available in realistic scenarios. Besides, we also can find that the dataset SUN consumes the most time during training, because it has the most categories and needs most time to compute the KL Divergence, while in testing, the dataset CUB spends the most time to calculate the category, because it has longest attribute vector.

Hyper-parameter analysis For Deep-CE+GTR, there is only one hyper-parameter β_3 that can affect the final result, so we here conduct experiments to show how it can affect the performance and illustrate the results in Fig. 5. From this figure, we can clearly find that when $\beta_3 \leqslant 10$ on CUB and when $\beta_3 \leqslant 1$ on AWA and aPY, the performance does not change significantly. In addition, the best performance on AWA appears when $\beta_3 = 1$, which is consistent with our setting. Although we cannot achieve the best performance on other three datasets when $\beta_3 = 1$, the accuracies approximate the maximum values, which shows that the optimal hyper-parameters chosen by cross validation are feasible.

5 Conclusion

In this paper, a general tranductive regularizer, namely GTR, is proposed to solve the domain shift problem in conventional inductive ZSL methods. GTR utilizes a KL Divergence objective as its loss function to constrain each unlabeled sample of unseen classes to be associated with a fixed attribute. Since GTR is only related with the projection function from visual space to attribute space and independent of the original inductive models, it can be easily extended to many compatible inductive methods. In this paper, GTR is employed on two linear models and one deep models, and the experiments on four popular datasets demonstrate its effectiveness.

6 Acknowledgement

This work is supported by the National Natural Science Foundation of China (No.61872187) and the Natural Science Foundation of Jiangsu Province (No.BK20160842). The corresponding author is Haofeng Zhang.

References

[1] Martin Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, and Matthieu Devin. Tensorflow: A system for large-scale machine learning. In *OSDI*,

pages 265–283, 2016.

[2] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for attribute-based classification. In *CVPR*, 2013.

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

[4] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE TPAMI*, 38(7):1425–1438, 2016.

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

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

[7] Yanwei Fu, Timothy M Hospedales, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Transductive multi-view embedding for zero-shot recognition and annotation. In *ECCV*, pages 584–599. Springer, 2014.

[8] Yanwei Fu, Timothy M Hospedales, Tao Xiang, and Shaogang Gong. Transductive multi-view zero-shot learning. *IEEE TPAMI*, 37(11):2332–2345, 2015.

[9] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Xu Bing, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[10] Yuchen Guo, Guiguang Ding, Xiaoming Jin, and Jianmin Wang. Transductive zero-shot recognition via shared model space learning. In *AAAI*, pages 3494–3500, 2016.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[12] Huajie Jiang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Learning class prototypes via structure alignment for zero-shot recognition. In *ECCV*, 2018.

[13] Elyor Kodirov, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Unsupervised domain adaptation for zero-shot learning. In *ICCV*, pages 2452–2460, 2015.

[14] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. In *CVPR*, pages 3174–3183, 2017.

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2015.

[16] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009.

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

[18] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9 (11):2579–2605, 2008.

[19] Genevieve Patterson, Chen Xu, Hang Su, and James Hays. The sun attribute database: Beyond categories for deeper scene understanding. *IJCV*, 108(1-2):59–81, 2014.

[20] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[21] Jie Song, Chengchao Shen, Yezhou Yang, Yang Liu, and Mingli Song. Transductive unbiased embedding for zero-shot learning. In *CVPR*, pages 1024–1033, 2018.

[22] Vinay Kumar Verma and Piyush Rai. A simple exponential family framework for zero-shot learning. In *ECML-PKDD*, pages 792–808. Springer, 2017.

[23] Wenlin Wang, Yunchen Pu, Vinay Kumar Verma, Kai Fan, Yizhe Zhang, Changyou Chen, Piyush Rai, and Lawrence Carin. Zero-shot learning via class-conditioned deep generative models. In *AAAI*, 2018.

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

[25] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *CVPR*, 2017.

[26] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *ICML*, pages 478–487, 2016.

[27] Yunlong Yu, Zhong Ji, Xi Li, Jichang Guo, Zhongfei Zhang, Haibin Ling, and Fei Wu. Transductive zero-shot learning with a self-training dictionary approach. *IEEE Transactions on Cybernetics*, 48(10):2908–2919, 2018.

[28] Haofeng Zhang, Yang Long, Yu Guan, and Ling Shao. Triple verification network for generalized zero-shot learning. *IEEE Transactions on Image Processing*, 28(1): 506–517, 2019.

[29] Haofeng Zhang, Yang Long, Li Liu, and Ling Shao. Adversarial unseen visual feature synthesis for zero-shot learning. *Neurocomputing*, 329:12–20, 2019.

[30] Haofeng Zhang, Yang Long, and Ling Shao. Zero-shot hashing with orthogonal projection for image retrieval. *Pattern Recognition Letters*, 117:201–209, 2019.

[31] Haofeng Zhang, Yang Long, Wankou Yang, and Ling Shao. Dual-verification network for zero-shot learning. *Information Sciences*, 470:43–57, 2019.