Relational Knowledge Transfer for Zero-Shot Learning

Donghui Wang, Yanan Li, Yuetan Lin, Yueting Zhuang
College of Computer Science, Zhejiang University, Hangzhou, China
{dhwang, ynli, linyuetan, yzhuang}@zju.edu.cn

Abstract

General zero-shot learning (ZSL) approaches exploit transfer learning via semantic knowledge space. In this paper, we reveal a novel relational knowledge transfer (RKT) mechanism for ZSL, which is simple, generic and effective. RKT resolves the inherent semantic shift problem existing in ZSL through restoring the missing manifold structure of unseen categories via optimizing semantic mapping. It extracts the relational knowledge from data manifold structure in semantic knowledge space based on sparse coding theory. The extracted knowledge is then transferred backwards to generate virtual data for unseen categories in the feature space. On the one hand, the generalizing ability of the semantic mapping function can be enhanced with the added data. On the other hand, the mapping function for unseen categories can be learned directly from only these generated data, achieving inspiring performance. Incorporated with RKT, even simple baseline methods can achieve good results. Extensive experiments on three challenging datasets show prominent performance obtained by RKT, and we obtain **82.43%** accuracy on the Animals with Attributes dataset.

Introduction

Traditional machine learning approaches for classification presuppose the existence of a large labelled dataset to optimize the parameters of object classifiers. Formally, the task of traditional supervised learning is to find a function $f : \mathcal{X} \to \mathcal{Y}$ from the training dataset $\{(\mathbf{x}_i, y_i)|1 \leqslant i \leqslant N\}$, where $\mathcal{X}$ denotes the d-dimensional input space and $\mathcal{Y}$ contains all K labels in the training dataset. When each input $\mathbf{x}_i$ belongs to one class, learning f is a traditional multi-class classification problem. From the perspective of space geometry, it maps $\mathbf{x}_i$ to one of the K vertices of a $(K, 1)$-hypersimplex in the K-dimensional space, when y_i is encoded as a one-hot vector. However, if $\mathbf{x}_i$ comes from multiple classes, f becomes a multi-label (ML) classification function (Huang, Gao, and Zhou 2014), which maps $\mathbf{x}_i$ to one of the 2^K vertices of a unit hypercube in the K-dimensional space.

Zero-shot learning (ZSL) aims to learn a classifier $f_u : \mathcal{X} \to \mathcal{Z}, \mathcal{Y} \cap \mathcal{Z} = \emptyset$ for unseen categories from the given training dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, where $y_i \in \mathcal{Y}$ is the training class label and $\mathcal{Z}$ denotes the testing label space (Lampert, Nickisch, and Harmeling 2009). Obviously, because of the lack of testing data, f_u can not be directly learned. Though $f_s : \mathcal{X} \to \mathcal{Y}$ can be obtained from the training dataset, due to the symmetry of hypersimplex, f_s has no ability to transfer knowledge for f_u to predict unseen categories in ZSL. To tackle this problem, the common practice is to introduce a sharable semantic knowledge space $\mathcal{S}$, as is shown in Figure 1. Learning f_u becomes a two stage process $f_u = l(g_u(\mathbf{x}), z)$, i.e. first learn a semantic mapping $g_u(\mathbf{x}) : \mathbf{x} \to \phi, \phi \in \mathcal{S}$ by ML methods when ϕ is binary or regression models if ϕ is continuous. Then it learns a class predictor $l(\phi) : \phi \to z$. Likewise, we can not get $g_u(\mathbf{x})$ for unseen categories. Since the semantic space $\mathcal{S}$ is shared between $\mathcal{Y}$ and $\mathcal{Z}$, it is hoped that by resorting to $g_s(\mathbf{x})$, ZSL could be addressed. Thereby, f_u is learned as $l(g_s(\mathbf{x}), z)$ in existing ZSL approaches.

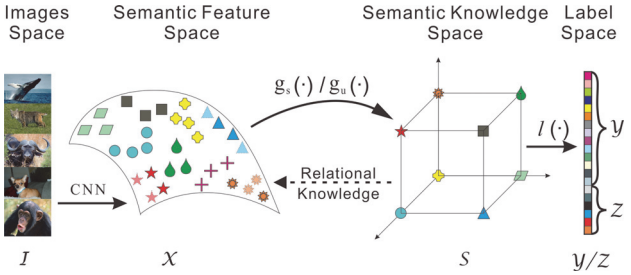

Figure 1: The overall framework of the proposed method for ZSL. First, it transfers the semantic correlation from $\mathcal{S}$ to $\mathcal{X}$; then it uses this correlation to restore the manifold structure of unseen categories by producing virtual labelled data; finally, it learns semantic mapping $g(\mathbf{x})$ for ZSL.

Nevertheless, approximating the real mapping $g_u(\mathbf{x})$ for unseen categories using $g_s(\mathbf{x})$ suffers an inherent problem. On one hand, $g_s(\mathbf{x})$ just optimize the training dataset where labelled information of unseen classes is missing. On the other hand, in real world situations the semantic relationship between different classes in $\mathcal{X}$ may be different from that in $\mathcal{S}$. Therefore, $g_s(\mathbf{x})$ has a shift from $g_u(\mathbf{x})$. Direct use of $g_s(\mathbf{x})$ in $f_u(\mathbf{x})$ instead of $g_u(\mathbf{x})$ will cause significant performance degradation in ZSL. Our main goal of this

work is to move towards this problem by taking advantage of above-mentioned geometric structure in $\mathcal{S}$.

