Iterative Visual Relationship Detection via Commonsense Knowledge Graph

Hai Wan[1][0000−0001−5357−9130], Jialing Ou[1][0000−0001−9194−1735], Baoyi Wang[1][0000−0002−4276−8777], Jianfeng Du[2*][0000−0002−7541−1387], Jeff Z. Pan[3][0000−0002−9779−2088], and Juan Zeng[4*][0000−0002−2366−9327]

[1]School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China
wanhai@mail.sysu.edu.cn,
oujl5@mail2.sysu.edu.cn,wangby9@mail2.sysu.edu.cn
[2]School of Information Science and Technology/School of Cyber Security, Guangdong University of Foreign Studies, Guangzhou, Chinia
jfdu@gdufs.edu.cn
[3]Department of Computing Science, The University of Aberdeen, Aberdeen, UK
jeff.z.pan@abdn.ac.uk
[4]School of Geography and Planning, Sun Yat-sen University, Guangzhou, China
zengjuan@mail.sysu.edu.cn

Abstract. Visual relationship detection, *i.e.*, discovering the interaction between pairs of objects in an image, plays a significant role in image understanding. However, most of recent works only consider visual features, ignoring the implicit effect of common sense. Motivated by the iterative visual reasoning in image recognition, we propose a novel model to take the advantage of common sense in the form of the knowledge graph in visual relationship detection, named Iterative Visual Relationship Detection with Commonsense Knowledge Graph (IVRDC). Our model consists of two modules: a feature module that predicts predicates by visual features and semantic features with a bi-directional RNN; and a commonsense knowledge module that constructs a specific commonsense knowledge graph for predicate prediction. After iteratively combining prediction from both modules, IVRDC updates the memory and commonsense knowledge graph. The final predictions are made by taking the result of each iteration into account with an attention mechanism. Our experiments on the Visual Relationship Detection (VRD) dataset and the Visual Genome (VG) dataset demonstrate that our proposed model is competitive.

Keywords: Commonsense Knowledge Graph · Visual relationship detection · Visual Genome.

1 Introduction

Visual relationship detection, introduced by [12], aims to capture a wide variety of interactions between pairs of objects in an image. Visual relation can be

*Corresponding Author

represented as a set of relation triples in the form of $(subject, predicate, object)$, *e.g.*, $(person, ride, horse)$. Visual relationship detection can be used for many high-level image understanding tasks such as image caption [1] and visual QA [6].

Recently, visual relationship detection attracts more and more attention. Visual relationship detection methods can be categorized into two branches.

One branch includes those detection models that take into consideration not only the information from the dataset but also external knowledge. [18] proposed a teacher-student knowledge distillation framework making use of the internal and external knowledge for visual relationship detection. [10] constructed a directed semantic action graph and used deep variation-structured reinforcement to predict visual relationships.

Another branch includes those detection models that only consider prior images and their annotations. [12] not only proposed a typical model with language prior but also introduced the Visual Relationship Dataset (VRD) for visual relationship detection task. [19] applied a translation embedding model for visual relationship detection. [20] used a parallel FCN architecture and a position-role-sensitive score map to tackle the visual relationship detection. [5] exploited the statistical dependencies between objects and their relationships to detect visual relationships. [21] introduced deep structured learning for visual relationship detection. [9] proposed a deep neural network framework with the structural ranking loss to tackle the visual this task.

However, most of the recent works only consider the appearance or spatial features, while ignoring the implicit effect of common sense.

