Embarrassingly Easy Zero-Shot Image Recognition

Wenli Song[1], Lei Zhang[1*], Jingru Fu[1]

[1]School of Microelectronics and Communication Engineering, Chongqing University
{swl,leizhang,jrfu}@cqu.edu.cn

Abstract. Zero-shot Learning (ZSL) aims to transfer knowledge from seen image categories to unseen ones by leveraging semantic information. It is generally assumed that the seen and unseen classes share a common semantic space. A number of methods propose to design a common space to accomplish the projection between image and class embeddings by learning a compatibility function, which make up sample pairs to train the object function. However, considering the drawbacks of previous compatibility function, we design a new compatibility function in this paper. Different from previous compatibility pattern, our proposed compatibility function is more discriminative by employing label vectors, which can measure the similarity between the projected image features and all seen class prototypes. Extensive experiments on four benchmark datasets show the effectiveness of our proposed approach.

Keywords: Zero-Shot Learning · Knowledge Transfer · Semantic Embedding

1 Introduction

At present a general strategy for ZSL is that both seen and unseen classes share a common semantic space. In this way, some knowledge learned during the training stage is able to transfer to the testing stage. Semantic space can be semantic attribute space [8], [7] or semantic word vector space [4]. In semantic space, the labels of seen and unseen classes can be represented as vectors called class prototypes [5].

Considering the projection from visual to semantic space may cause loss of available features, a large number of previous methods propose to employ a parameter to connect the image and semantic embedding [8], [1], [17]. The parameter is the visual-semantic mapping matrix to be learned which most existing approaches of ZSL construct the common space in this way. The differences of them mainly focus on how to represent the image and semantic vectors and how to design different regularization terms. However, the formulation of the bilinear compatibility is always fixed. Though it is a general framework that can be applied to any learning problem with more than one modality, ZSL solves the problem which the visual and semantic space are completely independent to each other. Therefore, it is doubtful whether the common space with this

compatibility function of applying a parameter is discriminative enough to complete classification. In addition, almost all of the object functions are designed by employing a ranking formulation. Due to arranging a corresponding and a non-corresponding semantic vector for each visual vector, the choose of sample pairs may cause discriminative information loss on the other different classes. Besides, how to choose and how much to choose positive and negative pairs are key to the bilinear compatibility function. When the number of sample pairs is not appropriate, it is easy to cause overfitting to seen classes while invalid to unseen classes.

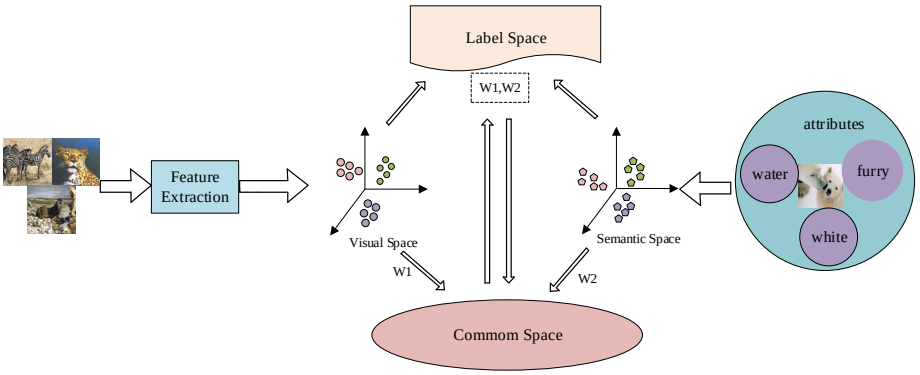

Fig. 1. The framework of our proposed method. In the flowchart, we can see that we first construct the visual and semantic space by feature extraction and attribute annotations. Then we build up a common space to align visual and semantic space by applying two parameters W_1 and W_2 separately. To make the common space discriminative, we combine the common space with the label space which design a compatibility function by also applying W_1 and W_2. Different from previous compatibility function, we don't design positive and negative pairs and avoid the overfitting of the training stage. Our method aims to utilize labels of all seen samples to construct robust relationship between visual and corresponding semantic feature.