To improve ZSL performance, several works focused on improving the semantic expression ability of $g_s(\mathbf{x})$. e.g. by jointly learning class labels and semantic embeddings (Akata et al. 2013). While other efforts were made to adopt novel classifiers for $l(\phi)$ to compensate for the less effectiveness of $g_s(\mathbf{x})$, such as absorbing Markov process (Fu et al. 2015b), label propagating (Rohrbach, Ebert, and Schiele 2013; Fu et al. 2015a) etc. However, all these above methods did not take into account the inherent problem in the process of transferring knowledge, i.e. $g_s(\mathbf{x})$ is still shifted from $g_u(\mathbf{x})$. In this paper, we study a relational knowledge transfer framework called RKT that is able to align $g_s(\mathbf{x})$ with $g_u(\mathbf{x})$ in two steps.

As illustrated in Figure 1, in the first step we extract the semantic correlation between unseen categories and training classes in $\mathcal{S}$ on sparse coding theory (Olshausen and Field 1997). Given the geometric structure, each unseen class is considered as locally linearly related to seen classes. Then, in the second step, we transfer this semantic correlation to help generate the manifold structure of unseen categories in $\mathcal{X}$. Under the proposed framework, ZSL performance can be improved in two different ways, i.e. by promoting the approximation ability of $g_s(\mathbf{x})$ to $g_u(\mathbf{x})$ or by directly learning $g_u(\mathbf{x})$ for unseen categories. Extensive experiments on several ZSL datasets show incorporating the proposed framework into baselines can achieve state-of-art performance.

The remainder of the paper is organized as follows. In the next section, we briefly review related methods for performing zero-shot learning. Then, we introduce our proposed method, followed by experimental results on several real world datasets. Finally, we draw conclusions.

Related Work

We briefly outline connections and differences to four related lines of research in ZSL.

Feature Spaces $\mathcal{X}$. For the past few years, deep semantic features have been proven effective for a variety of machine learning tasks, such as large scale image classification (Krizhevsky, Sutskever, and Hinton 2012), object detection (Girshick et al. 2014), attribute learning (Zhang et al. 2014; Luo, Wang, and Tang 2013) etc. Recently, latest ZSL approaches have also adopted various deep features for predict unseen categories. Comparing with low level features, they obtain more compelling results. In our work, we use two kinds of state-of-the-art deep features, extracted by VGG (Simonyan and Zisserman 2015) and GoogLeNet (Szegedy et al. 2014) for ZSL.

Semantic Spaces $\mathcal{S}$. In ZSL, there has been a body of work on the use of human-labelled visual attributes to help detect unseen object categories (Lampert, Nickisch, and Harmeling 2009; 2014). As an appealing source of information, attributes (binary or continuous) describe well known common properties of objects and can be acquired from domain experts or crowdsourced techniques (Xiao et al. 2010). However manually defining an attribute ontology is of high cost and long periodicity, leading to limit its application in large scale recognition.

As an alternative to manual annotation, automatically learning a vector representation for each class is gaining more and more attention. They are learned from a large external text corpus, e.g. Wikipedia, in an unsupervised fashion, based on an independent natural language modeling task (Mikolov et al. 2013b). Comparing with human supervision, they encode richer semantic relationships between labels and even achieve compelling performance. In this paper, we use both two different semantic spaces for the experiments.

Semantic Mappings $g_s(\mathbf{x})$. Most existing ZSL methods focus on improving the semantic mappings $g_s(\mathbf{x})$ mainly using multi-label classification methods (Hariharan, Vishwanathan, and Varma 2012; Mensink, Gavves, and Snoek 2014) or regression models (Norouzi et al. 2013). For example, DAP (Lampert, Nickisch, and Harmeling 2009), learns $g_s(\mathbf{x})$ independently for each attribute classifier by a Binary Relevance method in ML (Zhang and Zhou 2014). Recently, several papers suggest approaches for joint learning of $g_s(\mathbf{x})$ with relationships between features, attributes and classes. (Akata et al. 2015) propose a label embedding approach that implicitly learns the instances and semantic embeddings onto a common space. For the first time, they show purely unsupervised semantic embeddings achieve compelling results. Following the same principle, (Romera-Paredes, OX, and Torr 2015) further propose a simplified model called ESZSL that is extremely easy and efficient. It is able to outperform state of the art approaches on standard datasets.

Class Predictors $l(\phi)$. Conventional choice for class predictor $l(\phi)$ is neareast neighbor with different distance metrics, such as Euclidean, cosine or hamming distances. Additionally, some researchers attempt to adopt novel methods to make up for the deficiency of $g_s(\mathbf{x})$. For example, (Fu et al. 2015b) adopt an absorbing Markcov process on a semantic graph over all class labels after redefining the distance metric. While, (Rohrbach, Ebert, and Schiele 2013) uses label propagating (Zhou et al. 2004) on a graph structure over the whole testing instances. To further improve the ZSL performance, (Fu et al. 2015a) combines multiple semantic spaces and propagated label predictions on multiple graphs. However, these above methods all try to solve the shift problem after knowledge transferring, not the knowledge transferred itself.