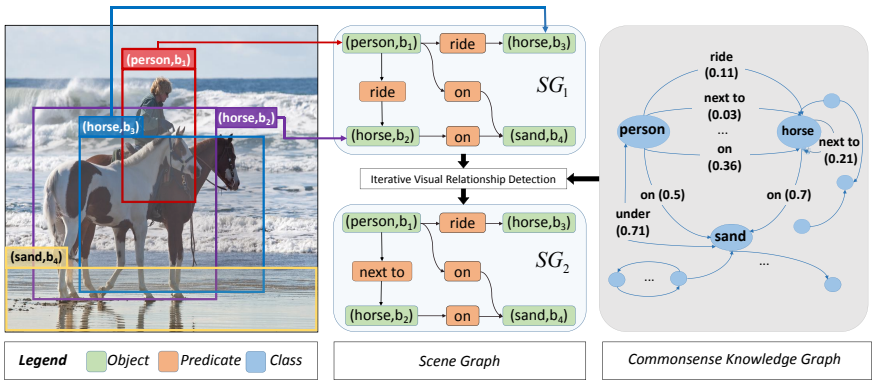

Fig. 1. An example image from VRD. The relation between $(person, b_1)$ and $(horse, b_2)$ is *ride*. Although $(horse, b_3)$ is similar with $(horse, b_2)$ in visual feature and positional feature, the relation between $(person, b_1)$ and $(horse, b_3)$ is *next to* but not *ride*.

Scene graph, introduced by [7], is a graph-based structural representation which describes the semantic contents of an image. Compared with scene graph, the well-known *knowledge graph* is represented as multi-relational data with enormous fact *triples* [2]. [17] further identified the *visual triples* of scene graph.

A scene graph is a set of visual triples in the form of $(head\ entity, relation, tail\ entity)$ in which an *entity* is composed of its *entity type* with *attributes* and grounded with a bounding box in its corresponding image, and a *relation* is the edge from the *head entity* to the *tail entity*. An example in VRD is shown in Figure 1. The relation between $(person, b_1)$ and $(horse, b_2)$ is $ride$. This visual triple is in the form of $((person, b_1), ride, (horse, b_2))$. The visual triple is shown in SG_1.

However, Figure 1 also shows that, if only considering the appearance or spatial feature between $(person, b_1)$ and $(horse, b_3)$, it is more likely that the relationship between these two objects are incorrectly detected as $ride$, as shown in SG_1. To avoid that, we introduce the notion of the *commonsense knowledge graph (CKG)*, in which each triple is labeled with its conditional probability. For example, the conditional probability of *next to* with 0.21 between $horse$ and $horse$ in CKG is higher than $ride$ with 0.11 between $person$ and $horse$, so we can get the correct visual triple $((person, b_1), next\ to, (horse, b_3))$ after iteratively updating with commonsense knowledge graph, as shown in SG_2. This suggests that it is important to consider CKG in visual relationship detection. While the task is challenging and there are at least three challenges:

1. CKG is a global graph for the image set rather than a graph that aims at one image, while visual relationship detection focuses on a given image.
2. A pair of object classes may have different relations even in the same image (e.g. "person" and "horse" show in the CKG of Figure 1), making it difficult to update CKG.
3. CKG and feature information of images should be considered jointly in order to facilitate visual relationship detection.

In this paper, by introducing the commonsense knowledge graph into visual relationship detection, we propose a novel model of iterative visual relationship detection framework with commonsense knowledge graph (IVRDC), which consists of a feature module and a commonsense knowledge module. The feature module is used to predict visual relationships by visual features and semantic features with a bi-directional recurrent neural network. The commonsense knowledge module outputs a predicate prediction based on the CKG, which initially is constructed according to the statistical frequency information of visual relationships from images. These two modules roll out iteratively and cross feed predictions to each other in order to update feature memory and CKG. On the one hand, the feature module provides a feedback to promote the global CKG to evolve towards the given image; on the other hand, the commonsense knowledge module offers commonsense to refine the estimates.

Finally, we evaluate our method by taking experiments on the VRD and Visual Genome (VG) datasets. Experiment results demonstrate that our proposed model outperforms the state-of-the-art methods.

The rest of this paper is organized as follows. We first introduce the preliminary in the next section. In the third section, we show our proposed model named iterative visual relationship detection with commonsense knowledge graph. Then we present the experiment results.

Due to space limit, omitted data, code, and supporting materials are provided in the online appendix (https://tinyurl.com/AAAI2019-4507).

2 Preliminary

In this section, we first recall the definitions of scene graph and visual relationship detection. Then we give the definition of commonsense knowledge graph. We also recall the bi-directional recurrent neural network used in our model.

