The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20)

Zero-Shot Ingredient Recognition by Multi-Relational Graph Convolutional Network

Jingjing Chen,[1] Liangming Pan,[2,3] Zhipeng Wei,[4]
Xiang Wang,[3] Chong-Wah Ngo,[5] Tat-Seng Chua[3]
[1]Shanghai Key Lab of Intelligent Information Processing, School of Computer Science, Fudan University;
[2]NUS Graduate School for Integrative Sciences and Engineering, National University of Singapore;
[3]School of Computing, National University of Singapore; [4]Jilin University; [5]City University of Hong Kong
chenjingjing@fudan.edu.cn; {e0272310, xiangwang}@u.nus.edu; weizp17@mails.jlu.edu.cn;
cscwngo@cityu.edu.hk; dcscts@nus.edu.sg

Abstract

Recognizing ingredients for a given dish image is at the core of automatic dietary assessment, attracting increasing attention from both industry and academia. Nevertheless, the task is challenging due to the difficulty of collecting and labeling sufficient training data. On one hand, there are hundred thousands of food ingredients in the world, ranging from the common to rare. Collecting training samples for all of the ingredient categories is difficult. On the other hand, as the ingredient appearances exhibit huge visual variance during the food preparation, it requires to collect the training samples under different cooking and cutting methods for robust recognition. Since obtaining sufficient fully annotated training data is not easy, a more practical way of scaling up the recognition is to develop models that are capable of recognizing unseen ingredients. Therefore, in this paper, we target the problem of ingredient recognition with zero training samples. More specifically, we introduce multi-relational GCN (graph convolutional network) that integrates ingredient hierarchy, attribute as well as co-occurrence for zero-shot ingredient recognition. Extensive experiments on both Chinese and Japanese food datasets are performed to demonstrate the superior performance of multi-relational GCN and shed light on zero-shot ingredients recognition.

Introduction

Automatically constructing a food diary that tracks the ingredients consumed can facilitate the estimation of nutrition facts, which is crucial to various health relevant applications. Existing works (Kitamura, Yamasaki, and Aizawa 2008; Meyers et al. 2015) estimate the nutrients from the food image uploaded by the user. A classifier is first trained to recognize the category of the food in the image. Then, the nutrients are estimated based on the food composition table for each food category. These methods are only feasible for restaurant food with relatively fixed ingredient composition (Chen and Ngo 2016). However, for home-made food, as they usually do not have standardized cooking methods, food presentation, and ingredients composition, a clear mapping from the food category to its composing ingredients often does not exist. This motivates the need to directly recognize ingredients from the food image. Moreover, training a model that is able to recognize all the dishes is not realistic, as there are endless kinds of food categories worldwide (Bolaños, Ferrà, and Radeva 2017). Recognizing ingredients is more feasible in terms of scale, as the number of ingredients is much less than the number of the food category.

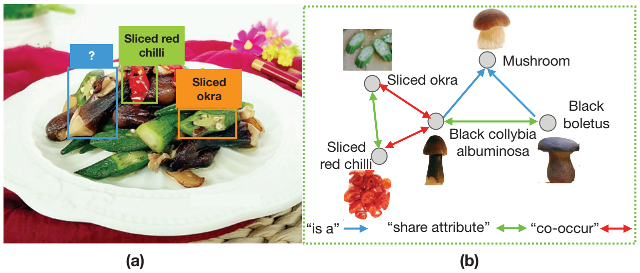


Figure 1: Given a food image (shown in (a)) that contains unseen ingredients, this work proposes to leverage multiple relations among ingredients (b) for unseen ingredients recognition.

Existing works on ingredient recognition mainly focus on recognizing a relatively narrow set of ingredients, ranging from 93 to 1,276 categories (Bolaños, Ferrà, and Radeva 2017; Chen and Ngo 2016; Chen, Ngo, and Chua 2017; Zhang, Lu, and Zhang 2016). The major obstacle for large-scale ingredient recognition is the lack of sufficient training samples. Collecting dish images that cover all ingredient categories is difficult, not mentioning the efforts devoted to label the ingredient composition for each image. Besides, for robust recognition, it requires to collect training samples under different cooking and cutting conditions, since the appearances of ingredients depend heavily on the way they are processed in cooking. To overcome this data sparsity issue, a more practical way is to endow the model with the ability to recognize ingredient which has zero training sample.

This paper studies the problem of zero-shot ingredient recognition. The key to dealing with zero-shot recognition problem is to transfer the knowledge obtained from familiar categories to unfamiliar ones. The knowledge transfer is either based on implicit knowledge representations (Frome et al. 2013; Norouzi et al. 2013), *i.e.* semantic embedding, or explicit knowledge bases which represents the knowledge as rules or relationships between objects (Misra, Gupta, and Hebert 2017). Recently, Wang *et al.* (2018) distill both implicit knowledge representations and explicit relationships (in the form of knowledge graph) for zero-shot recognition, which achieves state-of-the-art performance. Similar to Wang *et al.* (2018), we also utilize both implicit and explicit knowledge for zero-shot ingredient recognition.

However, compared with other zero-shot tasks, knowledge transfer for zero-shot ingredient recognition poses a unique challenge: relations among ingredients are complicated and cannot be modeled by a single-relational knowledge graph. The relations among ingredients can be characterised in multiple aspects. For example, some ingredients share similar visual appearance as they belong to the same hierarchy (*e.g.,* spinach and water spinach). Some ingredients are correlated as they share the same cutting or cooking methods (*e.g.,* diced tomato and diced red bell pepper). Other ingredients may be associated because they often cooccur with each other in a recipe (*e.g.,* corn and carrot). When building a knowledge graph to transfer knowledge from familiar ingredients to unseen ones, each type of relation actually carries diverse semantic information. Therefore, a single-relational knowledge graph cannot differentiate the different effects towards predicting unseen ingredients brought by different relations.

To address this, we define a multiple-relational graph to capture multiple types of relations among ingredients, such as *ingredient attribute* (*i.e.,* color or shape), *ingredient hierarchy*, and *ingredient co-occurrence* (as shown in Figure 1). We then propose a *multi-relational graph convolutional network*, termed mRGCN, to efficiently encode the multiple-relational graph so that the knowledge can be better transferred to unseen ingredients in zero-shot learning. Our major contributions can be summarized as follows:

- As the first to study the problem of zero-shot ingredient recognition, we construct a multi-relational knowledge graph that models three types of relations between ingredients that we find are crucial for this problem.
- We develop a new model namely mRGCN, which efficiently exploits three types of relations among ingredients, for zero-shot ingredient recognition.
- We explore several ways of coupling different types of relations in mRGCN, and verify the effectiveness of mRGCN for zero-shot ingredient recognition on two real-world datasets.

