LETTER

A Semantic Similarity Supervised Autoencoder for Zero-Shot Learning

Fengli SHEN[†], ***Nonmember and*** **Zhe-Ming LU**[†a)], ***Member***

SUMMARY This Letter proposes a autoencoder model supervised by semantic similarity for zero-shot learning. With the help of semantic similarity vectors of seen and unseen classes and the classification branch, our experimental results on two datasets are 7.3% and 4% better than the state-of-the-art on conventional zero-shot learning in terms of the averaged top-1 accuracy.

key words: *zero-shot learning, autoencoder, image classification*

1. Introduction

With a large number of labelled training data, supervised classification accuracy has rapidly increased by utilizing deep neural network in recent years. However, the downside of deep learning technology is that it can only classify classes appearing in training data. Therefore, the concept of Zero-Shot Learning (ZSL) was proposed in the scenario where test classes are not provided during the training stage. In order to transfer the knowledge learned from seen classes to unseen classes, we need auxiliary information, such as manually annotated attributes and word2vec learned from text, to link between seen classes and unseen ones in the semantic space.

According to how to establish the mapping function between the visual space and the semantic space, ZSL methods can be divided into four categories. In the first category, models learn a projection function from the visual feature space to the semantic space [1]. In order to alleviate the hubness problem in the semantic space, the second category chooses the reverse mapping direction [2]. Methods in the third category find some intermediate spaces for both visual feature vectors and semantic embeddings to be mapped to [3]. The last category is the combination of the first category and the second category, which learns mapping functions between the visual space and the semantic space simultaneously, represented by generative adversarial networks (GAN) and autoencoder based models [4]–[8]. We use an autoencoder as our main architecture. Thus our model belongs to the last category.

In this work, we present a semantic similarity supervised model to zero learning based on the autoencoder paradigm. Specifically, semantic similarities of all classes are calculated and serve as supervision to guide the classification branch in our autoencoder model. We decompose the relations of the semantic space to semantic similarities among classes. Our aim is to preserve these similarities in the visual feature space so as to appropriately inherit the structure of the semantic space. Unlike other methods just using indirect semantic relations between seen classes and unseen classes, our model can directly and indirectly utilize these semantic relations. The relations are directly used by calculating the semantic similarities between seen classes and unseen classes and indirectly used by obtaining the similarities between images features and projected semantic embeddings of all classes.

Our contributions are:

- We propose an autoencoder model with a classification branch which is trained by semantic similarity for ZSL.
- Our model can directly and indirectly utilize semantic relations between seen classes and unseen classes.
- Our experimental results on two ZSL datasets show a significant improvement.

2. Related Work

The fourth method mentioned in Sect. 1 is widely used in today's ZSL. The CLSWGAN [5] model uses a pretrained classifier to guide their generation of visual features of seen classes. The Cycle-CLSWGAN [6] model, which is based on the CLSWGAN model, adds a reconstruction constrain on semantic embeddings to preserve semantic compability between visual features and semantic embeddings. The CADA-VAE [7] model tries to preserve semantic relations in a low-dimensional immediate space for both visual feature vectors and semantic embeddings through reconstruction and cross-reconstruction criterion. The f-VAEGAN-D2 [8] model is a conditional generative model that combines the strength of VAE and GANs.

Our model differs from DCN [9] model in three aspects. First, the embedding space is different. they project images and attributes to an intermediate space while we project images and attributes to the visual feature space in order to alleviate the hubness problem. Second, our reconstruction model can ensure that our projected embeddings preserve all the semantic information contained in original semantic embeddings while DCN model cannot. Third, They use one-hot labels to supervise classification of seen classes which is different from our model wherein semantic

Manuscript received September 19, 2019.
Manuscript revised December 30, 2019.
Manuscript publicized March 3, 2020.