Existing ZSL methods mainly focus on the above four aspects to improve performance. Specially, many efforts have been made to optimize $g_s(\mathbf{x})$ for training dataset. Whereas how to reach the real function $g_u(\mathbf{x})$ with $g_s(\mathbf{x})$ has received little attention, which is still a bottleneck problem in ZSL. Contrarily, we are primarily concerned with this problem in this paper.

Proposed Method

For unseen class prediction, using $g_s(\mathbf{x})$ to replace $g_u(\mathbf{x})$ will lead to significant performance degradation. To solve this problem, there are two possible choices: 1) enhance the generalization capability of $g_s(\mathbf{x})$ for unseen classes prediction, or 2) learn $g_u(\mathbf{x})$ directly. Two strategies are very useful for enhancing the performance of the current zero-shot learning algorithms.

We propose a relational knowledge transfer (RKT) method to take into account two cases. RKT can be used as a common framework for any need to use $g_s(\mathbf{x})$ in replace of $g_u(\mathbf{x})$. In the following, we start by describing basic setup for ZSL. Then we explain how the problem can be tackled by our proposed method in two steps.

Basic Setup and Notation

Let $\mathcal{Y} = \{y_1, ..., y_p\}$ denotes a set of p seen class labels and $\mathcal{Z} = \{z_1, ..., z_q\}$ a set of q unseen categories with $\mathcal{Y} \cap \mathcal{Z} = \emptyset$. Zero shot learning aims to learn a classifier $f_u : \mathcal{X} \rightarrow \mathcal{Z}$ from a training dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, where $\mathbf{x}_i \in \mathbb{R}^d$ represents the i-th instance. Each class label y (or z) corresponds to an m-dimensional vector $\phi \in \mathcal{S}$. The vector ϕ can be binary semantic knowledge $A(0/1)$ describing absence/presence of attributes, or continuous semantic knowledge W encoding geometric manifold structures. For simplicity, we denote the whole training instances as $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_p] \in \mathbb{R}^{d \times N}$, where $\mathbf{X}_i = [\mathbf{x}_1, ..., \mathbf{x}_{N_i}] \in \mathbb{R}^{d \times N_i}$ are all N_i instances in class i. The corresponding semantic knowledge representations are $\mathbf{\Phi} = [\mathbf{\Phi}_1, ..., \mathbf{\Phi}_p], \mathbf{\Phi}_i = [\phi_i, ..., \phi_i] \in \mathbb{R}^{m \times N_i}$. In addition, semantic knowledge representations of all classes are denoted as $\mathbf{\Phi} = [\phi_1, ..., \phi_p, \phi_{p+1}, ..., \phi_{p+q}] \in \mathbb{R}^{m \times (p+q)}$.

RKT: Relational Knowledge Transfer for ZSL

The proposed RKT method includes two steps: 1) extract relational knowledge by sparse coding, and 2) generate labelled virtual instances for unseen classes. The key idea of RKT framework is to obtain the manifold dependence between the seen and unseen classes in the semantic knowledge space by using the sparse coding method. Then the extracted relational knowledge is transferred back to the semantic feature space for the generation of labelled virtual instances.

Without loss of generality, we assume that the function $g(\mathbf{x}) : \mathbf{x} \rightarrow \phi$ is a linear map from the linear feature space $\mathcal{X}$ into the semantic knowledge space $\mathcal{S}$. Then the following basic properties are satisfied without any topological restrictions imposed on the spaces $\mathcal{X}$ and $\mathcal{S}$.

Algebraic homomorphism: *A linear map $g(\mathbf{x})$ is a mapping of $\mathcal{X}$ into $\mathcal{S}$. For $\forall \mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}$, it has the following properties:*

- $g(\mathbf{x}_i + \mathbf{x}_j) = g(\mathbf{x}_i) + g(\mathbf{x}_j)$.
- $g(\lambda \mathbf{x}) = \lambda g(\mathbf{x})$.

Definition. Let $\mathbf{X}_s = [\mathbf{x}_s^1 ..., \mathbf{x}_s^m]$ and $\mathbf{X}_u = [\mathbf{x}_u^1, ..., \mathbf{x}_u^n]$ denote two subsets of data in $\mathcal{X}$. $\mathbf{Y}_s = [\mathbf{y}_s^1, ..., \mathbf{y}_s^m]$ and $\mathbf{Y}_u = [\mathbf{y}_u^1, ..., \mathbf{y}_u^n]$ are their corresponding sets of vectors in $\mathcal{Y}$. For $\forall \mathbf{x}_u^i$, assume $\exists \alpha_i \in \mathbb{R}^m$, $\mathbf{x}_u^i = \mathbf{X}_s \alpha_i$. Similarly, for $\forall \mathbf{y}_u^i, \exists \beta_i \in \mathbb{R}^m$, $\mathbf{y}_u^i = \mathbf{Y}_s \beta_i$. The set $\mathcal{K}_x = \{\alpha_i\}$ and $\mathcal{K}_y = \{\beta_i\}$ are the *relational knowledge* of $\mathbf{X}_u$ and $\mathbf{Y}_u$ on $\mathbf{X}_s$ and $\mathbf{Y}_s$ respectively.