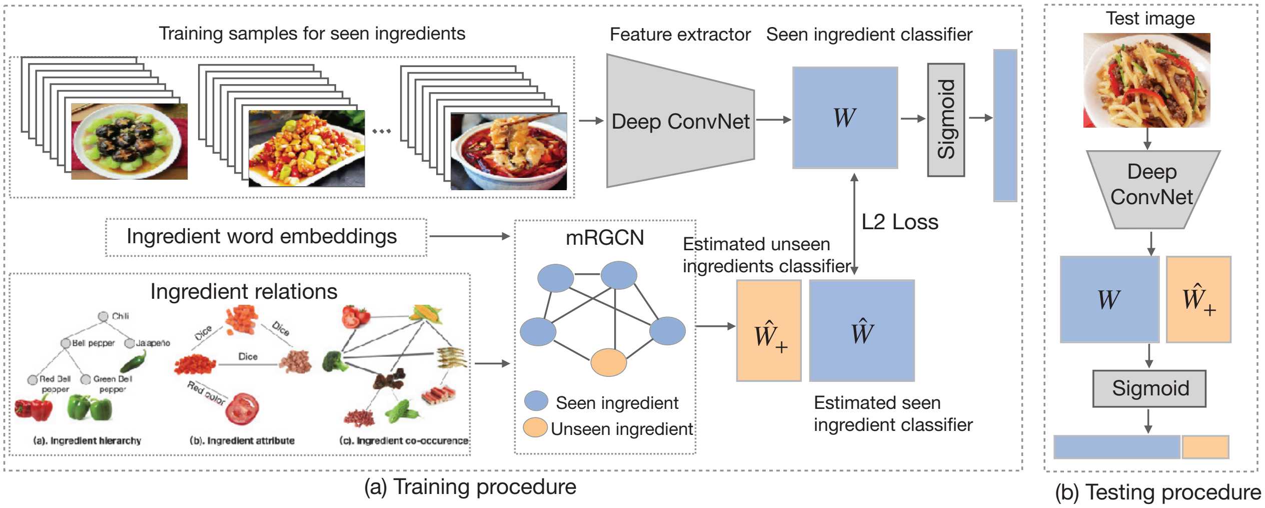
Related Work

Ingredient Recognition

Ingredient recognition receives much less attention than food categorization (Bolaños, Ferrà, and Radeva 2017; Chen and Ngo 2016; Chen, Ngo, and Chua 2017; Zhang, Lu, and Zhang 2016). Comparing with food categorization, ingredient recognition is much more challenging as ingredients are small in size and exhibit larger variances in appearance. Recent works on ingredient recognition are mostly based on deep models (Bolaños, Ferrà, and Radeva 2017; Chen and Ngo 2016; Chen, Ngo, and Chua 2017; Zhang, Lu, and Zhang 2016). For example, in (Chen and Ngo 2016), a VGG multi-task learning framework is proposed to simultaneously recognize food categories and ingredient labels, and then the ingredient labels are leveraged to retrieve the recipes belonging to unknown category. Similarly, (Zhang, Lu, and Zhang 2016) further incorporates cooking attribute recognition into multi-task learning. (Chen, Ngo, and Chua 2017) studies the interplay of ingredients, cutting and cooking attributes with a multi-task deep model. In (Min et al. 2017), multimodal deep Boltzmann machine is applied for ingredient recognition and food image retrieval. Segmentation of food into ingredients has also been explored (Meyers et al. 2015) by convolutional network and conditional random field (CRF). Nevertheless, existing efforts all devote to recognizing a pre-defined set of ingredients, zero-shot ingredient recognition has not been studied yet.

Multi-label Zero-shot Learning

Multi-label zero-shot learning (Mensink, Gavves, and Snoek 2014)(Lee et al. 2018) aims to predict the unseen labels that are not defined during the training process. Existing works on multi-label learning, such as (Chen et al. 2019) and (Marino, Salakhutdinov, and Gupta 2016), cannot be directly applied to multi-label zero-shot learning problems, as such methods assume all the labels are seen during the training and lack the ability to generalize to unseen class labels. Compared with single-label zero-shot learning, multi-label zero-shot learning receives less attention. Different with single-label zero-shot learning, which either relies on semantic attributes (Lampert, Nickisch, and Harmeling 2014)(Fu et al. 2012; Liu, Kuipers, and Savarese 2011) or semantic embeddings (Chen et al. 2018; Frome et al. 2013; Fu et al. 2015; Romera-Paredes and Torr 2015; Xian et al. 2016; Socher et al. 2013; Kodirov, Xiang, and Gong 2017; Zhang and Saligrama 2016), multi-label zero-shot learning usually relies on the relationships between seen labels and unseen labels (Mensink, Gavves, and Snoek 2014; Lee et al. 2018). One early model is COSTA (Mensink, Gavves, and Snoek 2014) which estimates the classifiers for new labels as the weighted combination of seen classes, leveraging their co-occurrence statistics. Another work is (Lee et al. 2018) which performs multi-label zero-shot learning on a structured knowledge graph with three types of relations among labels, namely “super-subordinate (ISA relation)” compiled from WordNet and “positive relation” as well as “negative relation” obtained from label similarities. Then Graph Search Neural Netwok (GSNN) (Marino, Salakhutdinov, and Gupta 2016) is applied to propagate the probabilities from seen labels to unseen labels. Our work differs from the aforementioned works in two aspects. First, apart from co-occurrence and ISA relation, the relation of attribute sharing is also considered in this work. Second, a novel mRGCN is proposed to model the interaction among different ingredient relations.



(a) Training procedure

(b) Testing procedure

Figure 2: Framework overview. During training, the proposed framework contains two major modules: a multi-label deep convolutional neural network for known ingredient recognition and a multi-relational graph convolutional neural network (mRGCN) for unseen ingredient classifier prediction. The learned ingredient classifiers extracted from multi-label CNN are used as ground-truth classifiers, supervising the learning of mRGCN. Thus, the knowledge learned from known ingredient will propagate through mRGCN to generate the classifiers for unseen ingredients. During testing, for a given image, the proposed framework will predict the unseen ingredient with the estimated unseen ingredient classifier.