[†]The authors are with School of Aeronautics and Astronautics, Zhejiang University, Hangzhou, 310027, P. R. China.
a) E-mail: zheminglu@zju.edu.cn (Corresponding author)

DOI: 10.1587/transinf.2019EDL8176

similarities serve as supervision of all classes in classification.

3. Proposed Method

First, the definition of ZSL is as follows. Given an visual feature set $X = X_{\text{tr}} \cup X_{\text{te}}$, where X_{tr} is the training visual feature set and X_{te} is the testing visual feature set. In coventional ZSL setting, visual features in X_{te} only come from unseen classes, while they may come from seen classes or unseen classes in generalized ZSL(GZSL) setting. Semantic embeddings for all classes is $A = \{\boldsymbol{a}_i\}_{i=1}^{\text{N}} = A_{\text{s}} \cup A_{\text{u}}$, where $A_{\text{s}} = \{\boldsymbol{a}_i\}_{i=1}^{\text{N}_{\text{s}}}$ is the semantic embedding set of seen classes and $A_{\text{u}} = \{\boldsymbol{a}_i\}_{i=1}^{\text{N}_{\text{u}}}$ is for unseen classes, N_{s} is the number of seen classes, N_{u} is the number of unseen classes, and N is the number of all classes, which equals to N_{s}+N_{u}. The label set of X_{tr} is Y_{tr} and the label set of X_{te} is Y_{te}.

The architecture of our proposed model is shown in Fig. 1. It consists of an encoder $F(\boldsymbol{a})$ and a decoder $G(\boldsymbol{p})$, where $\boldsymbol{a}$ is a semantic embedding and $\boldsymbol{p}$ is a class prototype in visual space, which is equal to $F(\boldsymbol{a})$.

Our learning objective has two parts. One is the reconstruction objective to ensure that a semantic embedding $\boldsymbol{a}_i$ belonging to i-th class can be mapped to i-th class prototype $\boldsymbol{p}_i$ and this class prototype is similar to a visual feature $\boldsymbol{x}_i$ of i-th class. Specifically,

$$L_1 = \left\|\boldsymbol{x}_i - \boldsymbol{p}_i\right\|_2^2 \tag{1}$$

where $\boldsymbol{p}_i = F(\boldsymbol{a}_i)$. In order to make sure that $\boldsymbol{p}_i$ contains semantic information of its class, it should be able to be restored to its original semantic embedding $\boldsymbol{a}_i$. Specifically:

$$L_2 = \|\boldsymbol{a}_i - \tilde{\boldsymbol{a}}_i\|_2^2 \tag{2}$$

where $\tilde{\boldsymbol{a}}_i = G(F(\boldsymbol{a}_i))$. Our second objective needs $\boldsymbol{p}_i$ to be unlike visual feature vectors of other classes. To boost the discriminative power of class prototypes, we add a classification branch to the autoencoder and use semantic similarity vectors to supervise its training. Let s_{ij} be a semantic similarity measure between different class embeddings. In order to calculate the softmax probability stably, we subtract the

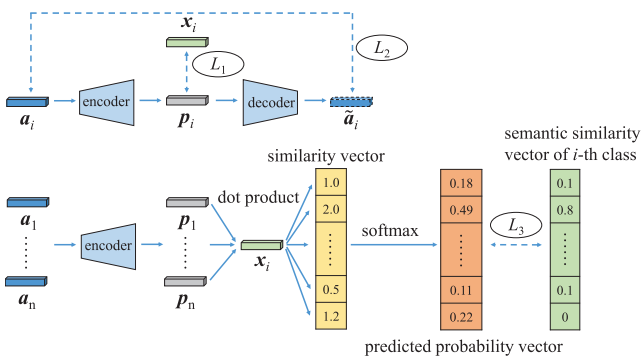

Fig. 1 Architecture of our model.