Under these circumstances, it is natural for us to consider whether only using a parameter to accomplish the procedure of compatibility is reasonable. In addition, designing a objection function which can reduce the discriminative information loss during the process of compatibility and generalize well to test examples is encouraging. To address above mentioned pitfalls, as is illustrated in Fig. 1, we propose to construct a common embedding space and explore structure for both visual and semantic representations simultaneously. Specifically, the proposed method utilizes two parameters to denote the projection for visual feature and semantic feature separately. They align the structure of visual and semantic space in the common space, at the same time a linear transformation is utilized to attributes, which can combine different attributes and make attributes of different object classes more discriminative. To make the common

space discriminative enough to complete classification, we propose a compatibility function by using the same two parameters in the common space. Different from the previous compatibility methods, we expect our compatibility function can represent each sample's true label which we add label space to our model as seen class classifiers. Our compatibility function no longer takes a sample of positive and negative pairs, instead we make a similarity contrast between each seen sample to all seen class prototypes which ensures that the compatibility between projected visual feature and corresponding class prototype is higher than that of all other class prototypes and thus can preserve the discriminative information for different classes. In this way, each seen sample can get close to corresponding class prototype and get far away any other class prototypes. We confirm that the common space combines with the proposed compatibility function can learn more robust relationships between visual and semantic features.

2 Related Work

In order to reduce the dependency on the lots of labeled datas, ZSL is proposed by [10]. It aims to tackle the problem of recognizing the classes that have never been trained before. Training attribute classifiers is an intuitive way to solve ZSL. For example, [9] proposes the DAP(Direct Attribute Predict) model and IAP(Indirect Attribute Predict) model. Considering the unreliability of the attribute classifiers, i.e. they can accurately predict attributes but they maybe poorly classify, lots of methods then solve ZSL based on label embedding. [4] and [8] both employ a ranking formulation for zero-shot learning using visual and semantic representations and recognize an image by the score of ranking formulation. Then [1] relates the image and semantic features linearly in a joint embedding space with several compatibility functions.

3 Proposed Method

3.1 Mathematical Notations

Suppose there are c_s seen classes with n_s labeled samples $\Phi_s = \{X_s, A_s, Z_s\}$ and c_u unseen classes with n_u unlabeled samples $\Phi_u = \{X_u, A_u, Z_u\}$. $X_s \in \mathbb{R}^{N_s \times d}$ and $X_u \in \mathbb{R}^{N_u \times d}$ are seen and unseen images visual feature vectors. N_s is the number of seen samples and N_u is the number of unseen samples, d is the dimension of visual features. $A_s \in \mathbb{R}^{N_s \times m}$ and $A_u \in \mathbb{R}^{N_u \times m}$ are seen and unseen samples corresponding semantic features, m is the dimension of semantic features. Z_s and Z_u are the prototype semantic representations of the seen and unseen classes. In zero-shot recognition settings, the seen and unseen classes are disjoint: $Z_s \cap Z_u = \oslash$. A_s and A_u are composed of prototype semantic representations of the seen and unseen classes respectively. The task of ZSL is to estimate A_u and then get labels of unseen samples.

3.2 Model Formulation

The key to solve ZSL task is to build up relationship between visual space and semantic space. Considering there exists lots of typical problems when using ridge regression to accomplish the projection from the visual space to semantic space, we explore a common space using two parameters to contact the visual space with semantic space. One matrix completes the dimension transformation from the visual space to common space. [6] points out that due to there are correlations among the attributes, it is necessary to build up relationship between attributes and attributes. Similarly, we use a linear transformation matrix to deal with the semantic vectors. In this way, the other matrix accomplishes the projection from the semantic space to common space and gets more discriminative semantic features. Then the objective function can be

$$\min_{W_1,W_2} \left\|W_1 X_s^T - W_2 A_s^T\right\|_F^2 \tag{1}$$

where $\|.\|_F$ denotes the Frobenius norm, W_1 and W_2 denote the learned projection matrix for visual feature X_s and semantic feature A_s respectively. Thus the visual space and semantic space can be projected to common space.

Considering the discriminative information loss on the other different classes by using previous compatibility function of applying a parameter, we want to design a compatibility function to preserve the discriminative information for different classes and make the common space more discriminative. Thus we design our compatibility function by utilizing the seen samples' labels. Specifically, the relationship between projected visual and semantic features is learned with labels by applying the same two parameters of the projection process, as is shown in Fig. 1:

$$\min_{W_1,W_2} \left\|X_s W_1^T W_2 Z_s^T - Y\right\|_F^2 \tag{2}$$

where $Y = [y_1, y_2, \cdots, y_{N_s}] \in \mathbb{R}^{N_s \times c_s}$ and y_i is a one-hot vector which represents the true label of x_i. Z_s is the prototype semantic representations of the seen classes. Thus, the common space can connect with the label space which makes the common space more discriminative. In addition, the proposed compatibility function preserve the discriminative information for different classes.