Graph Convolutional Neural Network

Our work is also related to Graph Neural Networks (GNNs). In the literature, GNN is introduced to learn the representations for irregular grid data, such as graph and network data (Bruna et al. 2013; Defferrard, Bresson, and Vandergheynst 2016; Henaff, Bruna, and LeCun 2015; Kipf and Welling 2017). In (Kipf and Welling 2017), Graph Convolutional Network (GCN) is proposed to learn the node representations for semi-supervised entity classification. In (Velickovic et al. 2018), they propose graph attention network (GAT) to introduce the attention mechanism in GCN. Nevertheless, these models mostly focus on modeling single-relational graph. As there are different relation types among ingredients, we design a multi-relational graph convolutional network in this work for zero-shot ingredient recognition.

Recently, some GNNs are proposed to learn representations of multi-relational graphs. In (Kawamae 2019), to encode a multi-relational graph, separate representations for nodes and edges are learnt via jointly optimizing on two tasks: link structure prediction and node attributes preservation. In (Nie, Sun, and Yu 2019), local graphs are first sampled from multi-hop neighboring entities and relations of a given entity. Afterwards, localized graph convolutions are employed to generate node and relation embeddings. Mao *et al.*, (2019) proposed ImageGCN for multi-relational image modeling, which models the image-level relations to generate more informative image representations. However, different relation edges are equally-weighted during propagation in above methods, which may fail to capture various interactions between different relations. To address this, our model introduces the attention mechanism into GCN, such that various relations can contribute differently during graph propagation. Although a similar idea was proposed in (Shang et al. 2018), in which they jointly learn attention weights and node features in graph convolution, their model are specially designed for chemical datasets, which have different attributes and relations with the food domain.

Methodology

The goal of this work is to leverage multiple relations among ingredients for unseen ingredient recognition. Figure 2 presents an overview of the proposed framework, which is composed of two modules in training: a multi-label deep convolutional neural network (DCNN) for known ingredient classifier learning and a multi-relational graph convolutional network (mRGCN) for unseen ingredient prediction. For ingredient with training samples, the multi-label DCNN is conducted to learn the classifiers. The learned classifiers are further viewed as ground-truth classifiers, supervising the learning of mRGCN. mRGCN is performed on the graph which includes both known and unknown ingredients. The input to mRGCN are ingredient word embeddings as well as multiple ingredients relation graphs, and the output of mRGCN are the predicted classifiers of each ingredient. By minimizing the L_2 distance between the predicted classifiers and the ground-truth classifiers, the knowledge learned from known ingredients will propagate to unknown ingredients and enable the zero-shot ingredient recognition. Next, we will have a detailed introduction on the proposed framework.

Multi-Label CNN

Given a set of food images $\mathcal{X}$ where each image is labeled with its ingredient composition, and the ingredient set is denoted as $\mathcal{M}$. For $|\mathcal{M}|$ ingredients, we train a multi-label DCNN to learn the ingredient classifiers. The multi-label DCNN is built upon ResNet-50 (He et al. 2016) by replacing the softmax loss with sigmoid cross-entropy loss since ingredient recognition is a multi-label learning problem. The convolutional layers can be considered as the feature learning layers, while the last fully connected layer as the classifier layer. We denote the learned classifier weight as $\boldsymbol{W}, \boldsymbol{W} \in \mathbb{R}^{m \times d}$, where m is the number of ingredient categories and d is the dimension of the image features. For the i^{th} ingredient in $\mathcal{M}$, the learned classifier weight $\boldsymbol{W}_i \in \mathbb{R}^d$ can be considered as a binary classifier, predicting whether the image contains the i^{th} ingredient or not. Then the learned binary classifiers $\boldsymbol{W}$ will be used as ground-truth classifiers supervising the learning of multi-relational Graph Convolutioanl Network.

Multi-relational graph convolutional network

Given a food image x that contains unknown ingredient c (i.e., $c \notin \mathcal{M}$, $x \notin \mathcal{X}$), our goal is to leverage the relations of ingredient c with the known ingredients in $\mathcal{X}$ to generate the classifier $W_c \in \mathbb{R}^d$ of ingredient c. In this work, we leverage multi-relational Graph Convolutioanl Network to model various relations among ingredients and propagate the knowledge learned from known ingredients to unknown ingredients. Specifically, we consider three kinds of relations among ingredients, ingredient hierarchy, ingredient attribute as well as ingredient co-occurrence, as shown in Figure 2.

- **Ingredient Hierarchy.** This typically introduces the "is a" relations among ingredients, suggesting the knowledge between parents and children nodes. The ingredient hierarchy is manually constructed according to retail websites and recipe websites.
- **Ingredient Attribute.** We link the ingredients that share the same attributes, such as color, shape or cooking methods, exhibiting the attribute-aware knowledge among ingredients. In this work, we consider 19 attributes in total, including 5 Common colors ("white", "black", "red", "green" and "yellow"), 8 Shape attributes ("slice", "dice", "minced", "powder", "roll", "trunk", "shred", "julienne") and 6 cooking methods (i.e., "deep-fry", "dry", "fry", "steam", "boil", "pickle") that will have significant affects on the ingredient appearances.
- **Ingredient Co-occurrence.** The motivation is that certain groups of ingredients co-occur more often while some ingredients are likely exclusive of each other. Such kinds of co-occurrence relation might help refine the recognition results of unseen ingredients. Therefore, we also consider the co-occurrence relations among ingredients.

We introduce multi-relational graph neural network (mRGCN) to model these three kinds of relations for graph propagation. Denote $\mathcal{N}$ as the whole ingredient set, which includes both seen ingredients $\mathcal{M}$ and unseen ingredients $\mathcal{C}$, $\mathcal{N} = \mathcal{M} \cup \mathcal{C}$. Denote $\boldsymbol{A}^i, \boldsymbol{A}^i \in \mathbb{R}^{n \times n}$ as the adjacent matrix for i^{th} relation over $\mathcal{N}$, $i \in \{1, 2, 3\}$. For graph convolutional neural network which only considers one type of relation, the graph propagation is performed through the following equation:

$$\boldsymbol{H}^{(l+1)} = \sigma(\hat{\boldsymbol{A}}\boldsymbol{H}^{(l)}\boldsymbol{\Theta}^{(l)}), \tag{1}$$

where $\hat{\boldsymbol{A}} \in \mathbb{R}^{n \times n}$ is the normalized version of binary adjacent matrix $\boldsymbol{A}$, $\boldsymbol{H}^{(l)}$ represents the activations in the l-th layer, and $\boldsymbol{\Theta} \in \mathbb{R}^{C \times F}$ denotes the trainable parameters; $\sigma(\cdot)$ is the nonlinear activation function. When $l = 0$, $\boldsymbol{H}^{(0)} \in \mathbb{R}^{n \times k}$ is the input node features with the dimensionality of k. Similar to (Wang, Ye, and Gupta 2018), we use the word embeddings of ingredients as the input of GCN.