the mean value from semantic similarities. Specifically,

$$t_{ij} = \boldsymbol{a}_i \cdot \boldsymbol{a}_j - \frac{1}{\text{N}}\sum_{j=1}^{\text{N}} \boldsymbol{a}_i \cdot \boldsymbol{a}_j \tag{3}$$

$$s_{ij} = \frac{\exp(\tau t_{ij})}{\sum\limits_{j=1}^{\text{N}} \exp(\tau t_{ij})} \tag{4}$$

where τ is the temperature, when $\tau \to 0$, $s_{ij} \to 1/(\text{N})$, which leads to a uniform distribution of s_{ij} with same i. As $\tau \to \infty$, only s_{ii} is 1 and other s_{ij} is 0. Then, we define $\boldsymbol{q}_i = (s_{i1}, s_{i2}, \ldots, s_{i\text{N}})$ as the semantic similarity vector of i-th class. We consider the softmax of dot products between $\boldsymbol{x}_i$ and $\boldsymbol{p}_j$ as the predicted category probability of $\boldsymbol{x}_i$. Finally, the cross-entropy loss between $\boldsymbol{q}_i$ and the predicted probability is considered as our classificiation loss to trian the classification branch:

$$P(j|\boldsymbol{x}_i, A) = \frac{\exp(\boldsymbol{x}_i \cdot F(\boldsymbol{a}_j))}{\sum\limits_{k=1}^{\text{N}} \exp(\boldsymbol{x}_i \cdot F(\boldsymbol{a}_k))} \tag{5}$$

$$L_3 = \sum_j -s_{ij} \log P(j|\boldsymbol{x}_i, A) \tag{6}$$

To sum up, our model minimizes the following objective function during training:

$$L_{all} = \min_{F,G} \frac{1}{|B|} \sum_B (L_1 + L_2 + L_3) \tag{7}$$

Here $|B|$ refers to the size of a mini-batch B.

In the testing stage, given a test sample $\boldsymbol{x}_i$, we infer its class as follows:

$$y^* = \arg\max_j \boldsymbol{x}_i \cdot F(\boldsymbol{a}_j) \tag{8}$$

4. Experimental Settings and Datasets

Our encoder and decoder are both implemented as 2 fully connected layers with 1800 hidden units. We use LeakyReLU as the nonlinear activation function. As for the optimization, we adopt Adam optimizer with a constant learning rate 0.0001 and our training mini-batch is 64.The temperature parameter is fine-tuned with a cross-validation procedure in the split of train and validation provided by [10]. Specifically, we have found that the proposed model works well when the temperature is set as 125.

We test our proposed model in two ZSL datasets, CUB-200-2011 [11] and FLO [12]. For both CUB and FLO, split

Table 1 Information about datasets CUB and FLO.

Dataset	N_{s}	N_{u}	$\lvert X_{\text{tr}}\rvert$	$\lvert X_{\text{te}}^{\text{u}}\rvert$+$\lvert X_{\text{te}}^{\text{s}}\rvert$
CUB [11]	150	50	7057	1764+2967
FLO [12]	82	20	1640	1155+5394

Table 2 Results on conventional ZSL and generalized ZSL (bold indicates best results).

	FLO				CUB			
Model name	T1(%)	u(%)	s(%)	H(%)	T1(%)	u(%)	s(%)	H(%)
CLSWGAN [5]	67.2	59.0	73.8	65.6	57.3	43.7	57.7	49.7
Cycle-CLSWGAN [6]	70.3	61.6	69.2	65.2	58.6	47.9	59.3	53
CADA-VAE [7]	-	-	-	-	-	**51.6**	53.5	52.4
f-VAEGAN-D2 [8]	67.7	56.8	74.9	64.6	61	48.4	60.1	53.6
Ours (AE)	38.7	13.7	13.6	13.6	35.8	14.1	17.2	15.5
Ours (SSAE)	**77.6**	**68.0**	**80.7**	**73.8**	**65**	50.8	**62.8**	**56.2**

methods in [6] are used. For fair comparison, we use visual features and semantic embeddings provided by [6] to evaluate our model as well. Visual features are 2048-dimensional vectors extracted by ResNet-101 for both datasets. Semantic embeddings have 1024 dimensions produced by CNN-RNN. More information about datasets in terms of the number of seen classes, unseen classes, the number of training and testing images can be found in Table 1. $|X_{tr}|$ is the number of training images. $\left|X^{u}_{te}\right|$ and $\left|X^{s}_{te}\right|$ represent the number of testing images belonging to unseen classes and seen classes respectively.

