Neural Networks 132 (2020) 333–341

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Multi-label zero-shot learning with graph convolutional networks

Guangjin Ou [a,b], Guoxian Yu [a,b,c,*], Carlotta Domeniconi [d], Xuequan Lu [e], Xiangliang Zhang [c]

[a] School of Software, Shandong University, Jinan, China
[b] College of Computer and Information Sciences, Southwest University, Chongqing, China
[c] CEMSE, King Abdullah University of Science and Technology, Thuwal, SA, Saudi Arabia
[d] Department of Computer Science, George Mason University, Fairfax, VA, USA
[e] School of Information Technology, Deakin University, Australia

ARTICLE INFO

Article history:
Received 19 June 2020
Received in revised form 21 August 2020
Accepted 11 September 2020
Available online 21 September 2020

Keywords:
Zero-shot learning
Graph Convolutional Networks
Multi-label classification
Label correlations

ABSTRACT

The goal of zero-shot learning (ZSL) is to build a classifier that recognizes novel categories with no corresponding annotated training data. The typical routine is to transfer knowledge from seen classes to unseen ones by learning a visual-semantic embedding. Existing multi-label zero-shot learning approaches either ignore correlations among labels, suffer from large label combinations, or learn the embedding using only local or global visual features. In this paper, we propose a Graph Convolution Networks based Multi-label Zero-Shot Learning model, abbreviated as MZSL-GCN. Our model first constructs a label relation graph using label co-occurrences and compensates the absence of unseen labels in the training phase by semantic similarity. It then takes the graph and the word embedding of each seen (unseen) label as inputs to the GCN to learn the label semantic embedding, and to obtain a set of inter-dependent object classifiers. MZSL-GCN simultaneously trains another attention network to learn compatible local and global visual features of objects with respect to the classifiers, and thus makes the whole network end-to-end trainable. In addition, the use of unlabeled training data can reduce the bias toward seen labels and boost the generalization ability. Experimental results on benchmark datasets show that our MZSL-GCN competes with state-of-the-art approaches.

1. Introduction

In conventional supervised learning, massive manually annotated data is required to train a model. However, the ever-increasing need for fine-grained annotations, the consistent emergence of new classes, and the exponential growth of large-scale datasets have made the manual annotations of data very costly and difficult to achieve. For example, when classifying a rare species of animals, the number of available labeled images is far from being sufficient to build reliable classifiers. Transfer learning has been introduced as a paradigm to deal with this fundamental problem (Pan & Yang, 2009). It learns using a limited number of classes, and transfers knowledge to classify data from new classes, either using only few labeled data points (i.e., few- and one-shot learning Li, Fergus, & Perona, 2006), or in the extreme case without any labeled data (i.e., zero-shot learning (ZSL) Xian, Lampert, Schiele, & Akata, 2019). In this paper, we focus on the more challenging ZSL setting, which has been studied recently in face verification, object recognition, video annotation, and other domains (Xian et al., 2019). ZSL aims at recognizing objects whose instances may not have been seen during training. ZSL distinguishes between two types of categories, *seen* and *unseen*, where labeled data is available only for seen categories.

The key of ZSL is to transfer knowledge from the seen classes to the target unseen classes via the semantics of labels. Existing ZSL methods assume that each class prototype is embedded in a semantic space (e.g., attribute space Lampert, Nickisch, & Harmeling, 2009, 2013, or in a word vector space Frome, et al., 2013; Liu, et al., 2019). In such space, each class name can be represented by a high-dimensional binary vector based on a manually-defined object ontology, or by a numeric vector based on a huge text corpus. In this way, the semantic relatedness between the seen and unseen labels is established. Compared to the extensively studied multi-class single-label ZSL (Xian et al., 2019), the more challenging multi-label ZSL (MZSL) has received far less attention (Lee, Fang, Yeh, & Wang, 2018; Mensink, Gavves, & Snoek, 2014), due to the enormous label combinations and more complex mappings between labels and visual features. Existing MZSL methods suffer from two issues: (i) they globally project the whole image and ignore the local features, and thus cannot differentiate the subtle difference between the projected visual vectors of the seen images (e.g., tiger) and the unseen

[*] Corresponding author at: School of Software, Shandong University, Jinan, China.
E-mail addresses: gjou@swu.edu.cn (G. Ou), guoxian85@gmail.com (G. Yu), carlotta@cs.gmu.edu (C. Domeniconi), xuequan.lu@deakin.edu.au (X. Lu), xiangliang.zhang@kaust.edu.sa (X. Zhang).

https://doi.org/10.1016/j.neunet.2020.09.010
0893-6080/© 2020 Elsevier Ltd. All rights reserved.

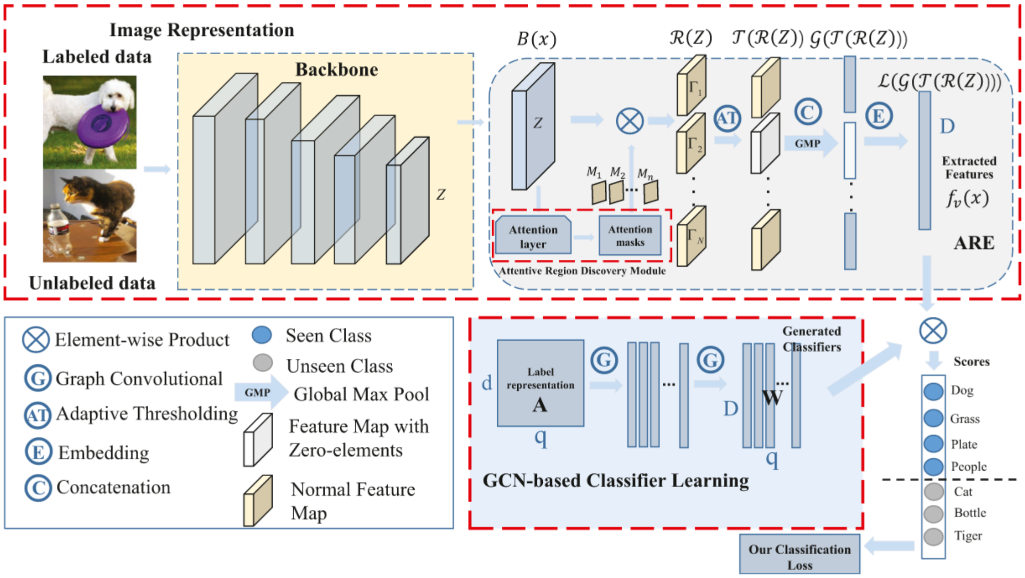


Fig. 1. The framework of MZSL-GCN. In the image subnet, the input image **x** is fed into the backbone net to generate feature map **Z**, which is used as the input for the attentive region discovery module and produces N attentive feature maps $\{\Gamma_n\}_{n=1}^{N}$ to capture different local visual features. Besides, we fix the last $\mathbf{M}_n$ as a matrix with all 1s to preserve the global features. In the GCN-based multi-label classifier learning subnet, stacked GCNs are adopted on the label graph to map the label semantic embeddings ($\mathbf{A} \in \mathbb{R}^{q\times d}$) into a set of inter-dependent classifiers ($\mathbf{W} \in \mathbb{R}^{q\times D}$) and with respect to visual features. Both the labeled data of seen classes and unlabeled data of unseen labels are used to train the model.

ones (e.g., leopard) (Lee et al., 2018; Shao, Guo, Ding, & Han, 2018); (ii) they ignore the global label correlations, which are crucial in multi-label classification (Zhu, Kwok, & Zhou, 2018). In essence, these methods either use simple label co-occurrence statistics (Mensink et al., 2014) or train an independent classifier for each class (Gaure, Gupta, Verma, & Rai, 2017; Zhang, Gong, & Shah, 2016). Those issues greatly restrict the applicability of these methods in effective multi-label zero-shot learning, where complicated correlations exist among labels.