From the viewpoint of semantic representation, the relational knowledge set $\mathcal{K}_x$ encodes a kind of dependence of manifold structure X_u on X_s in space $\mathcal{X}$, as well as $\mathcal{K}_y$ encodes that in space $\mathcal{Y}$.

Lemma. *If $\forall j, \alpha_j = \beta_j$, the linear map $g_s(\mathbf{x}) : \mathbf{x}_s \rightarrow \mathbf{y}_s$ learned from training data set $\{(\mathbf{x}_s^i, \mathbf{y}_s^i)\}$ can be used as $g_u(\mathbf{x}) : \mathbf{x}_u \rightarrow \mathbf{y}_u$ to make predictions for testing data set $\{(\mathbf{x}_u^i, \mathbf{y}_u^i)\}$.*

Proof. Let $\mathbf{x}_u^i = \sum_{j=1}^m \alpha_i^{(j)} \mathbf{x}_s^j$, then $g_s(\mathbf{x}_u^i) = g_s(\sum_{j=1}^m \alpha_i^{(j)} \mathbf{x}_s^j) = \sum_{j=1}^m \alpha_i^{(j)} g_s(\mathbf{x}_s^j) = \sum_{j=1}^m \beta_i^{(j)} \mathbf{y}_s^j = g_u(\mathbf{x}_u^i)$.

Most existing ZSL approaches implicitly assume that the relational knowledge $\mathcal{K}_x$ of the feature space $\mathcal{X}$ is consistent with that of the attribute space (or word2vec knowledge space) $\mathcal{S}$, and directly use $g_s(\mathbf{x})$ instead of $g_u(\mathbf{x})$ for unseen class prediction. In practice, the consistency assumption is often violated due to the following possible factors: 1) feature representations $\mathbf{X}_s, \mathbf{X}_u$ in $\mathcal{X}$ are lack of semantic structure, 2) $g(\mathbf{x})$ is not a linear mapping, and 3) the instances $\mathbf{x}_s^i, i \in c$ of one class c in $\mathcal{X}$ are variant.

For the first case, the more semantic feature representations, such as deep features, are required. For the second case, we can decompose $g(\mathbf{x})$ into local linear functions by sparse coding. Then the sparse coding coefficients can be used as the relational knowledge $\mathcal{K}$ instead of α and β. For the last case, we consider putting a probability noise model on the instances of each class in $\mathcal{X}$.

Corollary. *Suppose $\mathbf{x}_s^{i \in c} \sim \mathcal{N}(\mu_c, \sigma_c \mathbf{I})$, and a linear mapping $g_s(\mathbf{x}) : \mathbf{x}_s^{i \in c} \rightarrow \mathbf{y}_s^c$ is learned from the training dataset $\{(\mathbf{x}_s^{i \in c}, \mathbf{y}_s^c)\}$. Then for $\mathbf{x}_u^{j \in c_u}$, its mapping vector $\mathbf{y}_u^{c_u}$ in $\mathcal{Y}$ is a probability distribution instead of one point.*

Proof. Let $\mathbf{x}_u^{j \in c_u}$, $\mathbf{x}_u^{k \in c_u}$ be two instances of one class c_u, their relational knowledge representations in $\mathcal{X}$ are α_j and α_k, $\alpha_j \neq \alpha_k$. By linear assumption of $g_s(\mathbf{x})$, their corresponding projected points in $\mathcal{S}$ are $\mathbf{y}_u^j = g_s(\sum_{c=1}^m \alpha_j^{(c)} \mathbf{y}_s^c) \neq g_s(\sum_{c=1}^m \alpha_k^{(c)} \mathbf{y}_s^c) = \mathbf{y}_u^k$. So different instances $\mathbf{x}_u^{j \in c_u}$, $\mathbf{x}_u^{k \in c_u}$ that come from same unseen class don't project into the same point $\mathbf{y}_u$ in $\mathcal{Y}$, instead a probability distribution around it.

Obviously, simply using $g_s(\mathbf{x})$ as $g_u(\mathbf{x})$ will lead to the serious manifold divergence for unseen classes. But if we inject the $\mathbf{x}_u^{j \in c_u}$ with its mapping vector $\mathbf{y}_u$ into original training data $\{(\mathbf{x}_s^{i \in c}, \mathbf{y}_s^c)\}$ for the optimization of $g_s(\mathbf{x})$, we can avoid the manifold divergence in the mapping process. This inspires us to consider using the relational knowledge $\mathcal{K}_s$ in $\mathcal{S}$ to generate the labelled virtual data $\{(\hat{\mathbf{x}}_u^{j \in c_u}, \mathbf{y}_u^{c_u})\}$ for unseen classes. Then, we can use the generated virtual data to optimize $g_s(\mathbf{x})$ for unseen data or directly learn $g_u(\mathbf{x})$. In this process, a reverse knowledge transfer mechanism is adopted. The detailed descriptions of our RKT framework are given in the following contents.