Then we consider combining the two proposed terms to accomplish more effective classification by using two parameters. The final object function can be

$$\begin{aligned}\min_{W_1,W_2} &\left\|W_1 X_s^T - W_2 A_s^T\right\|_F^2 + \beta \left\|X_s W_1^T W_2 Z_s^T - Y\right\|_F^2 \\ &+ \lambda_1 \left\|W_1\right\|_F^2 + \lambda_2 \left\|W_2\right\|_F^2\end{aligned} \tag{3}$$

λ_1 and λ_2 are the coefficient of the regularizers, β is a weighting coefficient to control the importance of the first and second terms.

In summary, we learn a discriminative common space which can accomplish the projection of visual and semantic features respectively and force $W_1 x_i^T$ to

be as close as $W_2 a_j^T$ in the common space if a_j is x_i corresponding semantic indication, i.e. $i{=}j$. Besides, considering the case that a_j is not x_i corresponding semantic indication, i.e. $i{\neq}j$, we make a similarity contrast between each seen sample to all seen class prototypes which ensures that the compatibility between projected visual feature and corresponding class prototype is higher than that of all other class prototypes. Thus we can preserve the discriminative information for different classes and get a good recognition effect.

3.3 Optimization

It is obvious that Eq.(3) is not convex for W_1 and W_2 simultaneously, but it is convex for each of them separately. To optimise the objective in Eq.(3), we use an alternating optimization method. Specifically, we alternate between the following subproblems:

Fix W_2 and update W_1 To optimise Eq.(3), we can calculate derivative of W_1 and set it zero, then can get the Sylvester equation:

$$\begin{aligned}&(\beta W_2 Z_s^T Z_s W_2^T) W_1 + W_1(\lambda_1 (X_s^T X_s)^{-1} + I) \\ &= (W_2 A_s^T X_s + \beta W_2 Z_s^T Y^T X_s)(X_s^T X_s)^{-1}\end{aligned} \tag{4}$$

where I is the identity matrix, $A_1 = \beta W_2 Z_s^T Z_s W_2^T$, $B_1 = \lambda_1 (X_s^T X_s)^{-1} + I$, $C_1 = (W_2 A_s^T X_s + \beta W_2 Z_s^T Y^T X_s)(X_s^T X_s)^{-1}$. The Sylvester equation can be solved easily in MATLAB:

$$W_1 = sylvester(A_1, B_1, C_1) \tag{5}$$

Fix W_1 and update W_2 This problem can be solved in the same way as the solution to W_1, then can get the Sylvester equation:

$$\begin{aligned}&(\beta W_1 X_s^T X_s W_1^T) W_2 + W_2(A_s^T A_s + \lambda_2 I)(Z_s^T Z_s)^{-1} \\ &= (W_1 X_s^T A_s + \beta W_1 X_s^T Y Z_s)(Z_s^T Z_s)^{-1}\end{aligned} \tag{6}$$

where $A_2 = (\beta W_1 X_s^T X_s W_1^T)$, $B_2 = (A_s^T A_s + \lambda_2 I)(Z_s^T Z_s)^{-1}$, I is the identity matrix, $C_2 = (W_1 X_s^T A_s + \beta W_1 X_s^T Y A_s)(Z_s^T Z_s)^{-1}$. The Sylvester equation can be solved easily in MATLAB:

$$W_2 = sylvester(A_2, B_2, C_2) \tag{7}$$

In our experiments, the optimization process always converges after seven iterations, usually less than 25.

3.4 ZSL Classification

Due to we have two fields of restriction on the common space, we can perform ZSL in two methods.

Classification applying the compatibility function We can employ the learned W_1 and W_2 to build up relationships between the test sample x^u and unseen classes A_u. Specifically, considering the dimension of visual space is higher than semantic space, we classify in the visual space:

$$f(x_i^u) = \arg\min_j d(x_i^u, a_j^u W_2^T W_1) \tag{8}$$

Classification applying the projected function We can utilize the learned W_1 to accomplish the projection of original image data. For the unseen class prototypes, we project their attribute representations to the common space by the transformation matrix W_2.

$$f(x_i^u) = \arg\min_j d(x_i^u W_1^T, a_j^u W_2^T) \tag{9}$$

where x_i^u is the visual represent of the i-th unseen sample, W_1 and W_2 are the compatibility parameters, a_j^u is prototype attribute vector of the j-th unseen class, d is a cosine distance function, and $f()$ returns the predicted label of the unseen sample.

4 Experiments

4.1 Datasets and Settings

Datasets We perform experiments on four benchmark ZSL datasets, i.e. Animals with Attributes (AwA) [9], Caltech-UCSD Birds-200-2011 (CUB-200) [16], aPascal & aYahoo (aP&Y) [3], and SUN Attribute (SUN) [12]. The summary of these datasets is given in Table 1.