To address these issues, we propose an end-to-end model, called MZSL-GCN (as illustrated in Fig. 1), to capture the correlations between labels using Graph Convolutional Networks (GCNs) (Kipf & Welling, 2016) for multi-label zero-shot learning. Instead of treating multiple binary classifiers as a set of independent parameter vectors to be learned, we consider the crucial label correlations by sharing all the mapping parameters of label semantic embedding-to-classifier to train inter-dependent classifiers via the GCN subnet. The learned classifiers are then applied to visual features generated by the image representation subnet to compute the probability of each class. In this way, we can jointly optimize the parameters of GCN subnet and image representation subnet within a unified network in a coherent fashion. Discovering salient regions of an image can better account for knowledge transfer from seen labels to unseen ones, hence in the image representation subnet we adopt an attention mechanism to automatically find discriminate semantic regions (i.e., local parts). Besides the local features, we also preserve the global features to avoid semantic information loss by fixing the weights of the last attention mask to ones. The joint extraction of local and global features is useful for bridging the image representation and label semantics. In the training phase, we use the data of seen and unseen labels (without annotated data), and further incorporate a balance loss term to alleviate the preference toward seen classes. It is worth mentioning that the GCN-based classifier learning subnet generates classifiers for both seen and unseen labels. As a result, the classification scores on both seen and unseen classes can be obtained directly. When the labeled data of unseen labels becomes available, our model can be incrementally updated using these data.

Our key contributions are listed as follows:

- We propose a novel end-to-end trainable multi-label zero-shot learning (MZSL-GCN) framework, which integrates GCN to explore and capture label correlations and to learn a set of inter-dependent classifiers. We also propose another network to learn adaptive local and global visual features and to coordinate the training of inter-dependent classifiers.
- We propose a correlation matrix which considers both label co-occurrence and semantic similarity to compensate for unseen labels during the training phase.
- Experimental results on benchmark datasets show that MZSL-GCN competes with state-of-the-art methods (Lee et al., 2018; Shao et al., 2018; Weston, Bengio, & Usunier, 2011; Zhang et al., 2016). It also achieves the state-of-the-art results in conventional multi-label classification.

2. Related works

Our work has close connections with the popular multi-label learning (Zhang & Zhou, 2014) and ZSL (Xian et al., 2019).

Multi-label learning (MLL) aims at inducing a classifier to assign a set of non-exclusive labels to samples, it is a fundamental and practical task in various domains. Compared with ZSL, multi-label learning has been more widely studied. Most multi-label learning algorithms investigate how to make use of label correlations to boost the performance (Read, Pfahringer, Holmes, & Frank, 2011; Tsoumakas, Katakis, & Vlahavas, 2010; Zhu et al., 2018); some try to leverage labeled and unlabeled data (Chen, Song, Wang, & Zhang, 2008; Tan, Yu, Yu, & Wang, 2017); others learn from multi-label data with missing labels or with noisy labels (Sun, Zhang, & Zhou, 2010; Tan, Yu, Domeniconi, Wang, & Zhang, 2018; Yu, et al., 2018). However, all these MLL methods assume the labels of training data and those of testing data are seen during the training stage, and they cannot handle emerging new labels. Interesting readers of MLL are referred to Gibaja and Ventura (2015) and Zhang and Zhou (2014) for a comprehensive review.

ZSL aims at inducing a classifier to recognize novel classes without acquiring new training data but using the class attributes. Direct attribute prediction (DAP) trains attribute classifiers and then calculates the posterior of a test class for a given sample (Lampert et al., 2009). However, it handles attributes individually. Most ZSL methods learn a projection to map the visual features onto a common embedding space. For example, Attribute Label Embedding (ALE) (Akata, Perronnin, Harchaoui, & Schmid, 2013) uses the pairwise-ranking loss to learn the bi-linear compatibility function between the visual feature and attribute spaces. Other representative embedding-based methods (Frome, et al., 2013; Romera-Paredes & Torr, 2015) optimize the compatibility function from different perspectives (e.g., regularization term and embedding space). For example, Embarrassingly Simple ZSL (ESZSL) (Romera-Paredes & Torr, 2015) uses the square loss to learn the bi-linear compatibility and explicitly regularizes the objective by Frobenius norm. Deep neural networks have also been integrated to further boost ZSL (Tong, Wang, Klinkigt, Kobayashi, & Nonaka, 2019). Quasi-Fully Supervised Learning (QSFL) (Song, Shen, Yang, Liu, & Song, 2018) alleviates the prediction bias to seen classes by increasing the probability of being unseen ones. Latent Discriminative Features learning (LDF) (Li, Zhang, Zhang, & Huang, 2018) automatically discovers discriminative regions by a zoom network. Discriminative Latent Features for Zero-shot Learning (DLFZRL) (Tong et al., 2019) learns discriminative and generalizable representations with deep autoencoder. These approaches focus on the single-label scenario and can only assign one label to an instance.

Unlike the aforementioned ZSL methods, multi-label zero-shot learning (MZSL) considers a more complex scenario and tries to simultaneously assign several unseen and non-exclusive labels to a new sample. Several attempts have been made toward this challenging task. Co-Occurrence statistics (COSTA) (Mensink et al., 2014) uses co-occurrences of visual concepts and estimates a classifier for a new label as a weighted combination of related seen labels. However, the noise interference in co-occurrence statistics is inevitable with sparse labels. For real-world datasets, the co-occurrence often happens among labels that have large semantic and visual difference (Shao et al., 2018). Zero-shot Multi-Label Predictor (ZS-MLP) extends traditional ZSL to MZSL by treating each label set as a single label (Fu, Yang, Hospedales, Xiang, & Gong, 2014). As a result, it suffers from not only a high computational complexity due to the combinatorial nature of the output space, but also from a poor performance for scanty data with respect to each label set. Fast Zero-shot image Tagging (Fast0Tag) separates relevant and irrelevant labels by learning principal directions for image data in the embedding space (Zhang et al., 2016), but it ignores the label correlations. Multiple instance Visual-Semantic Embedding (MiVSE) (Ren, Jin, Lin, Fang, & Yuille, 2015) uses a region-proposal method to detect salient regions in images and then maps the regions (local features) to their corresponding labels in the semantic embedding space. Consequently, its performance relies on the region-proposal method. Multi-label Zero-Shot Learning with Knowledge Graph (MZSL-KG) uses structured knowledge graphs to describe the relationships between multiple labels and to exploit correlations between seen and unseen labels (Lee et al., 2018). But it ignores the local visual features, highly depends on the constructed knowledge graph and initial beliefs induced from the base classifier. Label factorization with regularized least squares (LFRLS) (Shao et al., 2018) learns a shared latent space by label factorization and uses the label semantics as the decoding function, but its performance deteriorates as the number of categories increase. Besides, multi-label zero-shot/few-shot learning had been proposed for gene function prediction by mining hierarchical label correlations (Yu, Zhu, & Domeniconi, 2015; Yu, Zhu, Domeniconi, & Liu, 2015).