The most straightforward way to extend the single relational graph convolutional network to multi-relational graph convolutional network is to sum the graph convolution outputs from different relation adjacent matrices together, as:

$$\boldsymbol{H}^{(l+1)} = \sigma(\sum_{i=1}^{3} \hat{\boldsymbol{A}}^i \boldsymbol{H}^{(l)}\boldsymbol{\Theta}^{(l,i)}), \tag{2}$$

where $\boldsymbol{\Theta}^{(l,i)}$ is the weight matrix for i-th relation.

Another way to aggregate different relations is concatenating graph convolution outputs from different relation adjacent matrices, which is formulated as:

$$\boldsymbol{H}^{(l+1)} = \sigma(\text{Concat}_{i=1}^{3} \hat{\boldsymbol{A}}^i \boldsymbol{H}^{(l}\boldsymbol{\Theta}^{(l,i)}). \tag{3}$$

In this way, different kinds of relations will be included during the knowledge propagation process. As the GCN is trained to predict the classifier weights of ingredients, the number of output channel for the last convolution layer should be the same with the dimension of image features. Denote $\hat{\boldsymbol{W}} \in \mathbb{R}^{n \times d}$ as the predicted classifier weights for ingredient, which are the outputs of GCN. The loss function for training GCN is the Mean Square Error between the predicted classifier weights and ground-truth classifier weights of known ingredients. Thus, we have

$$L = \frac{1}{md} \sum_{i=1}^{m} \sum_{j=1}^{d} (\hat{\boldsymbol{W}_{i,j}} - \boldsymbol{W}_{i,j})^2, \tag{4}$$

where m is the number of known ingredients. By using the ground-truth classifier weights of known ingredients to supervise the training of GCN, the knowledge of known ingredients can be propagated to unknown ingredients.

Attentive Multi-relational graph convolutional network

Intuitively, various relations should contribute differently towards knowledge propagation for zero-shot learning. However, neither directly summing the graph convolution outputs from different relation graphs nor concatenating them can capture the difference of interactions among different relations. In order to deal with interactions among different relations, we introduce the attention mechanism to learn different weights for different relations during the training of multi-relational graph convolutional network. Similar to (Ma et al. 2018), we also introduce relation specific representation that considers both within relation interactions and across relation interactions during graph propagation. Denote $\boldsymbol{H}^{(l+1,i)}$ as the relation specific representation for i-th relation, we have:

$$\boldsymbol{H}^{(l+1,i)} = \sigma(\overbrace{\hat{\boldsymbol{A}}^i \boldsymbol{H}^{(l)} \boldsymbol{\Theta}^{(l,i)}}^{\text{within-relation interactions}} \underbrace{+ \sum_{j \neq i}^{3} \alpha_{i,j} \boldsymbol{H}^{(l)} \boldsymbol{\Theta}^{(l,j)}}_{\text{across-relation interactions}}), \tag{5}$$

where $\hat{\boldsymbol{A}}^i \boldsymbol{H}^{(l)} \boldsymbol{\Theta}^{(l,i)}$ presents the updated representation which aggregates the knowledge from the ingredients connected with i-th relation, reflecting the inner-relation interactions; meanwhile, as one relation might influence the message-passing or information propagation process of the other relations, we formulate such a cross-relation interaction as $\alpha_{i,j} \boldsymbol{H}^{(l)} \boldsymbol{\Theta}^{(l,j)}$. For i^{th} relation, the representation of across-relation interactions is the weighted sum of the projected dimension specific representation. Wherein, the attentive weight $\alpha_{i,j}$ models the importance of relation i to relation j, and $\sum_{i=1}^{3} \alpha_{i,j} = 1$. We formulate such attention mechanism through the following bilinear function:

$$\alpha_{i,j} = \frac{exp(tr(\boldsymbol{\Theta}_i^T \boldsymbol{M} \boldsymbol{\Theta}_j))}{\sum_{i=1}^{3} exp(tr(\boldsymbol{\Theta}_i^T \boldsymbol{M} \boldsymbol{\Theta}_j))} \tag{6}$$

where $tr(.)$ is the trace of a matrix and $\boldsymbol{M}$ is the parameters to be learned in the bilinear function. Softmax is applied to normalize the attention scores. The motivation of designing such kind of attention mechanism is based on the observation that if two relations are highly similar, the two projection matrices should also be highly related. Therefore, the attention weights should be learned based on these projection matrices. To this end, a bilinear function is used to model the relations between projection matrices and learn the attention weights.

Having established the relation specific representations, we concatenate them together as the final updated representations, as:

$$\boldsymbol{H}^{(l+1)} = \text{Concat}_{i=1}^{3} \boldsymbol{H}^{(l+1,i)}. \tag{7}$$

Implementation details

As illustrated in (Kampffmeyer et al. 2018), the aim of GCN is to exchange information between nodes in the graph in regression setting. However, stacking multiple GCN layers easily leads to information dilution and might hinder accurate regression. Therefore, distinct from (Wang, Ye, and Gupta 2018) which uses 6 GCN layers, our mRGCN is composed of 4 convolutional layers with output channel numbers as 2048-1024-512-D. Since we use ResNet for image feature learning and ground-truth classifier learning, D is 2048 in our settings. Similar to (Wang, Ye, and Gupta 2018), we apply LeakyReLU (Maas, Hannun, and Ng 2013) with the negative slope of 0.2 as the activation function and perform L_2 normalization on the output of mRGCN to regularize the output into similar magnitudes.

To obtain the word embeddings for GCN inputs, we use word2vec (Mikolov et al. 2013) model trained on the cooking instructions of Recipe 1M (Salvador et al. 2017), which leads to 300-d vectors. Similar to (Wang, Ye, and Gupta 2018), for the classes whose names contain multiple words, we match all the words in the trained model and find their embeddings. By averaging these word embeddings, we obtain the ingredient word embeddings.