Table 1. Statistics of different datasets: AWA, CUB, aP&Y, SUN in terms of instance numbers, dimension of semantic vector, seen and unseen classes numbers

Database	Instance	Attributes	Seen / Unseen
AwA	30475	85	40 /10
CUB-200	11788	312	150 /50
aP&Y	15339	64	20 / 12
SUN	14340	102	707 / 10

Parameter settings In our experiments, we use GoogleNet features [15] which is the 1024D activation of the final pooling layer as in [1]. We use attribute annotations as the semantic space for the datasets.

4.2 Evaluations of the Proposed Framework

We compare our method with previous methods in Table 2. Our proposed model improves the state-of-the-art performance on the datasets. For AWA dataset, our model achieves 85.16% and 85.03% separately by using proposed two recognition methods. Both of them obtain comparative results and achieve the best performance. For CUB dataset, our result is lower than SAE [7](61.4%) and SCoRe [11](58.4%). Compared with the previous models that applying traditional compatibility function, such as ALE [8] and SJE [1], our proposed method observes a significant improvement, which demonstrates the effectiveness of the common space. For aP&Y dataset, our result ranks the second. The split of aP&Y dataset is 20/12. The reason may lie in a smaller number of seen classes, which causes less discriminative on label space. For SUN dataset, our model achieves 92.0%, which is higher than almost all previous methods. The encouraging result further confirms that it is effective to employ label vectors in the common space.

Table 2. Zero-shot recognition results on AWA, CUB, aP&Y, SUN(%). CF means that we use compatibility function to classify and PF means that we use projected function to classify. '*' denotes the visual features are extracted by the imagenet-vgg-verydeep-19 [14] pre-trained model.

Method	AWA	CUB	aP&Y	SUN
DAP* [9]	57.2	44.5	38.2	72.0
IAP [9]	57.2	36.7		40.8
SynC [2]	72.9	54.5		62.8
SJE [1]	66.7	50.1		
LatEM [17]	71.9	45.5		
ALE [8]	49.7	35.8	30.9	38.2
ESZSL* [13]	75.3		24.2	82.1
DeViSE [4]	56.7	33.5		
SCoRe [11]	78.3	58.4		
SAE [7]	84.7	**61.4**	**55.4**	91.0
CF(Ours)	**85.16**	54.71	50.80	**92.00**
PF(Ours)	85.03	56.06	47.50	90.00

5 Conclusion

In this paper, we use attributes as semantic vector to construct semantic space and evaluate our method on four datasets. We employ two parameters to separately embed the visual and semantic features into a common embedding space and the common space is combined with the label space. In this way, the learned compatibility parameters will be discriminative with category information. It is reasonable to think that the common space combined with label space is the

key to effective ZSL. The explicit and closed solution makes the method efficient to optimize. Our method obtains competitive results on the four benchmark datasets.

References

1. Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *Computer Vision & Pattern Recognition*, 2015.
2. Soravit Changpinyo, Wei Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *Computer Vision & Pattern Recognition*, 2016.
3. A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *IEEE Conference on Computer Vision & Pattern Recognition*, 2009.
4. A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. In *International Conference on Neural Information Processing Systems*, 2013.
5. Yanwei Fu, Timothy M. Hospedales, Xiang Tao, Zhenyong Fu, and Shaogang Gong. Transductive multi-view embedding for zero-shot recognition and annotation. In *European Conference on Computer Vision*, 2014.
6. Huajie Jiang, Ruiping Wang, Shiguang Shan, Yang Yi, and Xilin Chen. Learning discriminative latent attributes for zero-shot classification. In *IEEE International Conference on Computer Vision*, 2017.
7. Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. 2017.
8. Christoph H Lampert, Nickisch Hannes, and Harmeling Stefan. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 36(3):453–465, 2014.
9. Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. *Learning to detect unseen object classes by between-class attribute transfer.* 2009.
10. Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *Aaai Conference on Artificial Intelligence*, 2014.
11. Pedro Morgado and Nuno Vasconcelos. Semantically consistent regularization for zero-shot recognition. 2017.
12. Genevieve Patterson, Xu Chen, Su Hang, and James Hays. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108(1-2):59–81, 2014.
13. Bernardino Romera-Paredes and Philip H S Torr. An embarrassingly simple approach to zero-shot learning. In *International Conference on International Conference on Machine Learning*, 2015.
14. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *Computer Science*, 2014.
15. Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. 2014.
16. Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds200-2011 dataset. *Advances in Water Resources - ADV WATER RESOUR*, 07 2011.
17. Y. Xian, Z. Akata, G. Sharma, Q. Nguyen, M. Hein, and B. Schiele. Latent embeddings for zero-shot classification. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 69–77, June 2016.