2.1 Commonsense Knowledge Graph

[17] identified the *visual triples* of *scene graph*. We only consider entities and relations without attributes in this paper and give the definition of scene graph as follows. *W.l.o.g.* we assume that all images are in a finite set $\mathcal{I}$. All *classes* in $\mathcal{I}$ are in a finite set $\mathcal{C}$. All *predicates*[1] in $\mathcal{I}$ are in a finite set $\mathcal{P}$.

Definition 1 (Scene Graph). Given an image $I \in \mathcal{I}$, its scene graph is a set of *visual triples* $\mathcal{T}_I \subseteq \mathcal{O}_I \times \mathcal{P}_I \times \mathcal{O}_I$, $\mathcal{O}_I$ is the object set and $\mathcal{P}_I$ is the predicate set. Each object $o_{c,I,k} = (c, b_{I,k}) \in \mathcal{O}_I$ is packed with a class $c \in \mathcal{C}$ and a bounding box $b_{I,k}$ in image I, where $k \in \{1, 2, ..., |\mathcal{O}_I|\}$. A visual triple is of the form $(o_{c,I,k}, p, o_{c',I,k'}) \in \mathcal{T}_I$, where the two objects $o_{c,I,k}, o_{c',I,k'} \in \mathcal{O}_I$ and the predicate $p \in \mathcal{P}_I$. In general, we name $o_{c,I,k}$ as *subject* and $o_{c',I,k'}$ as *object*.

There are 4 objects, 2 predicates, and 4 visual relation triples in the scene graph $\mathcal{T}_I$ of Figure 1, *e.g.*, $((person, b_1), ride, (horse, b_2))$. For simplicity, we write it as $(person, ride, horse)$.

Definition 2 (Visual Relationship Detection).
Given an image $I \in \mathcal{I}$ and its object set $\mathcal{O}_I$, *visual relationship detection* is to detect the predicate p between two objects $o_{c,I,k}$ and $o_{c',I,k'}$, where $o_{c,I,k}, o_{c',I,k'} \in \mathcal{O}_I$ and $p \in \mathcal{P}_I$.

As shown in Figure 1, the predicate between "*person*" and "*sand*" is detected as "*on*" by the visual relationship detection. In other words, we construct a visual triple $(person, on, sand)$ for this image.

Definition 3 (Commonsense Knowledge Graph). Given an image set $\mathcal{I}$, the class set in $\mathcal{I}$ is $\mathcal{C}$, the predicate set in $\mathcal{I}$ is $\mathcal{P}$, the *commonsense knowledge graph* is a directed edge-labeled graph $\mathcal{G} = (\mathcal{C}, \mathcal{P}, \lambda)$, where each node $c \in \mathcal{C}$, each edge (c_s, p, c_o) represents a relationship between two nodes c_s and c_o ($c_s, c_o \in \mathcal{C}$, $p \in \mathcal{P}$), λ is the labeling function which means the confidence of (c_s, p, c_o), denoted by the conditional probability $P(p|c_s, c_o)$.

As shown in Figure 1, $(person, on, sand)$ is an edge with its confidence $P(on|person, sand) = 0.5$. Intuitively, if two nodes are irrelevant, the confidence of the edge between them is zero. There may exist some edges that connect the same node, *e.g.*, $(horse, next\ to, horse)$.

[1] Throughout this paper, we identify that the *predicate* in visual relationship detection is the *relation* in scene graph.

2.2 Bi-directional Recurrent Neural Network

Bi-directional recurrent neural network (Bi-RNN), proposed by [15], is successfully applied in natural language processing. [11] improved the original Bi-RNN and applied it to detect visual relation.

Bi-RNN In Bi-RNN, the vector x_i is the input of a sequence and y is the output, and h_i is a hidden layer. There are two hidden layers: a forward sequences $\overrightarrow{h}_i$ and a backward sequences $\overleftarrow{h}_i$.

$$\overrightarrow{h}_i = f(U_f x_i + W_f \overrightarrow{h}_{i-1} + b_f) \tag{1}$$

$$\overleftarrow{h}_i = f(U_b x_i + W_b \overleftarrow{h}_{i-1} + b_b) \tag{2}$$

$$y = \sum V_{f,i} \overrightarrow{h}_i + \sum V_{b,i} \overleftarrow{h}_i + b_y \tag{3}$$

where U_f and U_b denote the input-hidden weight matrixes. W_f and W_b denote the hidden-hidden weight matrixes. f is the activation function of the hidden layers (ReLU function). V_b and V_f denote the output-hidden weight matrixes. b_f, b_b and b_y denote the bias vectors.