In this paper, we introduce MZSL-GCN to learn interdependent classifiers for seen and unseen classes using GCNs and word embeddings of semantic labels. In addition, we incorporate the attention strategy to automatically extract both local and global visual features of input data to guide the semantic-label embeddings. Given that, it overcomes the issues suffered by existing solutions and achieves a more competitive performance than them.

3. Proposed method

3.1. Problem formulation

Let us consider a training set $\mathcal{D}^s = \{\mathbf{X}^s, \mathbf{Y}\}$, where $\mathbf{X}^s \in \mathbb{R}^{n_s \times d}$ represents d-dimensional n_s training instances, and $\mathbf{Y} \in \{0, 1\}^{n_s \times s}$ denotes the label matrix across a set of seen classes. $\mathbf{Y}_{ic} = 1$ if the ith instance is annotated with the cth seen label; $\mathbf{Y}_{ic} = 0$ otherwise. $\mathcal{S} = \{1, 2, \dots, s\}$ is the seen label set of the training data. With multi-label data, each row of $\mathbf{Y}$ may have more than one entry with a value of 1. A test set $\mathcal{D}^t = \{\mathbf{X}^t\}$ is also given. The labels of samples in $\mathcal{D}^t$ are not available for training. For ZSL, we assume there is a set of unseen classes, $\mathcal{U} = \{s+1, s+2, \dots, s+u\}$ such that $q = s + u$, and none of the labels in $\mathcal{U}$ appears in the labeled training data. Additionally, we assume that the semantic embedding of the seen and unseen classes are available as $\mathbf{A} = [\mathbf{A}_s; \mathbf{A}_u] \in \mathbb{R}^{q \times m}$, where $\mathbf{A}_s \in \mathbb{R}^{s \times m}$ is the embedding for seen classes, and $\mathbf{A}_u \in \mathbb{R}^{u \times m}$ is the embedding of unseen ones. We aim to learn a multi-label classification model using the training data to achieve good performance not only in the conventional setting (prediction on unseen labels), but also in the generalized setting (prediction on both seen and unseen labels) of MZSL. The overall framework of our MZSL-GCN is shown in Fig. 1, which is composed of two sub-nets, namely the image representation sub-net and the GCN learning sub-net. Next, we explain the two sub-nets in detail.

3.2. GCN learning

GCN (Kipf & Welling, 2016) has been widely-used in semi-supervised learning and network representation learning, it updates the node representations by propagating information among connected nodes. We construct a graph to model the inter dependency between labels, which captures the topological structure of the label space. Specifically, we represent each node (label) of the graph as word embeddings of the label, and propose to use GCN to map these label embeddings into a set of interdependent classifiers, which can be applied to image classification. Two factors motivate the design of GCN learning part. Firstly, as the parameters of embedding-to-classifier mapping are shared on all classes, the learned classifiers can retain the weak semantic structures in the word embedding space, where semantic related concepts are close to each other. Meanwhile, the gradients of all classifiers can impact the classifier generation function, which implicitly models the label dependency (both seen classes and unseen ones). Second, we design a novel label correlation matrix based on their co-occurrence patterns and semantic similarity to explicitly model the label dependency by GCN, with which the update of node features will absorb information from correlated nodes (labels).

We extend GCNs to learn a classification model $f(\cdot, \cdot)$ that takes the class label embeddings $\mathbf{A}^l \in \mathbb{R}^{q \times m}$ and the label correlation matrix $\mathbf{S} \in \mathbb{R}^{q \times q}$ as inputs, where m indicates the dimensionality of the word embeddings. The updating rule of GCN can be written as:

$$\mathbf{A}^{l+1} = f(\mathbf{A}^l, \mathbf{S}), \tag{1}$$

where $\mathbf{A}^{l+1}$ is the updated label representation.

After applying the convolution operation, it can be further represented as

$$\mathbf{A}^{l+1} = \phi(\mathbf{S}\mathbf{A}^{l}\mathbf{P}^{l}), \tag{2}$$

where $\mathbf{P}^{l}$ is the to-be-learned transformation matrix, and $\phi(\cdot)$ denotes a non-linear activation function (LeakyRelu is used in this work).

In our model, we consider higher-order label correlations. As such, the convolution on nodes of the label graph $\mathbf{S}$ depends on the nodes that are at H steps away from the target node. In other words, the output signals of the convolution are defined by a H-order approximation of localized spectral filters on networks. Thus, the convolution operation is further formulated as:

$$\mathbf{A}^{l+1} = \phi(\sum_{h=1}^{H}\mathbf{S}^{h}\mathbf{A}^{l}\mathbf{P}^{l}). \tag{3}$$

Through stacking multiple GCN layers, we can model the complex inter-relationships among classes (see Fig. 1).

We design the final output of each GCN node to be the classifier of the corresponding label in our task. Here we discuss the training of inter-dependent classifiers for q labels, i.e., $\mathbf{W} = \{\mathbf{w}_c\}_{c=1}^{q}$ via a GCN-based mapping function $f(\cdot,\cdot)$. We use stacked GCNs, where the new label representation $\mathbf{A}^{l+1}$ is updated by the input $\mathbf{A}^{l}$. For the first layer, we use a 300-D GloVe word embedding (Pennington, Socher, & Manning, 2014) vector $\mathbf{A} \in \mathbb{R}^{q\times 300}$ as the input. For the last layer, the output $\mathbf{W} \in \mathbb{R}^{q\times D}$ can be seen as a classifier for q labels with respect to the D-dimensional sample representation. In this way, we can obtain the predicted score as

$$\tilde{\mathbf{y}} = \mathbf{W}f_v(x), \tag{4}$$

where $f_v(\mathbf{x}) \in \mathbb{R}^{D}$ is the representation of image $\mathbf{x}$, which will be introduced later.

Now we describe how to get the correlation matrix $\mathbf{S}$. We estimate the label correlation matrix $\mathbf{S}$ by mining label co-occurrence patterns in the dataset. Let n^{c} denote the number of training samples with label c, and let n^{cs} be the number of training samples annotated with both labels c and s. Then the estimated label correlation is n^{cs}/n^{c}. When co-occurrences are rare, this estimation may be inaccurate. In addition, the co-occurrences in the training data may differ from those in the testing data. In this case, the correlation matrix over-fits the pattern in the training data and thus reduces the generalization ability of the model. To alleviate this problem, we define a binary correlation matrix via a threshold τ to filter out noisy edges, as shown in Eq. (5).

$$\mathbf{S}^{1}_{cs} = \begin{cases} 0, & \text{if } n^{cs}/n^{c} < \tau \\ 1, & \text{if } n^{cs}/n^{c} \geq \tau \end{cases} \tag{5}$$

where $\mathbf{S}^{1}$ is the binary version of the label correlation matrix.