Experiments

Dataset

The performances are evaluated on two datasets, a Chinese food dataset VIREO Food 172 (Chen and Ngo 2016) and a Japanese food dataset UEC Food-100 (Matsuda and Yanai 2012). **VIREO Food-172** covers 172 most common Chinese dishes, being labeled with 353 ingredients. In total, this dataset contains 110,241 food images. We split 60% as the training set, 30% as the test set and the remaining for validation. As the appearance of ingredients depends heavily on the applied cooking and cutting methods, Chen et al. (Chen and Ngo 2016) integrate the cooking and cutting methods into ingredient labels. Therefore, there are ingredient labels like "sliced boiled egg", "minced garlic". We link ingredients that share the same color, shape or cooking methods to generate the ingredient attribute sharing graph. For hierarchy, we first compile different ingredient hierarchies from recipe websites such as "Go cooking"[1] and online groceries shopping website such as "Redmart"[2], and manually merge them in order to form a complete ingredient hierarchy. Then we map the ingredients in VIREO Food-172 dataset to our ingredient hierarchy to obtain the "is a" relation of 353 ingredients. Including the inner nodes, there are 519 nodes in total. The co-occurrence relation among ingredient is built according to the co-occurrence statistics of the training data. If the co-occurrence value between two ingredients larger than a given threshold, then there will be an edge linking them. Note that the co-occurrence statistics can also be obtained from external recipe database. **UEC Food-100** contains 14,136 food images from 100 Japanese food categories. Chen et al. (Chen and Ngo 2016) labeled this dataset with 190 ingredient labels. We merge the duplicate ingredient labels and finally get 174 labels. Similar to VIREO Food-172 dataset, we build the ingredient attribute sharing relation, ingredient co-occurrence relation, as well as the ingredient hierarchy. Including the inner nodes of the ingredient hierarchy, we have 268 nodes in total. Table 1 shows the statistic of these two datasets.

Table 1: Dataset statistics of VIREO Food-172 and UEC Food-100.

Dataset	Train/Test	Edge type	Edge #
VIREO Food-172	283/70	"is a"	1,175
		"share attri."	1,092
		"co-occur."	1,150
UEC Food-100	139/37	"is a"	559
		"share attri."	424
		"co-occur."	507

[1] www.xiachufang.com

[2] www.redmart.com

Table 2: Effect of different relation graphs to zero-shot ingredient recognition.

Dataset	Model	Hit@k (%)				
		1	2	5	10	20
VIREO Food-172	RGCN_{H}	8.0	12.3	17.5	23.1	28.8
	RGCN_{A}	1.7	1.8	2.6	4.4	11.2
	RGCN_{C}	2.8	3.6	6.2	10.5	15.8
	$\text{mRGCN}_{\text{H\&A}}$	3.6	12.0	21.9	23.4	29.0
	$\text{mRGCN}_{\text{A\&C}}$	21.5	26.8	33.2	42.5	47.9
	$\text{mRGCN}_{\text{H\&C}}$	21.9	24.3	31.7	37.0	44.4
	mRGCN	**23.5**	**28.5**	**37.7**	**43.6**	**48.8**
UEC Food-100	RGCN_{H}	3.1	5.0	5.7	16.5	21.7
	RGCN_{A}	1.3	1.6	3.1	8.1	16.5
	RGCN_{C}	1.5	1.5	1.8	3.1	4.5
	$\text{mRGCN}_{\text{H\&A}}$	3.6	3.6	4.6	15.7	18.9
	$\text{mRGCN}_{\text{A\&C}}$	1.5	17.4	20.6	**22.6**	25.5
	$\text{mRGCN}_{\text{H\&C}}$	1.5	1.6	2.5	5.2	20.9
	mRGCN	**17.0**	**20.3**	**22.4**	22.4	**36.7**

Experimental settings

We randomly select 20% of the nodes in VIREO Food-172 and UEC Food-100 as unseen ingredients for evaluation. In total, we sample 70 and 37 ingredients as test unseen ingredient sets on VIREO Food-172 and UEC Food-100, respectively. During the training, we drop the images that contain any of the 20% selected ingredients. To obtain the ground-truth classifiers, we fine-tune ResNet-50 which is pre-trained with ImageNet on VIREO Food-172 and UEC Food-100 for multi-label ingredient recognition. For training the mRGCN, we adopt Adam (Kingma and Ba 2014) optimizer and the learning rate is set as 0.001. We train the mRGCN for 500 epochs for every experiment and report the performance of the model which attains the best result on validation set.

The evaluation metric we use is the Top-K hit ratio, which measures the percentage of hitting the ground-truth labels among the top-k positions of predictions. Following (Wang, Ye, and Gupta 2018), we include both training and the predicted classifiers during the testing. Different from previous work on zero-shot recognition, our zero-shot ingredient recognition is a multi-label recognition problem, and the testing image may contain both known and unknown ingredients. For fair evaluation, we skip the predictions of known ingredients which appear in ground-truth labels when calculating the top-k hit rate.

Experimental results

Effect of Different Relation Graphs As the modeling of different ingredient relations is at the core of mRGCN, we investigate its impact and get deep insights on the knowledge propagation among classifiers. We use RGCN_{H}, RGCN_{A}, and RGCN_{C} to indicate the model trained only with Hierarchy, Attribute, and Co-occurrence relations, respectively; moreover, $\text{mRGCN}_{\text{H\&A}}$ shows that the Hierarchy and Attribute relations are used together, and similar notations for others. We summarize the results in Table 2.

By analyzing the performance on Tables 2, we have the following observations: (1) Among models using an individual relation, RGCN_{H} consistently achieves the best performance, indicating that the "is a" relation is of more importance to transfer knowledge between classifiers than the others; (2) When the co-occurrence relation are simultaneously considered with another relations, the improvement can be achieved compared with the models with single relations. It is reasonable since the co-occurrence relation is capable of revealing the statistical patterns or association rules that are hard to exhibit in other relations. We hence attribute such improvements to the complementary relationships between the co-occurrence and hierarchy/attribute graphs; (3) It is interesting that $\text{mRGCN}_{\text{H\&A}}$ performs poor on VIREO dataset. This might be caused by the cross-relation interaction, which could assign high attentive weights since the hierarchy and attribute relations have much overlapped and homogeneous knowledge. We leave such exploration as our future work; And (4) mRGCN substantially outperforms the variants, verifying the rationality of multiple relation modeling.By exploiting the message-passing mechanism of GNNs, mRGCN is capable of fusing heterogeneous information together and propagating knowledge from the seen ingredients to the unseen.