We follow the evaluation protocol proposed by [10] to evaluate our model. For the conventional ZSL setting, the averaged top-1 accuracy for each unseen class is computed, denoted as **T1**. For the GZSL setting, the averaged accuracy of seen classes is denoted by **s** and the averaged accuracy of unseen classes is denoted by **u**, and their harmonic mean defined as $\mathbf{H} = 2\mathbf{su}/(\mathbf{s}+\mathbf{u})$.

5. Results and Analysis

We compare our model with four state-of-the-art (SOTA) methods, i.e., CLSWGAN [5], Cycle-CLSWGAN [6], CADA-VAE [7] and f-VAEGAN-D2 [8] in Table 2, because we use the same semantic embeddings as they do. We also conduct experiments for ablation analysis. First, we remove the classification branch to show the its importance to the performance of our model. Then, we set different temperatures to show the influence of semantic relations provided by semantic similarity vectors. Model without our classification branch is called AE in Table 2. Our whole model containing the classification branch is referred as SSAE in Table 2.

In Table 2, our model apparently establishes new SOTA results on both datasets in the ZSL setting. Our model achieves 77.6% on FLO, and 65.0% on CUB in terms of **T1**, which is 7.3% and 4% higher than previous SOTA, 70.3% and 61%. In the more challenging GZSL setting, our model gets better results than others on both datasets in terms of **H** as well. On FLO, the performance of our model is 73.8% in terms of **H**, which achieves significantly higher result than the previous best 61%. On CUB, our result 56.2% is 2.6% higher than SOTA result 53.6% in terms of **H**.

Our model outperforms other models in Table 2, because we use semantic similarity vectors to guide the training of our autoencoder and our model can utilize semantic embeddings of unseen classes during training. The CLSWGAN model uses a pretrained classification module to guide their generation module, which can not use semantic embeddings of unseen classes. The Cycle-CLSWGAN model is based on the CLSWGAN model, so it has CLSWGAN model's shortcoming as well. The f-VAEGAN-D2 model combines the benefits of the variational autoencoder and GAN, but their model does not contain a classification module to guide their generation model to generate different visual features. It can be seen that CADA-VAE model performs best in the terms of **u**, because their model can learn a encoding that retains the information contained in all modalities they used.

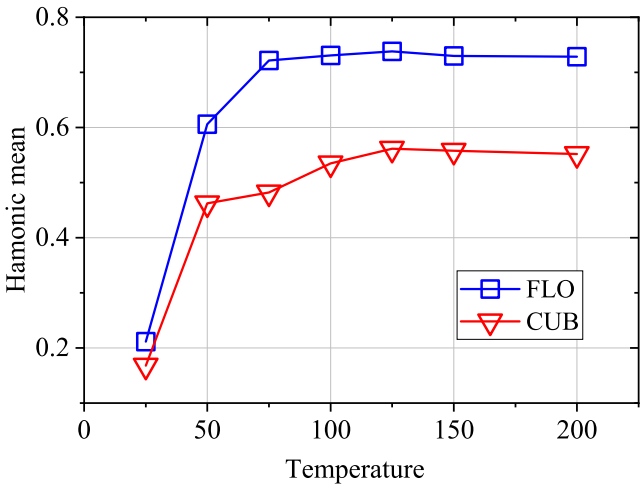

Fig. 2 **H** on CUB and FLO with increasing temperature.