Our model can be regarded as a quasi-fully supervised classifier (Song et al., 2018), since the nodes in the graph correspond to both seen and unseen classes, and the model should be able to predict input data with respect to each class. However, the unseen labels do not appear in the training data, so we cannot obtain seen-to-unseen and unseen-to-unseen co-occurrence statistics. As an alternative, we adopt the semantic similarity to construct a label similarity matrix. Specifically, we use the 300-D GloVe embedding to calculate the Euclidean distance between labels and setup the binary similarity matrix as Eq. (6):

$$\mathbf{S}^{2}_{cs} = \begin{cases} 1, & \text{if } c \in \mathcal{N}_v(s) \text{ or } s \in \mathcal{N}_v(c) \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $\mathcal{N}_v(c)$ denotes the set of v nearest neighbors of class c. To build the complete correlation matrix $\mathbf{S}$ used in Eq. (1), we combine the co-occurrence statistics with the semantic similarity in the following simple yet effective way:

$$\mathbf{S}_{cs} = \begin{cases} \frac{\mathbf{S}^{1}_{cs}+\mathbf{S}^{2}_{cs}}{2}, & \text{if } c \in \mathcal{S} \text{ and } s \in \mathcal{S} \\ \mathbf{S}^{2}_{cs}, & \text{otherwise} \end{cases} \tag{7}$$

3.3. Image representation

The discrimination power implied in local regions (paragraphs) of images (documents) often corresponds to specific semantic information, and thus they can assist the semantic transfer between seen/unseen classes. Attention Region Embedding (ARE) can capture discriminant regions automatically, without any part-level manual annotation (Xie, et al., 2019). Take the image input data for example, we can use any CNN-based model as the backbone network to learn the representation features of an image. For other data types (i.e., documents), other network structures (i.e., RNN) can also be adopted here.

In Fig. 1, we feed the last convolutional feature map $\mathbf{Z}$ of the backbone (ResNet101 in the experiment) for image $\mathbf{x}$ to ARE (shown in upper right of Fig. 1). The first part of ARE is the Attention Region Discovery (ARD) module, followed by an adaptive thresholding (AT) procedure. Through the ARD, the attention regions can be effectively highlighted, and the AT operation can filter out the ones with low attentive strength. Afterward, we exploit the global maximum pooling (GMP) for these feature maps and then concatenate them. Last, a fully connected layer is used to control the dimension of the sub-net, and to fuse both local and global features to form the final image representation. The vision embedding can be formulated as follows:

$$f_v(\mathbf{x}) = \mathcal{L}(\mathcal{G}(\mathcal{T}(\mathcal{R}(\mathbf{Z})))), \quad \mathbf{Z} = \mathcal{B}(\mathbf{x}), \tag{8}$$

where $\mathcal{B}$, $\mathcal{R}$, $\mathcal{T}$, $\mathcal{G}$ and $\mathcal{L}$ are the backbone network operation, the ARD operation, the AT operation, the GMP operation, and the fully connected operation, respectively.

We use the attention mechanism to automatically discover the important regions of an input image $\mathbf{x}$, which serves as the bridges for semantic transfer at the region level. Suppose $\mathbf{Z}$ is the last convolutional feature map of the backbone net. $\mathbf{Z} \in \mathbb{R}^{H\times W\times C}$ is a 3D tensor, where C is the number of channels, and H and W are the height and width of a channel. Let $z(h, w, c) \in \mathbb{R}$ be the value in location (h, w) of the cth channel from $\mathbf{Z}$. We further denote the number of regions as N. Some of these regions can be matched with the specific semantic attributes and help the classification task (Xie, et al., 2019). Inspired by the attention models developed in various fields, we design an attention mechanism to capture the semantic regions and to further narrow the semantic gap between seen/unseen images. We first generate N 2D masks $\mathbf{M}_n \in \mathbb{R}^{H\times W}(n = 1, 2, \ldots, N)$:

$$\mathbf{M}_n = \mathcal{M}_{MG_n}(\mathbf{Z}), \tag{9}$$

where $\mathcal{M}_{MG_n}$ is a mask generation operation. Instead of several fully connected layers, the operation is implemented by convolution on $\mathbf{Z}$ followed by the Sigmoid activation function. We can get the attentional convolutional feature map $\Gamma_n \in \mathcal{R}(\mathbf{Z})$ as follows:

$$\Gamma_n = \mathcal{O}_{Reshape}(\mathbf{M}_n) \otimes \mathbf{Z}, \tag{10}$$

where $\mathcal{O}_{Reshape}$ reshapes the size of the input to be the same as $\mathbf{Z}$, and $\otimes$ denotes the element-wise product.

The generalized N attention maps inevitably involve redundancy such as background noise, we exploit the AT operation to filter these maps. Firstly, AT calculates a maximum value of each 2D mask map $(\mathbf{M}_n)$ from the N attention feature maps, and yields the maximum value vector $\mathbf{m}_v \in \mathbb{R}^{N\times 1}$. Then, the maximum of $\mathbf{m}_v$ denoted by AT_{max} can be calculated as

$$AT_{max} = \max_{1\leq n\leq N} \mathbf{m}_v(n) \tag{11}$$

AT_{max} is the global maximum value of these N feature maps. We use this value and set the thresholding bound as $T_B = \alpha \times AT_{max}$, where α is an adaptive coefficient ($0 \leq \alpha \leq 1$). Thus we can get the information-rich feature map $\mathcal{T}(\Gamma_n)$ by AT operator i.e., if the nth value in $\mathbf{m}_v$ is less than T_B, the corresponding map Γ_n will be set to zero. Instead of adopting the widely-used global average pooling, we adopt the global maximum pooling(GMP) for all feature maps generated by ARD and then concatenate them. The global maximum pooling can preserve the most salient features (such as image edge and image texture) than global average pooling.

Following the GMP, a fully connected operator $\mathcal{L}$ is leveraged to get the final image representation which is matched with the dimension of class classifiers, then we can get the extracted feature $f_v(x)$ of image x.

3.4. Loss function of the MZSL-GCN network

As described above, the architecture of MZSL-GCN is similar to the conventional fully supervised classification model, with a $(s+u)$-way classifier for both the seen and unseen classes. Unfortunately, only the data for seen classes are labeled and the data from unseen classes are unlabeled. Given that and inspired by Song et al. (2018), we propose a quasi-fully supervised learning loss as:

$$L = \frac{1}{n_s}\sum_{i=1}^{n_s} L_p(\mathbf{W}f_v(\mathbf{x}_i^s), y_i) + \frac{\lambda}{n_t}\sum_{i=1}^{n_t} L_b(\mathbf{W}f_v(\mathbf{x}_i^t)) \tag{12}$$

We assume that the ground truth label of an image is $\mathbf{y} \in \mathbb{R}^q$, where $y_i^c \in \{0, 1\}$ denotes whether label c is annotated to the sample x_i or not. We use traditional multi-label classification loss as follows on training data.

$$L_p(\mathbf{x}_i^s, y_i) = \sum_{c\in\mathcal{S}} y_i^c \log(\sigma(\tilde{y}_i^c)) + (1-y_i^c)\log(1-\sigma(\tilde{y}_i^c)) \tag{13}$$

where $\sigma(\cdot)$ is the Sigmoid function and $\sigma(\tilde{y}_i^c)$ is the predicted probability of $\mathbf{x}_i^t$ with respect to class c.