Step 1: Relational Knowledge Extraction by Sparse Coding The set $\{\phi_1, ..., \phi_{p+q}\}$ is bound by the geometric distribution of all classes in the semantic knowledge space. The specific geometric distribution not only express the manifold structures of the seen and the unseen classes separately, but also encode the geometric dependence between them. We propose extracting this linear geometric dependence as relational knowledge for RKT framework.

For a semantic vector ϕ_i of i-th class in $\mathcal{S}$, its relational knowledge is represented as a coefficient vector $\mathbf{w}_i$ in sparse

coding. Following this idea, the relational knowledge of each unseen class can be extracted by linearly relating to the seen classes. In the standard learning paradigm, it is formulated as follows,

$$\min_{\mathbf{W}} ||\mathbf{\Phi}_{ts} - \mathbf{\Phi}_{tr}\mathbf{W}||_F^2 + \lambda\Omega(\mathbf{W}), \quad (1)$$

where the parameter matrix $\mathbf{W}$ is composed of $\{\mathbf{w}_i\}_{i=1}^q$ columns, describing the semantic correlation between i-th unseen class and training classes. and Ω is a regularizer. $\mathbf{\Phi}_{tr} = [\phi_1, ...\phi_p]$ and $\mathbf{\Phi}_{ts} = [\phi_{p+1}, ..., \phi_{p+q}]$. Problem (1) encompasses several approaches, depending on the choice of Ω. In this paper, we consider Ω as ℓ_1 norm on each $\mathbf{w}_i$, i.e. $\Omega(\mathbf{W}) = \sum_{i=1}^q ||\mathbf{w}_i||_1$.

Step 2: Generate Labelled Virtual Instances for Unseen Classes In this step, we use $\mathbf{W}$ to generate labelled virtual instances for unseen classes. These generated data inherit the geometric dependence in the semantic knowledge space and enrich the manifold structures in the feature space.

We assume that all instances of j-th class constitute a Gaussian distribution $\mathcal{N}(\mu_j, \mathbf{\Sigma}_j)$ within the feature space $\mathcal{X}$. Then $\mu_j \in \mathbb{R}^d$ and $\mathbf{\Sigma}_j \in \mathbb{R}^{d\times d}$ can be computed as the empirical mean vector and covariance matrix of data in j-th class.

Using the relational knowledge $\mathbf{W}$, the mean vector of the Gaussian distribution for unseen class i is generated as $\mu_i = \sum_{j=1}^C \mathbf{W}_{j,i}\mu_j$. While its covariance matrix is $\mathbf{\Sigma}_i = \sigma\mathbf{I}$, where σ is a predefined prior knowledge. For each unseen class, we randomly generate M points from its Gaussian distribution, denoted as $\mathbf{X}_{aug}$, with semantic knowledge representations denoted as $\mathbf{\Phi}_{aug}$.

Application of RKT in ZSL Provided with the generated data $\mathbf{X}_{aug}$, we consider two strategies to improve recognition accuracy in ZSL. One is to enhance the generalization of $g_s(\mathbf{x})$ for unseen categories by learning from the augmented datasets $\tilde{\mathbf{X}} = [\mathbf{X}, \mathbf{X}_{aug}]$ and $\tilde{\mathbf{\Phi}} = [\mathbf{\Phi}, \mathbf{\Phi}_{aug}]$. The other is to directly learn the semantic mapping $g_u(\mathbf{x})$ just using the generated data $\mathbf{X}_{aug}$ and $\mathbf{\Phi}_{aug}$. These two strategies can be incorporated in any ZSL approaches using mid-level semantic space.

Experiments

In order to assess our approach and the validity of the statements we make, we conduct a set of experiments.

Experimental Setup

Datasets We evaluate our work on three real-world ZSL datasets: Animals with Attributes (AwA) (Lampert, Nickisch, and Harmeling 2009), Caltech UCSD Birds (CUB) (Wah et al. 2011) and Stanford Dogs (Dogs) (Khosla et al. 2011). They consist of images belonging to categories in different scopes: coarse-grained animals, fine-grained birds and dogs respectively. AwA and CUB contain attribute-labelled categories, containing 85 and 312 attributes respectively. Dogs dataset has no attributes available. For each dataset, we learn Word2Vec (Mikolov et al. 2013a) category representations from Wikipedia.

Features For comparative purposes, we use 2 types of deep features, extracted from 2 popular CNN architectures, i.e. VGG (Simonyan and Zisserman 2015) and GoogLeNet (Szegedy et al. 2014). For GoogLeNet, we use the 1024-dim activations of last layer but one as features, denoted as *Goog*. While for VGG, we choose the 1000-dim last fully connected layer activations as features, denoted as *fc8*. They are both low-dimension (relative to the website offered features) and high-semantic features.

Baselines We compare with two baseline methods, i.e. DAP (Lampert, Nickisch, and Harmeling 2009) and ESZSL (Romera-Paredes, OX, and Torr 2015). DAP is the earlier-proposed and widely-used standard single-task learning baseline, in which each attribute is learned separately. Proposed in 2015, ESZSL is a latest multi-label learning algorithm for zero-shot learning. It has three nice properties: effectiveness, efficiency and easy to implement. In contrast to DAP, it plays a more important role in ZSL when ϕ is continuous.