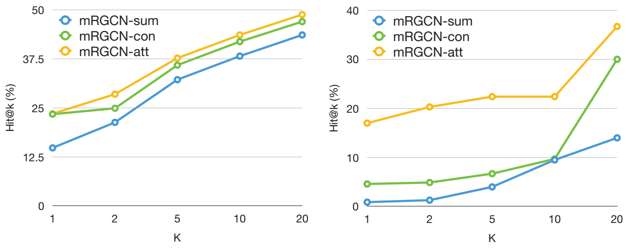
To get deep insights on how different relations influence the recognition, we visualize the top-5 predictions on unseen classes for three images, as Figure 3 illustrates. In particular, we show the results of mRGCN with all three relations and mRGCN_{H}. As we can see, mRGCN_{H} tends to generate higher predictions for the unseen ingredients under the same parents with seen ingredients. For example, for the first image, mRGCN_{H} predicts "shredded pickled bamboo shoot" in the most salient place, since such ingredient and the seen ingredient "hot pickled mustard" belong to the same subtree, thus being assigned with higher prediction. Moreover, it ranks "Fern root noodles" and "Macaroni" in top five positions, since they share the same parent node "noodles" with the seen ingredient "sweet potato starch noodles". As a result, the unseen ground-truth label "green vegetables" are not in top-5 predictions. On the other hand, as "green vegetables" usually co-occurs with noodles, such as "sweet potato starch noodles", our mRGCN which also utilizes co-occurrence relation is able to rank "green vegetables" at the salient place. Similar observations can be also found in the second and third images.

By analyzing the individual performance of each relation per category, we have the following findings. First, the co-occurrence relation has an impact on transferring knowledge among the ingredients which usually co-occur with other seen categories, such as "green vegetables" and "white sesame". Second, the attribute relation plays a pivotal role for the ingredients (*e.g.* "julienne ham") whose appearances are determined by their attributes. Moreover, the hierarchy relation is capable of effectively transferring knowledge for the ingredients near to the leaf nodes (*e.g.* "black chicken"), since their parent nodes provide more discriminative attributes for information propagation.

Effect of Attention Mechanism To explore the effect of attention mechanism, we consider the variants of mRGCN that uses different ways to fuse outputs of multiple relations — more specifically, the sum (*c.f.* Equation (2)), concatenation (*c.f.* Equation (3)), and attention (*c.f.* Equa-

Test Image	Ground truth	Hierarchy	All
	Groundnut kernels (train) Chili oil (train) Hot pickled mustard (train) Sweet potato starch-noodles (train) Green vegetables (test)	Shredded pickled-bamboo shoot Preserved vegetables Noodles Fern root noodles Macaroni	**Green vegetables** Crushed pepper Black sesame Dried pepper Sichuan peppercorns
	Scrambled egg (train) Loofah (test)	Crushed pepper Crushed egg crepe Bitter gourd slices Okra slices Search pepper	Crushed egg crepe Green vegetables Crushed pepper **Lofah** Giner slices
	Sweet and sour sauce (train) Fish slices (test) White sesame (test)	Seared pepper Batonnet tenderloin Ribbonfish Beef tripes Hot and dry pepper	Seared pepper **Fish slices** Sichuan peppercorns **White sesame** Ribbonfish

Figure 3: Qualitative results on Vireo Food-172. For each image, the top 5 zero-shot predictions on unseen classes of our mRGCN with all the three relation types and single relation GCN with ingredient hierarchy are visualized. Predictions are ordered in decreasing score, with correct predictions in bold.



(a) VIREO Food-172

(b) UEC Food-100

Figure 4: Comparison of the performance between mRGCN with Attentive mRGCN. The testing are done on both seen and unseen class.

tion (7)) operations, termed as mRGCN-sum, mRGCN-con, and mRGCN-att, respectively. The results on two datasets are summarized in Figure 4.

Clearly, mRGCN-con consistently achieves better performance than mRGCN-sum, suggesting that the concatenation operation integrates the characteristics of individual relation graphs in a better way. Furthermore, mRGCN-att outperforms the other variants by a large margin. In terms of Hit@10 and Hit@20, it improves around 2% over the strongest variants on VIREO Food-172; meanwhile, the improvements are more than 12% and 6% of Hit@10 and Hit@20 respectively, on UEC Food-100 dataset. The results verify that the across-relation interactions in the attention mechanism works well, hence enables better zero-shot recognition performance.

Performance Comparison with Baselines We further compare our mRGCN against different baseline methods, including ConSE (Norouzi et al. 2013), COSTA(Mensink, Gavves, and Snoek 2014), Fast0Tag(Zhang, Gong, and Shah 2016), and GCNZ (Wang, Ye, and Gupta 2018). **ConSE** (Norouzi et al. 2013) feeds the test images into ConvNet that is trained only on the training classes. With the output probabilities, it selects top T predictions and the word embeddings of these classes. It then generates a new word embedding by averaging the T embeddings weighted with the predicted probabilities. Such output embedding is then applied to perform nearest neighbor search in the word embeddings of testing classes. The top retrieved classes are selected as the final result. As ingredient recognition is a multi-label recognition problem, the top T predictions may contain training classes which are in the ground-truth label list. We therefore skip those predictions that are known ingredients and appear in ground-truth labels, when generate the word embeddings test classes. Similar to (Wang, Ye, and Gupta 2018), we consider different values of T for evaluations. **COSTA** exploits co-occurrences of visual concepts in images for transferring the knowledge from seen categories to unseen categories. In this work, the classifiers of unseen classes are estimated with the weighted combination of related classes, using the co-occurrences to define the weight. **Fast0Tag** proposes to solve zero-shot image tagging by estimating the principal direction for an image with both linear mappings and nonlinear deep neural networks. In the estimated principal direction, the word vectors of relevant tags for a given image rank ahead of the irrelevant tags. **GCNZ** (Wang, Ye, and Gupta 2018) is one of the state-of-the-art methods, which shows much better performance than ConSE on ImageNet dataset. GCNZ leverages graph convolutional network to model the class relations for unseen class prediction. As GCNZ uses WordNet as knowledge graph which contains "is a" relations solely, it is a special case of our mRGCN based on hierarchy relations (*e.g.* mRGCN_H in Section). There are also some other zero-shot recognition baselines, such as DeViSE (Frome et al. 2013), however, most of them are designed for single-label recognition problem, which are not suitable for our task.