Different from the conventional definition, where the loss is the classification loss L_p only, we define a balance term L_b to reduce the bias to seen classes on unlabeled data:

$$L_b(\mathbf{x}_i^t) = -\ln \frac{\sum_{c\in\mathcal{S}}\sigma(\tilde{y}_i^c)}{\sum_{c\in\mathcal{S}}\sigma(\tilde{y}_i^c)+\sum_{c\in\mathcal{U}}\sigma(\tilde{y}_i^c)} \tag{14}$$

The bias term encourages the model to increase the sum of probabilities of being any unseen class. Specifically, the term ensures that the ratio between the sum of the predicted probabilities for unseen label set and that for all labels will be not too small. Consequently, it prevents testing instances from being classified as seen labels only, and thus alleviates the bias toward seen ones. λ is a trade-off weight between two different terms, and we compute its value by cross-validation. By minimizing the above loss, we can optimize the GCN subnet for multiple interdependent classifiers ($\mathbf{W}$) and the image representation subnet ($f_v(\mathbf{x})$) in an end-to-end coherent fashion within a unified network.

4. Experiments

4.1. Experimental setup

We conducted experiments on two benchmark multi-label image classification datasets (MS-COCO and NUS-WIDE[1]) to test our model on (generalized) multi-label zero-shot classification. We provide visual results to further analyze the advantages of our model. The statistics of the two datasets are summarized in Table 1. We use the cleaned version of NUS-WIDE, which contains 130,203 images, with 100,000, 10,203 and 20,000 for training, validation and testing, respectively. More detailed information of these two datasets can be found in Chen, Wei, Wang, and Guo (2019) and Lee et al. (2018). Following Lee et al. (2018), for the methods which predict labels according to the ranking scores of the labels, we assigned the K highest-ranked labels to the image, and compared the assigned labels to the ground truth. The commonly-used metrics of precision (P), recall (R), and F1-measure are considered.

Table 1
Statistics of datasets.

Dataset	#training	#validation	#testing	#classes
NUS-WIDE	100,000	10,203	20,000	1,000/81
MS-COCO	78,081	4,000	40,137	80

We mixed labeled and unlabeled data for training. Each batch of training images (default size is 32) are randomly selected from the mixed collection. Our GCN-based classifier learning subnet consists of two GCN layers with an output dimensionality of 1024 and 2048, respectively. For the label semantic embedding **A**, we use a 300-D GloVe trained on the Wikipedia dataset as the label representation. For the label whose name contains multiple words (such as "baseball bat"), we obtain the label representation as the average of the embedding of all words. For the label correlation matrix, we empirically set τ in Eq. (5) to 0.4. We adopt ResNet-101 as the backbone framework, which was pre-trained on ImageNet in our experiments. All the input images are randomly cropped and resized into 448 × 448 with random horizontal flips for data augmentation during training follows Chen et al. (2019). So we can obtain 2048 × 14 × 14 feature maps from the "conv5-x" layer of ResNet-101. We use SGD as the optimizer and implement the network by PyTorch. The momentum is fixed to 0.9 and the weight decay is fixed to 10^{-5}. The initial learning rate is 0.1, which decays by a factor of 2 at every 50 epochs, and the network is trained for 200 epochs. As to the attention region discovery module, the number of regions N (10 in our experiment) is empirically selected from $\{N \in \mathbb{N}_+ | 4 \leq N \leq 12\}$ and the AT parameter α is selected from $\{0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$, as suggested in Xie, et al. (2019). These hyper-parameters are selected via cross-validation. After getting the predicted label likelihoods from interdependent GCN classifiers, we adopt 0.5 as the threshold per label to get the binary label vector for each sample. This threshold value is also applied to other compared methods. The demo code of MZSL-GCN is shared at http://mlda.swu.edu.cn/codes.php?name=MZSL-GCN.

4.2. Multi-label classification

We first test our model on the conventional multi-label classification task on two datasets NUSIWDIE-81 and MS-COCO. To reach a comprehensive comparison, we compare the performance of ML-GCN against representative and related multi-label classification methods:

- **WSABIE** (Weston et al., 2011) and **Fast0Tag** (Zhang et al., 2016) predict labels according to the ranking scores of the labels, we choose the top K labels. Following conventional settings, we report results for $K = 3$.
- **Logistics** (Wright, 1995) is a conventional baseline method based on Logistics regression. We adopt the regularization coefficient C as 2.0, penalty parameter as $L2$ and optimization algorithm solver as 'saga' for experiments. Logistics is implemented by the scikit-learn package.

[1] http://lms.comp.nus.edu.sg/research/ NUSWIDE.htm.

Table 2
Multi-label classification results on NUS-WIDE (81 labels) and MS-COCO (80 labels). Results for WSABIE, ML-KNN, and Fast0Tag are with respect to the top $K = 3$ relevant labels for each image.

	NUS-81			MS-COCO		
	P	R	F1	P	R	F1
WSABIE	27.0	46.1	34.1	54.0	55.7	54.8
Logistics	35.2	43.3	38.8	67.3	60.1	63.5
ML-KNN	25.0	43.6	31.8	54.9	56.7	55.8
Fast0Tag	28.9	**51.2**	36.9	57.2	61.3	59.2
MZSL-KG	41.2	45.3	43.2	72.8	63.4	67.8
ML-GCN	53.1	37.3	43.8	82.4	69.7	75.5
MZSL-GCN	**53.3**	38.4	**44.6**	**83.7**	**71.3**	**77.0**

Table 3
Results for the conventional/generalized MZSL tasks on NUS-1000 with 81 unseen labels and 925 seen labels.

	Conventional			Generalized		
	P	R	F1	P	R	F1
ESZSL	17.9	34.2	23.5	10.5	13.0	11.6
LFRLS	19.2	33.5	24.4	16.5	21.2	18.6
Fast0Tag	21.0	**35.2**	26.3	18.7	**24.2**	21.1
MZSL-KG	**26.9**	30.1	28.4	**20.4**	23.4	21.8
MZSL-GCN	25.7	32.3	**28.6**	20.3	23.9	**22.0**

- **ML-KNN** (Zhang & Zhou, 2007) is a multi-label lazy learning approach derived from traditional K-nearest neighbor (KNN) algorithm with the maximum a posterior principle. The number of nearest neighbors is set to default 10.
- **MZSL-KG** (Lee et al., 2018) is a recent MZSL method that uses knowledge graph to update the belief vector. It can also produce a satisfactory performance on the standard task of multi-label classification. We fix the propagation step T as 5.
- **ML-GCN** (Chen et al., 2019) adopts the GCN to mine label correlations and to induce interdependent classifiers for multi-label classification. Its parameter settings are the same as our MZSL-GCN.

For all the compared algorithms, codes and/or suggested parameters in the original papers are used here. We use ResNet-101 as the visual backbone network for all compared methods. Here, MZSL-GCN uses the traditional multi-label classification loss only and sets the initial learning rate as 0.1, which decays by a factor of 10 at every 30 epochs. The network is trained for 50 epochs.