Evaluation on the Ability of RKT to Boost $g_s(\mathbf{x})$

To evaluate the ability of RKT, we consider 4 different semantic knowledge: manual binary attributes (A(0/1)), continuous attributes (A[0,1]), word vectors (W) and a combination of continuous attributes and word vectors (A[0,1]+W). We follow the general protocol for learning Word2Vec using Wikipedia. On AwA, we testify the efficacy of 2 deep features. For Dogs and CUB, we use only the *Goog* feature, for fair comparison. Experimental results are shown in Table 1. Excitingly, we find our proposed method boosts *ALL* the results of the baselines in all different settings.

On AwA dataset, we obtain the inspiring accuracy **82.43%**, even higher than the state-of-the-art result 80.5% by (Fu et al. 2015a) to the best of our knowledge. Notably, in the binary attribute space, ZSL recognition accuracy improves by 2.5% while attribute prediction accuracy is risen by only 1.1%.

On CUB and Dogs, the performance is not as good as in AwA. The best result is 43.12% on CUB. However, the improvements are still promising. We ascribe this performance reduction to two possible reasons. One is that there are much more attributes in CUB, leading to a more complex semantic manifold. Another reason is the much finer granularity in their classes, which can hardly be reached by these general deep features. Accordingly, the more consistent semantics are in $\mathcal{X}$ and $\mathcal{S}$, the more stronger it can boost the ZSL performance, as we have discussed.

Evaluation on Directly-Learned $g_u(\mathbf{x})$ with Only Augment Data

As we state that the shift between the semantic mapping $g_s(\mathbf{x})$ from the training dataset and the real mapping $g_u(\mathbf{x})$ for unseen categories leads to performance degradation. One solution we propose is to learn directly $g_u(\mathbf{x})$ for unseen categories. To validate this solution, we further conduct a set of experiments.

In these experiments, the only data we get for training classes are their average vectors. We perform ZSL only on

Table 1: Average zero-shot recognition accuracy (%) obtained by DAP, ESZSL and our proposed method RKT on AwA, CUB and Dogs. RKT denotes incorporating RKT into DAP or ESZSL. Values on the left side of the DAP column denote the attribute recognition accuracy. Notations A(0/1), A[0,1] and W represent binary attribute knowledge, continuous attribute knowledge and Word2Vec knowledge, respectively. '–' means no result reported.

			A(0/1)		A[0,1]		W		A[0,1]+W	
Datasets	Feature	Dim	DAP	RKT	ESZSL	RKT	ESZSL	RKT	ESZSL	RKT
AwA	*Goog*	1024	77.57/53.96	79.77/58.69	67.07	71.59	54.43	59.05	74.84	79.40
	fc8+Goog	2024	77.78/55.23	80.30/62.91	72.44	75.99	53.72	58.12	79.03	***82.43***
CUB	*Goog*	1024	87.19/27.03	88.98/33.83	32.24	33.48	22.24	23.21	42.14	**43.12**
Dogs	*Goog*	1024	–	–	–	–	20.49	**22.51**	–	–

the generated data of unseen classes. We denote this strategy as RKT_AO. In all cases, we use the same model parameters and *Goog* features as the ones adopt in baselines, but a smaller size of virtual data than RKT. We use continuous attribute knowledge for clarity. Two main formulations of Word2Vec for object classes are learned, i.e. skip-gram and continuous bag-of-words (cbow) (Mikolov et al. 2013b). For AwA and Dogs, we use 200-dim and 500-dim skip-gram vectors respectively. While for CUB, we adopt 500-dim cbow representation. According to the results shown in Table 2, we see RKT_AO significantly outperforms RKT in all cases, which shows that the directly learned $g_u(\mathbf{x})$ with augment data only is more fit for the unseen classes.

Table 2: Average classification accuracy (%) using RKT and RKT_AO methods, where RKT_AO means using only augmented data for ZSL.

Methods		RKT	RKT_AO	Increase
AwA	A	71.59	72.49	+0.9
	W	59.05	76.36	**+17.31**
	A+W	79.40	**81.31**	+1.91
CUB	A	33.48	39.62	**+6.14**
	W	23.21	25.62	+2.41
	A+W	43.12	**46.24**	+3.12
Dogs	W	22.51	**28.29**	**+5.78**

We further display directly the performance improvement over baselines in Figure 2. Values in this figure denote increase of the absolute accuracy. We can see that RKT performs better no matter what the semantic knowledge is used for these datasets. Spectacularly, the accuracy improvement over baselines reaches the highest 21.9%. This result also shows a meaningful point that the improvement using W is very high, which proves unsupervised-learned word vectors have the potential of boosting performance, while at the same time avoiding the cost of manual attribute knowledge.