In ablation study, as we expected our AE model does not perform well, which is 38.7% and 35.8% in terms of **T1** in the ZSL setting on FLO and CUB in Table 2. In the GZSL setting, its **H** is well below the average, confirming that it is important to add a classification branch to our model. In Fig. 2 we show the performance of our model under different temperaures. As shown in Fig. 2 both for CUB and FLO, our model has a significant edge when the temperature is small, for example, 25 and 50. This shows that with the increasing of the temperature, semantic similarity vectors provide our model with more information about semantic relations. Going towards the large tempearture, the performance keeps stable and reaches the maximum at 125 for both datasets. This is expected since semantic similarity vectors can provide more information about semantic relations than one-hot labels.

6. Conclusion

In this work, we focus on utilizing semantic similarity to assist the autoencoder to map from the semantic space to the visual space correctly. We add a classification branch to our main autoencoder model and calculate semantic similarity vectors to guide its training. Evaluations on different settings of the proposed model are carried out and results outperforming state-of-the-art methods are achieved. Our results show the importance of our classificaiton branch and semantic information provided by our semantic similarity vectors.

References

[1] A. Frome, G.S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov, "Devise: A deep visual-semantic embedding model," Advances in Neural Information Processing Systems, on Lake Tahoe, Nevada, pp.2121–2129, Dec. 2013.

[2] V.K. Verma and P. Rai, "A simple exponential family framework for zero-shot learning," Joint European Conference on Machine Learning and Knowledge Discovery in Databases, on Skopje, Macedonia, pp.792–808, Sept. 2017. DOI:10.1007/978-3-319-71246-8_48

[3] H. Jiang, R. Wang, S. Shan, and X. Chen, "Learning class prototypes via structure alignment for zero-shot recognition," ECCV, on Munich, Germany, pp.118–134, Sept. 2018. DOI:10.1007/978-3-030-01249-6_8

[4] E. Kodirov, T. Xiang, and S. Gong, "Semantic autoencoder for zero-shot learning," CVPR, on Honolulu, Hawaii, pp.3174–3183, July 2017. DOI:10.1109/CVPR.2017.473

[5] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, "Feature generating networks for zero-shot learning," CVPR, on Salt Lake City, Utah, pp.5542–5551, June 2018. DOI:10.1109/CVPR.2018.00581

[6] R. Felix, B.G.V. Kumar, I. Reid, and G. Carneiro, "Multi-modal cycle-consistent generalized zero-shot learning," ECCV, on Munich, Germany, pp.21–37, Sept. 2018. DOI:10.1007/978-3-030-01231-1_2

[7] E. Schönfeld, S. Ebrahimi, S. Sinha, T. Darrell, and Z. Akata, "Generalized zero- and few-shot learning via aligned variational autoencoders," CVPR, on Long Beach, CA, pp.8247–8255, June 2019.

[8] Y. Xian, S. Sharma, B. Schiele, and Z. Akata, "f-VAEGAN-D2: a feature generating framework for any-shot learning," CVPR, on Long Beach, CA, pp.10275–10284, June 2019.

[9] S. Liu, M. Long, J. Wang, and M.I. Jordan, "Generalized zero-shot learning with deep calibration network," NIPS, Montréal, Canada, pp.2005–2015, Dec. 2018.

[10] Y. Xian, C.H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly," IEEE Trans. Pattern Anal. Mach. Intell., vol.41, no.9, pp.2251–2265, 2018. DOI:10.1109/TPAMI.2018.2857768

[11] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 dataset," California Institute of Technology, http://www.vision.caltech.edu/visipedia/CUB-200-2011.html, accessed Nov. 2011.

[12] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," Indian Conference on Computer Vision, Graphics & Image Processing, on Bhubaneswar, India, pp.722–729, Dec. 2008. DOI:10.1109/ICVGIP.2008.47