Table 2 gives the results on the NUS-81 and MS-COCO datasets. Our approach, MZSL-GCN, achieves a superior or comparable performance against the baselines. Fast0Tag is introduced for the MZSL task, but can also be used in the conventional multi-label setting. MZSL-GCN clearly achieves improved results on both datasets. Both ML-GCN (Chen et al., 2019) and MZSL-GCN adopt GCN to learn label correlations and inter-dependent classifiers, but ML-GCN ignores the important local regions of the input images, which help to bridge the semantic mapping gap between visual features and inter-dependent classifiers. In contrast, MZSL-GCN considers both the local and global visual features of the images and achieves a better semantic mapping, thus leading to a better performance. MZSL-KG and MZSL-GCN use different techniques to explore label correlations, and MZSL-KG often loses to MZSL-GCN. This is because MZSL-KG ignores the local visual features and heavily depends on the constructed knowledge graph to establish the visual-semantic embedding, whereas MZSL-GCN leverages both local and global features and learns compatible inter-dependent classifiers with respect to these features. Since MZSL-GCN is explicitly designed for MZSL, its advantages in conventional multi-label learning tasks largely benefit the MZSL task.

4.3. MZSL and generalized MZSL

To comprehensively and comparatively study the performance on multi-label zero-shot learning tasks, we consider four related and competitive methods: ESZSL (Romera-Paredes & Torr, 2015), LFRLS (Shao et al., 2018), Fast0Tag, and MZSL-KG (all were introduced in Section 2). Besides the conventional MZSL task, we also consider the challenging task of generalized MZSL task, for which models are trained on seen labels but are required to predict both seen and unseen labels during testing. We use the same data splits for all the compared methods. The configurations of these compared methods are as follows:

- **ESZSL** (Romera-Paredes & Torr, 2015) models the relationships between features, attributes, and classes as a two linear layers network It has 3 super-parameters. We set $\gamma = 0.1$, $\lambda = 0.001$, and $\beta = \gamma\lambda$.
- **LFRLS** (Shao et al., 2018) learns a shared latent space by label factorization and uses the label semantics as the decoding function. It has two types of losses and the regularized least square loss is used in our experiment. We set $\alpha = 0.1$, $\beta_1 = 0.01$, and $\beta_2 = 0.01$.
- **Fast0Tag** (Zhang et al., 2016) predicts labels according to the ranking scores of the tags, a small K in generalized MZSL will result in low recall due to a large number of tags predicted for each image, so we report the conventional MZSL results with $K = 3$ and the generalized MZSL results with $K = 10$.
- **MZSL-KG** (Lee et al., 2018) uses knowledge graph to update the belief vector. For a fair comparison, we use the similar supplementary information (label co-occurrence and GloVe semantics) as ours to construct the knowledge graph for MZSL-KG, and fix the propagation step $T = 5$.

We report the results under conventional/generalized MZSL on the NUS-WIDE dataset. Following the work in Lee et al. (2018), we use a set of 81 labels from NUS-WIDE as the unseen label set $\mathcal{U}$; the seen label set $\mathcal{S}$ is derived from NUS-1000 with 75 duplicated labels removed and 925 seen ones. From the reported results in Table 3, we have the following observations:

(i) MZSL-GCN keeps a good balance of Precision and Recall, and thus has a higher F1 value in both conventional and generalized settings than compared methods. Fast0Tag has a better Recall, one possible reason is that it does not consider the different distributions of the seen and unseen classes (the unseen classes are more sparse), and Fast0Tag is more focused on seen classes, which leads to a low Precision but a high Recall. On the other side, MZSL-KG has a better Precision, since it uses label propagation to guide the knowledge transfer from seen classes to unseen ones, but this strategy may not find some “difficult” positive samples. As a result, it has a lower Recall compared with other methods. F1 measure is a balance between Precision and Recall, and MZSL-GCN achieves the highest F1 value, which confirms the effectiveness of our proposed ML-GCN.

(ii) MZSL-GCN-base yields comparable performance with existing MZSL methods except MZSL-KG. This fact indicates that even if there is no sample corresponding to the unseen class, the interdependent classifiers can be learned synchronously based on the correlations between the classes, which alleviates the difference in the distribution of the seen and unseen classes. It also proves the rationality to adopt GCN for training interdependent classifiers.

(iii) MZSL-GCN performs much better than MZSL-GCN-base. This contrast confirms our introduced balanced loss can significantly improve the performance. MZSL-GCN can effectively leverage

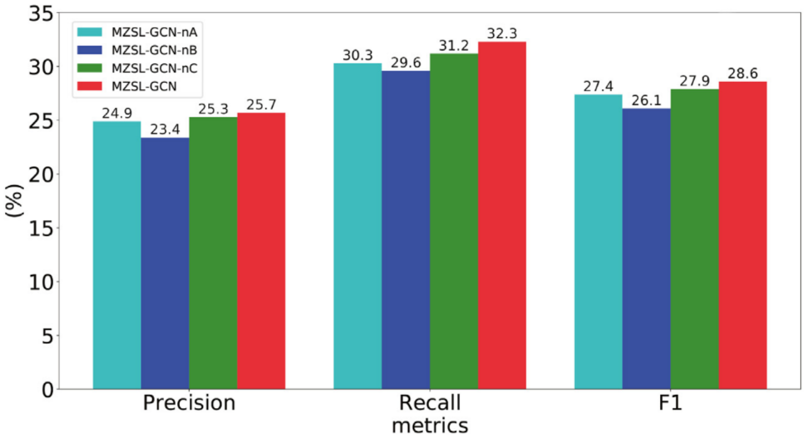
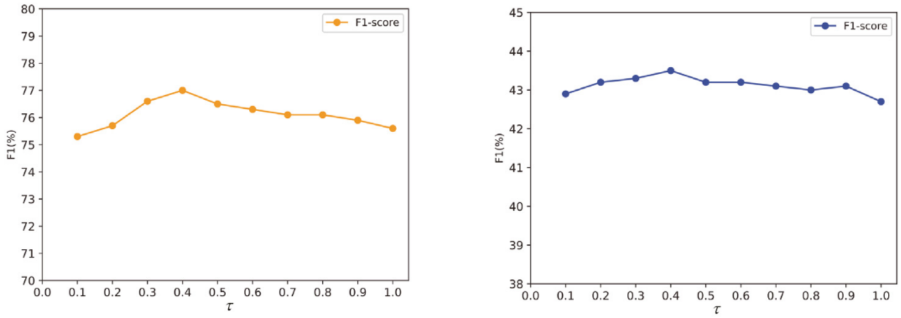


Fig. 2. Comparisons between different baselines of MZSL-GCN.



(a) F1 value vs. τ on NUSWIDE-81

(b) F1 value vs. τ on MS-COCO

Fig. 3. Performance vs. τ (threshold for the label correlations).

the valuable information embedded in the unlabeled data to facilitate the knowledge transfer from seen classes to unseen ones. The deep MZSL methods also perform better than non-deep ones (ESZSL and LFRLS), since they can capture nonlinear feature correlations and more complex visual-semantic mappings.

Overall, the results in Tables 2 and 3 clearly show that our proposed MZSL-GCN can extract the local and global features of image data, also leverage the unlabeled data and balance loss to gain a competent multi-label classifier in different scenarios.