In another experiment, we test how the number of synthesised instances affects the performance. Figure 3 illustrates the performance of RKT on a varying number of generated data, where the divisor in the x-axis denotes average number of instances per class. We observe that RKT outperforms the baselines with just a small number of data. The performance drops down as a result of over-fitting. For reason of space-saving, here we only show 4 configurations (different word

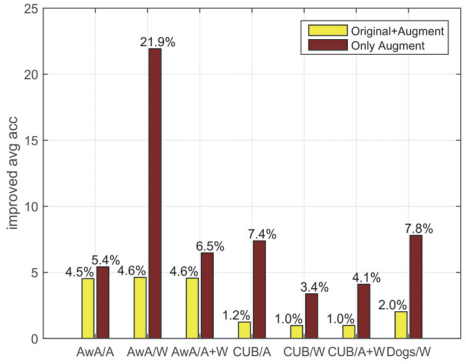

Figure 2: Accuracy improvement over baselines. Results are obtained through using original and augmented data (shown in yellow bars) and only augmented data (in red).

vectors and dimensions), and they exhibit the same trend.

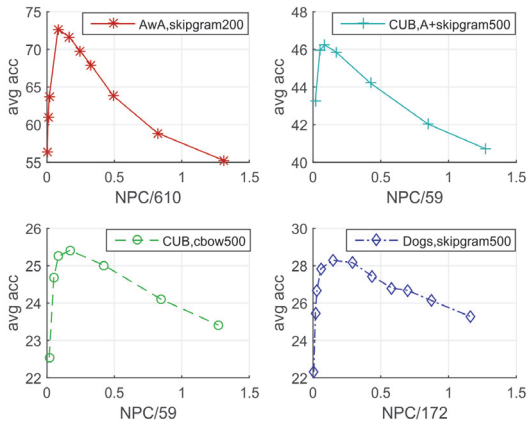

Figure 3: ZSL average accuracy obtained with only the augmented data for unseen categories. The x-axis denotes different number of instances per class (NPC), where the divisor is the average number per class.

These results provide strong evidence for our statement that relational knowledge in the semantic space benefits ZSL

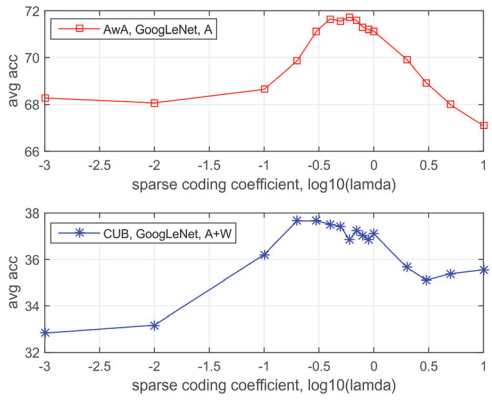

Figure 4: Performance (%) w.r.t. different λ in sparse coding.

a lot. Testing instances synthesised using this knowledge are able to cover the real manifold structure for unseen categories. It is worth mentioning that, we only need the mean vector of training class in $\mathcal{X}$ and its semantic descriptions in $\mathcal{S}$. With just a little semantic information of a new instance, it can perform well. Therefore, the proposed strategy would benefit *large scale* or *online* zero-shot recognition, which is our future work.

Table 3: Comparisons with the state-of-the-art ZSL methods. '–' means no result reported. (*: Method uses word vectors learned from a specific corpus.)

Methods	ϕ	AwA	CUB	Dogs
SJE	A/W	66.7	**50.1***	**33.0***
HAP	A	45.6	17.5	–
ZSLwUA	A	43.01	–	–
PST	A	42.7	–	–
TMV-HLP	A+W	80.5	47.9	–
AMP	A+W	66	–	–
RKT	A	75.99	39.62	–
	W	76.36	25.62	28.29
	A+W	**82.43**	**46.24**	–

Comparison with state of the art

To better show the ability of our proposed method, we compare RKT with 6 state-of-the-art ZSL methods, i.e. SJE (Akata et al. 2015), HAP (Huang et al. 2015), PST (Rohrbach, Ebert, and Schiele 2013), ZSLwUA (Jayaraman and Grauman 2014), AMP (Fu et al. 2015b) and TMV-HLP (Fu et al. 2015a). All methods use CNN features except PST. For simple comparison, we use same settings and author-provided results. From the results in Table 3 we can see RKT significantly outperforms other methods on AwA dataset. While on CUB and Dogs, the best performances are given by SJE, which uses specific word vectors, making $\mathcal{S}$ more semantic than ours. However, when adopting the same Word2Vec as ours, their recognition accuracy falls off. TMV-HLP's work on CUB is higher than ours, because they improve the classifier $l(\phi)$, while we focus on $g_u(\mathbf{x})$ and use the simple cosine distance for $l(\phi)$. Since our results are obtained based on the baselines, we expect the result to be further improved when incorporated with other ZSL methods in this table, which is also a work in the future.

Further evaluations

In the first RKT experiment, we evaluate how the sparse coding coefficient affects the performance. Larger λ makes the semantic relation $\mathbf{w}$ more sparse. The NPC is set to the average size of training classes. The result in Figure 4 shows that, when $\log(\lambda)$ is about -0.4, the performance is the highest. This result makes sense for the use of sparse coding.

In the second experiment, we assess how the approach performs on varying number of augmented data in training dataset. From the results on AwA displayed in Figure 5, we can see our method outperforms the baselines and improves with more instances. This is because the *Goog* features are more semantic. This result further validates our statement about the semantic geometric structure in $\mathcal{S}$.