We compare our methods with these baselines under two experimental settings: testing on unseen class and testing on both seen and unseen class. Table 3 lists the performance comparison. ConSE is denoted as ConSE(T) where T is searched in $\{1, 2, 5\}$. From the results, mRGCN yields the best performance when testing on both unseen classes and all classes in most cases. When testing on only unseen classes, our method improves more than 18% over ConSE(5) and GCNZ in terms of Hit@10; meanwhile, when testing on all the classes, it consistently outperforms all baseline methods w.r.t all evaluation metrics, and the improvement can be as high as 20% compared with GCNZ w.r.t Hit@10. It is worthwhile mentioning that GCNZ as well as our method performs much better than ConSE in most cases, suggesting that combining knowledge graph with word embeddings could lead to much better results than the methods with word embeddings only. Besides, our methods shows better performance compared with GCNZ and COSTA on both experimental settings, showing the advantages of introducing more relations types for zero-shot learning.

Analogously, on UEC Food-100 dataset, mRGCN also outperforms all baselines on both settings in most cases.

Table 3: Comparison of the performance of our attentive mRGCN with different baselines.

Dataset	Settging/Metric Model	Seen & Unseen					Unseen				
		Hit@k (%)					Hit@k (%)				
		1	2	5	10	20	1	2	5	10	20
VIREO Food-172	ConSE(1) (Norouzi et al. 2013)	0	2.4	6.1	11.9	14.3	14.4	16.8	20.5	26.3	28.7
	ConSE(2) (Norouzi et al. 2013)	0	1.6	5.7	9.6	13.5	14.5	16.0	20.1	24.0	27.9
	ConSE(5) (Norouzi et al. 2013)	0	0.5	2.3	4.8	8.9	15.3	15.7	17.5	20.0	24.1
	COSTA (Mensink, Gavves, and Snoek 2014)	0	0	0	1.3	2.7	13.9	15.7	19.4	23.7	30.3
	Fast0Tag (Zhang, Gong, and Shah 2016)	0	1.8	5.8	11.2	19.2	1.9	5.7	17.4	34.2	**67.7**
	GCNZ (Wang, Ye, and Gupta 2018)	8.0	12.3	17.5	23.1	28.8	11.0	17.8	24.3	30.3	37.3
	mRGCN (Our)	**23.5**	**28.5**	**37.7**	**43.6**	**48.8**	**24.2**	**30.8**	**43.1**	**47.4**	60.6
UEC Food-100	ConSE(1) (Norouzi et al. 2013)	0	2.8	2.9	3.0	8.1	11.5	14.2	14.4	14.5	19.6
	ConSE(2) (Norouzi et al. 2013)	0	0	0.9	3.7	5.8	11.2	11.2	12.1	14.9	17.1
	ConSE(5) (Norouzi et al. 2013)	0	0	0.4	2.3	7.3	11.5	11.5	11.9	13.9	18.9
	COSTA (Mensink, Gavves, and Snoek 2014)	1.3	2.6	2.6	15.9	16.0	1.5	3.3	11.1	20.2	38.5
	Fast0Tag (Zhang, Gong, and Shah 2016)	0	1.3	4.0	9.4	16.7	0	1.7	9.0	23.2	**43.7**
	GCNZ (Wang, Ye, and Gupta 2018)	3.1	5.0	5.7	16.5	21.7	3.1	5.0	12.3	19.4	26.0
	mRGCN (Our)	**17.0**	**20.3**	**22.4**	**22.4**	**36.7**	**17.9**	**21.7**	**22.5**	**24.3**	42.0

When testing only on unseen class, the accuracy of our attentive mGCN is almost 2 times as that of ConSE in Hit@2 and Hit@5. When testing on both seen and unseen class, our method improves more than 20% compared with ConSE and GCNZ in terms of Hit@2 and Hit@5. The results verify that our attentive mRGCN is effective in modeling different relation types hence lead to better zero-shot recognition performance.

Discussion Note that it has been recently demonstrated that BERT (Devlin et al. 2018) learns better word representation than traditional methods such as GloVe (Pennington, Socher, and Manning 2014) and Word2Vec. However, the superior performance of BERT comes from the ability of learning contextualized word embeddings that depend on the contexts where the word appears in. Hence, the embeddings for the same word will be different in different sentences. Nevertheless, in our problem, we need an unify word embedding for each word as the input of GCN. Therefore, we can only take a word's average BERT embeddings over its multiple appearances in the recipe contexts. By averaging contextualized word embeddings, we found this achieves much worse results than the static word embedding learned with Word2Vec.

We also investigate the effects of different backbone networks by comparing the performance of zero-shot recognition between VGG-16 (Simonyan and Zisserman 2014) and ResNet-50 (He et al. 2016) on Vireo Food-172 dataset. Table 4 summarizes the results when testing on unseen classes. Basically, compared to VGG-16, using ResNet-50 as the backbone network achieves better performance. This is mainly because that the features learned from ResNet-50 are better than the features learned from VGG-16.

Table 4: Effect of different backbone models to zero-shot ingredient recognition on Vireo Food-172 dataset.

Backbone model	Hit@k (%)				
	1	2	5	10	20
VGG-16	20.1	25.4	34.5	43.2	52.8
ResNet-50	24.2	30.8	43.1	47.4	60.6

Conclusion

We have presented the multi-relational graph convolutional network (mRGCN) that leverages ingredient hierarchy, ingredient co-occurrence as well as ingredient attribute for zero-shot ingredient recognition. Particularly, we studied different ways of coupling all these three relations in mRGCN, including summing, concatenating, and attention based integration methods. Experimental results on two datasets basically confirm the merit of using multiple relations for zero-shot ingredient recognition. Moreover, we have demonstrated attention mechanism is effective in modeling the interaction among different relations during the graph propagation process and enables better zero-shot recognition performance.

Acknowledgement

The work described in this paper was supported in part by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 11203517) and NExT++ project supported by National Research Foundation, Prime Minister's Office, Singapore under its IRC@SG Funding Initiative.

References

Bolaños, M.; Ferrà, A.; and Radeva, P. 2017. Food ingredients recognition through multi-label learning. In *Proceedings of ICIAP*.