4.4. Further analysis

To evaluate the contribution of specific key components of our model, we perform an ablation study by removing such components from the model and measuring how the performance is affected. In particular, we remove the attention region discovery module by directly adding a GMP operator after the last convolutional layer as baseline 'MZSL-GCN-nA'. We also train our proposed model with only labeled source data, i.e., the inductive version of our model. In this case, our loss (in Eq. (14)) degrades to traditional multi-label classification loss. We denote this baseline by 'MZSL-GCN-nB'. Finally, we remove the label correlation by replacing the correlation matrix **S** with an identity matrix as baseline 'MZSL-GCN-nC'. We observe in Fig. 2 that the F1-score drops 1.2% without the attentive module, reduces 2.5% without the balance constraint and 0.7% without the label correlation. This ablation study again shows that the full version of our model is preferable.

We vary the threshold value τ in Eq. (5) for converting the label correlation matrix into binary ones, and plot the F1 values under different input values of τ in Fig. 3. The F1 score manifests an increase pattern when filtered out the edges between two labels with low correlations (i.e., noisy edges). This pattern comes into a plateau when $\tau \approx 0.4$. As τ further increases, the number of filtering edges rises, but F1 score gradually reduces. That is because the moderately correlated labels are deemed as not related when a large τ is used. In our experiment, if we do not filter any edges, the model will not converge. Thus, there is no result for $\tau = 0$ in the figure. Based on this observation, we adopt $\tau = 0.4$ for experiments on MS-COCO and NUSWIDE. Interesting, the relatively best τ on two datasets are the same. One possible explanation is that we adopted the same word embedding-GloVe, and the threshold is more closely related to the embeddings of label nodes, instead of individual datasets.

We also vary the number of GCN layers for our model and summarize the results in Table 4. With three GCN layers, the output dimensionalities are 1024, 1024, and 2048. With four GCN layers, the output dimensionalities are 1024, 1024, 1024, and 2048. We observe that when the number of graph convolution layers increases, the performance drops. We hypothesis that the propagation between nodes is accumulated when using excessive layers, and these inter-dependent classifiers would be over-smoothed (see Fig. 5).

λ in Eq. (14) balances the loss with respect to unseen classes. We test λ with several different input values vary in $\{0, 0.1, \ldots, 0.9, 1\}$ and report the results in Fig. 4. The best result is obtained when $\lambda = 0.7$. A too small λ leaves the bias problem unsolved. On the other side, a too large λ yields negative effects on building the relationship between input features and semantic embedding. We can also observe that $\lambda = 0$ gives nearly the lowest result. This fact suggests that with an appropriate value

Table 4
Comparison of different GCN depths for the multi-label classification task.

Layer	MS-COCO			NUS-81		
	P	R	F1	P	R	F1
2-layer	83.7	71.3	77.0	53.3	36.4	43.3
3-layer	83.0	71.0	76.6	53.0	35.9	42.8
4-layer	81.4	70.1	75.3	52.3	34.5	41.6

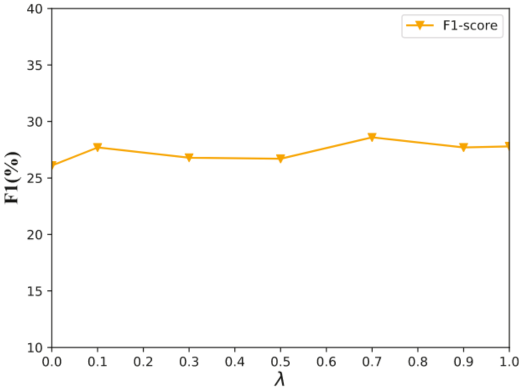


Fig. 4. Performance of MZSL-GCN with varying λ in (conventional setting).

Table 5
Runtimes (in hours) of compared methods under zero-shot learning task. 10.0 is the deep feature pre-training time for respective shallow methods.

	ESZSL	LFRLS	Fast0Tag	MZSL-KG	MZSL-GCN
NUS-1006	2.1+10.0	3.9+10.0	72.1	30.7	70.9

Table 6
Runtimes (in hours) of compared methods on multi-label classification task.

	WSABIE	ML-KNN	Fast0Tag	ML-GCN	MZSL-GCN
NUS-81	6.3+10.0	8.4+10.0	24.1	22.3	23.6
MS-COCO	4.9+7.7	6.1+7.7	21.5	17.1	18.9

of λ, our balanced loss in Eq. (12) can reduce the bias to seen labels. Given that we adopt $\lambda = 0.7$ for experiments.

The runtimes of our model and of compared methods are shown in Tables 5 and 6. For shallow methods (such as ESZSL and LFRLS), we need to use the pre-trained ResNet101 model to get the visual representation (about 10 h). For deep methods, the extraction of image visual features and model training are integrated into an end-to-end process. We can see that the runtime of shallow methods is typically smaller than the deep ones, but they have a lower classification performance. The runtime of our MZSL-GCN is closed to that of Fast0Tag. Since the propagation iterations of MZSL-KG are small ($T = 5$), its runtime is the lowest compared with other deep methods. Overall, our model has an acceptable runtime, while maintains the most prominent performance. All the experiments are performed on a server with following configurations: CentOS 7.5, 1 TB RAM, Inter Xeon 6148 and NVIDIA V100 GPU.

4.5. Classifier visualization

The effectiveness of our model has been quantitatively evaluated through comparison with existing methods under different settings and detailed ablation studies. In this section, we investigate whether a meaningful semantic topology is captured by visualizing the inter-dependent classifiers of MZSL-GCN. t-SNE is a variation of Stochastic Neighbor Embedding (Hinton & Roweis, 2003) that is much easier to optimize, and produces significantly better visualizations by reducing the tendency to crowd points together in the center of the map. First, we get the final outputs from the GCN sub-net as the classifiers for labels in NUS-81. Then we use t-SNE (Maaten & Hinton, 2008) to visualize the classifiers in a 2D plot (Fig. 5). The visualization method is implemented by t-SNE in the sklearn package.

We can see that the classifiers indeed maintain a meaningful semantic topology. For example, classifiers {'dog', 'cat', and 'tiger'} within the super concept 'animals' or classifiers {'surf', 'lake', 'waterfall' and 'ocean'} within the super concept 'water' tend to be closer to each other. This like cluster pattern also holds for other classifiers within a super concept (e.g., plant and buildings). This visualization result further shows the effectiveness of the proposed model on learning inter-dependent classifiers.

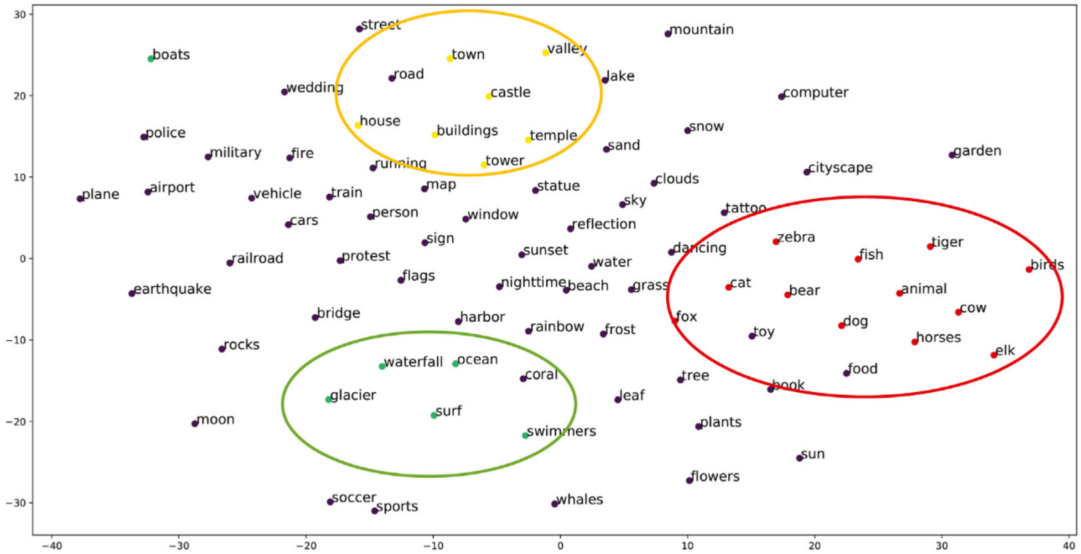


Fig. 5. 2D plot of inter-dependent classifiers of MZSL-GCN on NUS-WIDE 81.