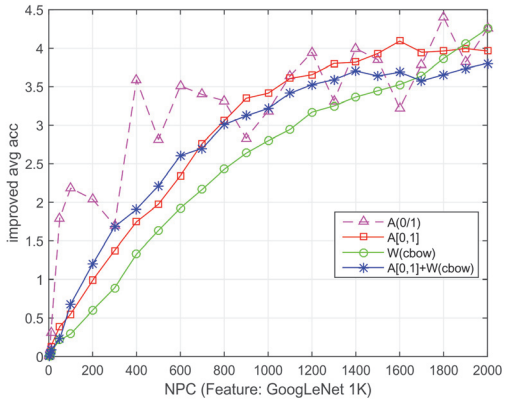

Figure 5: Improved ZSL performance on AwA when varying NPC are added. A denotes attribute knowledge, while W represents word vector knowledge.

Discussion and Conclusion

In this paper, we propose a relational knowledge transfer (RKT) mechanism for zero-shot learning (ZSL) that is general to boost most existing ZSL approaches and show good performance on three real world datasets. It resolves the inherent semantic shift problem and achieves ZSL through restoring the missing manifold structure of unseen categories in feature space by synthesising instances via relational knowledge from the semantic space.

Since we focus on optimizing semantic mapping $g(\mathbf{x})$, the joint optimization of $g(\mathbf{x})$ and $l(\phi)$ is still encouraged in the future work.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (No. 61473256) and China Knowledge Center for Engineering Sciences and Technology.

References

Akata, Z.; Perronnin, F.; Harchaoui, Z.; and Schmid, C. 2013. Label-embedding for attribute-based classification. In *CVPR*, 819–826. IEEE.

Akata, Z.; Reed, S.; Walter, D.; Lee, H.; and Schiele, B. 2015. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, 2927–2936.

Fu, Y.; Hospedales, T.; Xiang, T.; and Gong, S. 2015a. Transductive multi-view zero-shot learning. *PAMI* 1–1.

Fu, Z.; Xiang, T.; Kodirov, E.; and Gong, S. 2015b. Zero-shot object recognition by semantic manifold distance. In *CVPR*, 2635–2644.

Girshick, R.; Donahue, J.; Darrell, T.; and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 580–587. IEEE.

Hariharan, B.; Vishwanathan, S.; and Varma, M. 2012. Efficient max-margin multi-label classification with applications to zero-shot learning. *Machine learning* 88(1-2):127–155.

Huang, S.; Elhoseiny, M.; Elgammal, A.; and Yang, D. 2015. Learning hypergraph-regularized attribute predictors. *arXiv preprint arXiv:1503.05782*.

Huang, S.-J.; Gao, W.; and Zhou, Z.-H. 2014. Fast multi-instance multi-label learning. In *AAAI*.

Jayaraman, D., and Grauman, K. 2014. Zero-shot recognition with unreliable attributes. In *NIPS*, 3464–3472.

Khosla, A.; Jayadevaprakash, N.; Yao, B.; and Fei-Fei, L. 2011. Novel dataset for fine-grained image categorization. In *First Workshop on Fine Grained Visual Categorization, CVPR*. Citeseer.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*, 1097–1105.

Lampert, C. H.; Nickisch, H.; and Harmeling, S. 2009. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 951–958. IEEE.

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

Luo, P.; Wang, X.; and Tang, X. 2013. A deep sum-product architecture for robust facial attributes analysis. In *ICCV*, 2864–2871. IEEE.

Mensink, T.; Gavves, E.; and Snoek, C. G. 2014. Costa: Co-occurrence statistics for zero-shot classification. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, 2441–2448. IEEE.

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

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013b. Distributed representations of words and phrases and their compositionality. In *NIPS*, 3111–3119.

Norouzi, M.; Mikolov, T.; Bengio, S.; Singer, Y.; Shlens, J.; Frome, A.; Corrado, G. S.; and Dean, J. 2013. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*.

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

Rohrbach, M.; Ebert, S.; and Schiele, B. 2013. Transfer learning in a transductive setting. In *NIPS*, 46–54.

Romera-Paredes, B.; OX, E.; and Torr, P. H. 2015. An embarrassingly simple approach to zero-shot learning. In *ICML*, 2152–2161.

Simonyan, K., and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. *ICLR*.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2014. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology.

Xiao, J.; Hays, J.; Ehinger, K.; Oliva, A.; Torralba, A.; et al. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 3485–3492. IEEE.

Zhang, M.-L., and Zhou, Z.-H. 2014. A review on multi-label learning algorithms. *Knowledge and Data Engineering, IEEE Transactions on* 26(8):1819–1837.

Zhang, N.; Paluri, M.; Ranzato, M.; Darrell, T.; and Bourdev, L. 2014. Panda: Pose aligned networks for deep attribute modeling. In *CVPR*, 1637–1644. IEEE.

Zhou, D.; Bousquet, O.; Lal, T. N.; Weston, J.; and Schölkopf, B. 2004. Learning with local and global consistency. *NIPS* 16(16):321–328.