Bruna, J.; Zaremba, W.; Szlam, A.; and LeCun, Y. 2013. Spectral networks and locally connected networks on graphs. *arXiv:1312.6203*.

Chen, J., and Ngo, C.-W. 2016. Deep-based ingredient recognition for cooking recipe retrieval. In *Proceedings of ACM MM*.

Chen, L.; Zhang, H.; Xiao, J.; Liu, W.; and Chang, S.-F. 2018. Zero-shot visual recognition using semantics-preserving adversarial embedding network. In *Proceedings of CVPR*, volume 2.

Chen, Z.-M.; Wei, X.-S.; Wang, P.; and Guo, Y. 2019. Multi-label image recognition with graph convolutional networks. In *Proceedings of CVPR*.

Chen, J.; Ngo, C.-W.; and Chua, T.-S. 2017. Cross-modal recipe retrieval with rich food attributes. In *Proceedings of ACM MM*.

Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of NIPS*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Frome, A.; Corrado, G. S.; Shlens, J.; Bengio, S.; Dean, J.; Mikolov, T.; et al. 2013. Devise: A deep visual-semantic embedding model. In *Proceedings of NIPS*.

Fu, Y.; Hospedales, T. M.; Xiang, T.; and Gong, S. 2012. Attribute learning for understanding unstructured social activity. In *Proceedings of ECCV*.

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

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of CVPR*.

Henaff, M.; Bruna, J.; and LeCun, Y. 2015. Deep convolutional networks on graph-structured data. *arXiv:1506.05163*.

Kampffmeyer, M.; Chen, Y.; Liang, X.; Wang, H.; Zhang, Y.; and Xing, E. P. 2018. Rethinking knowledge graph propagation for zero-shot learning. *arXiv:1805.11724*.

Kawamae, N. 2019. Marine: Multi-relational network embeddings with relational proximity and node attributes. In *The World Wide Web Conference*, 470–479. ACM.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv:1412.6980*.

Kipf, T. N., and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. *Proceedings of ICLR*.

Kitamura, K.; Yamasaki, T.; and Aizawa, K. 2008. Food log by analyzing food images. In *Proceedings of ACM MM*.

Kodirov, E.; Xiang, T.; and Gong, S. 2017. Semantic autoencoder for zero-shot learning. *arXiv:1704.08345*.

Lampert, C. H.; Nickisch, H.; and Harmeling, S. 2014. 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.; and Frank Wang, Y.-C. 2018. Multi-label zero-shot learning with structured knowledge graphs. In *Proceedings of CVPR*.

Liu, J.; Kuipers, B.; and Savarese, S. 2011. Recognizing human actions by attributes. In *Proceedings of CVPR*.

Ma, Y.; Wang, S.; Aggarwal, C. C.; Yin, D.; and Tang, J. 2018. Multi-dimensional graph convolutional networks. *arXiv:1808.06099*.

Maas, A. L.; Hannun, A. Y.; and Ng, A. Y. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of ICML*, volume 30, 3.

Mao, C.; Yao, L.; and Luo, Y. 2019. Imagegcn: Multi-relational image graph convolutional networks for disease identification with chest x-rays. *arXiv preprint arXiv:1904.00325*.

Marino, K.; Salakhutdinov, R.; and Gupta, A. 2016. The more you know: Using knowledge graphs for image classification. *arXiv preprint arXiv:1612.04844*.

Matsuda, Y., and Yanai, K. 2012. Multiple-food recognition considering co-occurrence employing manifold ranking. In *Proceedings ICPR*, 2017–2020.

Mensink, T.; Gavves, E.; and Snoek, C. G. 2014. Costa: Co-occurrence statistics for zero-shot classification. In *Proceedings of CVPR*.

Meyers, A.; Johnston, N.; Rathod, V.; Korattikara, A.; Gorban, A.; Silberman, N.; Guadarrama, S.; Papandreou, G.; Huang, J.; and Murphy, K. P. 2015. Im2calories: towards an automated mobile vision food diary. In *Proceedings of ICCV*.

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

Min, W.; Jiang, S.; Sang, J.; Wang, H.; Liu, X.; and Herranz, L. 2017. Being a supercook: Joint food attributes and multimodal content modeling for recipe retrieval and exploration. *IEEE Transactions on Multimedia* 19(5):1100–1113.

Misra, I.; Gupta, A.; and Hebert, M. 2017. From red wine to red tomato: Composition with context. In *Proceedings of CVPR*.

Nie, B.; Sun, S.; and Yu, D. 2019. An end-to-end structure aware graph convolutional network for modeling multi-relational data. In *Pacific Rim International Conference on Artificial Intelligence*, 296–308. Springer.

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

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *Proceedings of EMNLP*, 1532–1543.

Romera-Paredes, B., and Torr, P. 2015. An embarrassingly simple approach to zero-shot learning. In *Proceedings of ICML*.

Salvador, A.; Hynes, N.; Aytar, Y.; Marin, J.; Ofli, F.; Weber, I.; and Torralba, A. 2017. Learning cross-modal embeddings for cooking recipes and food images. In *Proceedings of CVPR*.

Shang, C.; Liu, Q.; Chen, K.-S.; Sun, J.; Lu, J.; Yi, J.; and Bi, J. 2018. Edge attention-based multi-relational graph convolutional networks. *arXiv preprint arXiv:1802.04944*.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Socher, R.; Ganjoo, M.; Manning, C. D.; and Ng, A. 2013. Zero-shot learning through cross-modal transfer. In *Proceedings of NIPS*.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2018. Graph attention networks. *Proceedings of ICLR* 1(2).

Wang, X.; Ye, Y.; and Gupta, A. 2018. Zero-shot recognition via semantic embeddings and knowledge graphs. In *Proceedings of CVPR*.

Xian, Y.; Akata, Z.; Sharma, G.; Nguyen, Q.; Hein, M.; and Schiele, B. 2016. Latent embeddings for zero-shot classification. In *Proceedings of CVPR*.

Zhang, Z., and Saligrama, V. 2016. Zero-shot learning via joint latent similarity embedding. In *Proceedings of CVPR*.

Zhang, Y.; Gong, B.; and Shah, M. 2016. Fast zero-shot image tagging. In *Proceedings of CVPR*, 5985–5994.

Zhang, X.-J.; Lu, Y.-F.; and Zhang, S.-H. 2016. Multi-task learning for food identification and analysis with deep convolutional neural networks. *Journal of Computer Science and Technology* 31(3):489–500.