5. Conclusion

In this paper, we studied multi-label zero-shot learning and proposed a novel framework (MZSL-GCN) to learn inter-dependent classifiers using GCN and extract compatible local and global visual features via an attention mechanism. The introduced attention mechanism enables better knowledge transfer from seen classes to unseen ones, and the proposed bias loss term can reduce the bias to seen classes. Empirical study shows that MZSL-GCN outperforms state-of-the-art methods in different MZSL tasks and in traditional multi-label classification.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by National Natural Science Foundation of China (62031003, 61872300 and 62072380).

References

Akata, Z., Perronnin, F., Harchaoui, Z., & Schmid, C. (2013). Label-embedding for attribute-based classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 819–826).

Chen, G., Song, Y., Wang, F., & Zhang, C. (2008). Semi-supervised multi-label learning by solving a sylvester equation. In *Proceedings of the SIAM international conference on data mining* (pp. 410–419).

Chen, Z.-M., Wei, X.-S., Wang, P., & Guo, Y. (2019). Multi-label image recognition with graph convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5177–5186).

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al. (2013). Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems* (pp. 2121–2129).

Fu, Y., Yang, Y., Hospedales, T., Xiang, T., & Gong, S. (2014). Transductive multi-label zero-shot learning. In *British machine vision conference* (pp. 1–12.

Gaure, A., Gupta, A., Verma, V. K., & Rai, P. (2017). A probabilistic framework for zero-shot multi-label learning. In *International conference on uncertainty in artificial intelligence* (pp. 1–10).

Gibaja, E., & Ventura, S. (2015). A tutorial on multilabel learning. *ACM Computing Surveys*, *47*(3), 52.

Hinton, G. E., & Roweis, S. T. (2003). Stochastic neighbor embedding. In *Advances in neural information processing systems* (pp. 857–864).

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

Lampert, C. H., Nickisch, H., & Harmeling, S. (2009). Learning to detect unseen object classes by between-class attribute transfer In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 951–958).

Lampert, C. H., Nickisch, H., & Harmeling, S. (2013). Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(3), 453–465.

Lee, C.-W., Fang, W., Yeh, C.-K., & Wang, Y.-C. F. (2018). Multi-label zero-shot learning with structured knowledge graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1576–1585).

Li, F.-F., Fergus, R., & Perona, P. (2006). One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(4), 594–611.

Li, Y., Zhang, J., Zhang, J., & Huang, K. (2018). Discriminative learning of latent features for zero-shot recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7463–7471).

Liu, X., Li, Z., Wang, J., Yu, G., Domeniconi, C., & Zhang, X. (2019). Cross-modal Zero-shot Hashing. In *IEEE international conference on data mining* (pp. 449–458).

Maaten, L. v. d., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(11), 2579–2605.

Mensink, T., Gavves, E., & Snoek, C. G. (2014). Costa: Co-occurrence statistics for zero-shot classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2441–2448).

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345–1359.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 1532–1543).

Read, J., Pfahringer, B., Holmes, G., & Frank, E. (2011). Classifier chains for multi-label classification. *Machine Learning*, *85*(3), 333.

Ren, Z., Jin, H., Lin, Z., Fang, C., & Yuille, A. (2015). Multi-instance visual-semantic embedding. arXiv preprint arXiv:1512.06963.

Romera-Paredes, B., & Torr, P. (2015). An embarrassingly simple approach to zero-shot learning. In *International conference on machine learning* (pp. 2152–2161).

Shao, H., Guo, Y., Ding, G., & Han, J. (2018). Zero-shot multi-label learning via label factorisation. *IET Computer Vision*, *13*(2), 117–124.

Song, J., Shen, C., Yang, Y., Liu, Y., & Song, M. (2018). Transductive unbiased embedding for zero-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1024–1033).

Sun, Y.-Y., Zhang, Y., & Zhou, Z.-H. (2010). Multi-label learning with weak label. In *AAAI conference on artificial intelligence* (pp. 593–598).

Tan, Q., Yu, G., Domeniconi, C., Wang, J., & Zhang, Z. (2018). Incomplete multi-view weak-label learning. In *Proceedings of the international joint conference on artificial intelligence* (pp. 2703–2709).

Tan, Q., Yu, Y., Yu, G., & Wang, J. (2017). Semi-supervised multi-label classification using incomplete label information. *Neurocomputing*, *260*, 192–202.

Tong, B., Wang, C., Klinkigt, M., Kobayashi, Y., & Nonaka, Y. (2019). Hierarchical disentanglement of discriminative latent features for zero-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 11467–11476).

Tsoumakas, G., Katakis, I., & Vlahavas, I. (2010). Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, *23*(7), 1079–1089.

Weston, J., Bengio, S., & Usunier, N. (2011). WSABIE: scaling up to large vocabulary image annotation. In *International joint conference on artificial intelligence* (pp. 2764–2770).

Wright, R. E. (1995). *Logistic regression*. American Psychological Association.

Xian, Y., Lampert, C. H., Schiele, B., & Akata, Z. (2019). Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(9), 2251–2265.

Xie, G.-S., Liu, L., Jin, X., Zhu, F., Zhang, Z., Qin, J., Yao, Y., & Shao, L. (2019). Attentive region embedding network for zero-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 9384–9393).

Yu, G., Chen, X., Domeniconi, C., Wang, J., Li, Z., Zhang, Z., & Wu, X. (2018). Feature-induced partial multi-label learning. In *IEEE international conference on data mining* (pp. 1398–1403).

Yu, G., Zhu, H., & Domeniconi, C. (2015). Predicting protein functions using incomplete hierarchical labels. *BMC Bioinformatics*, *16*(1), 1–12.

Yu, G., Zhu, H., Domeniconi, C., & Liu, J. (2015). Predicting protein function via downward random walks on a gene ontology. *BMC Bioinformatics*, *16*(271), 1–14.

Zhang, Y., Gong, B., & Shah, M. (2016). Fast zero-shot image tagging. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5985–5994).

Zhang, M.-L., & Zhou, Z.-H. (2007). ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition*, *40*(7), 2038–2048.

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

Zhu, Y., Kwok, J. T., & Zhou, Z.-H. (2018). Multi-label learning with global and local label correlation. *IEEE Transactions on Knowledge and Data Engineering*, *30*(6), 1081–1094.