While detecting visual relations between objects, the order of input sequences is of significance, because different orders can lead to distinct visual relationship detection results. For example, the visual relation between the object pair $(person, horse)$ can be totally different from that between the object pair $(horse, person)$. Bi-RNN can fit this character. To apply it to our model, we take the corresponding feature vectors of object pairs as inputs of Bi-RNN's and take the prediction vector as its output. The details will be shown in the next section.

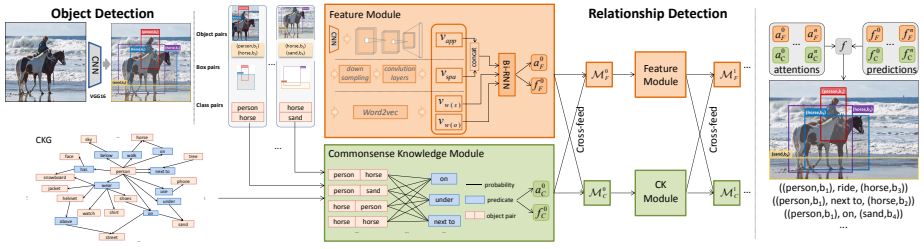

Fig. 2. The overview of our visual relation detection framework. Beside an object detector that gives a group of detected bounding boxes and their corresponding classification probability, the framework has two modules to perform detection: a feature module and a commonsense knowledge module. Both modules roll-out iteratively while cross-feeding beliefs. The final prediction f is produced by combining each prediction with attention mechanism.

3 Method

In this section, we propose a model named Iterative Visual Relationship Detection with Commonsense Knowledge Graph (IVRDC). The overall pipeline of IVRDC(Figure 2) is divided into *object detection* and *relationship detection*. Relationship detection consists of two modules: *feature module* and *commonsense knowledge module*. Both modules roll-out iteratively while cross-feeding beliefs. The final prediction is obtained by combining predictions from each iteration with attention mechanism.

In object detection, for each image I, we use Faster R-CNN [14] to obtain a group of bounding boxes and their classes and pack each bounding box $b_{I,k}$ with its class c together to be an object $o_{c,I,k}$. So for each image, we obtain several objects labeled with classes and the corresponding boxes.

Visual relation prediction is to predict visual triples $(subject, predicate, object)$. The feature module captures the interactions between objects by using feature vectors. And the commonsense module provides the conditional probability for reference. We construct a memory for iteration to store information. Then the model combines the outputs of the two modules, f_F and f_C, to update the two memories, $\mathcal{M}_F$ and $\mathcal{M}_C$. We will discuss the iteration and attention mechanism of each module in detail.

3.1 Feature Module

In the feature module, three features are taken into consideration: appearance feature, spatial feature and word vector. And the module employs Bi-RNN to learn those features to detect predicates [11].

We encode an image I of shape $H \times W \times C$, where H and W denote the height and the width, and C denotes the channels of the image. For our work, $C = 3$. For each image I, each candidate object $o_{c,I,k} = (c, b_{I,k}) \in \mathcal{O}_I$ has a bounding box $b_{I,k} = (x_{min}, y_{min}, x_{max}, y_{max})$ and its detected class c. Since visual information of an image can implicit interaction among objects and is particularly useful for visual relation detection, we construct an appearance feature v_{app} to encode visual information, which restores not only object features but also their context information. For preprocessing, we construct a new larger bounding box $b_{o,o'}$ to encompass the two boxes of an object pair (o, o'). We use VGG16 [16] to encode the region enclosed by $b_{o,o'}$, of shape $H' \times W' \times C$, where $H' = W' = 224$ and $C = 3$. The region through VGG16 net and we obtain the corresponding features. Then make it as inputs of a convolution net of two convolution layers and one 300-D fully-connected layer to get the appearance feature v_{app}.

Spatial information is also a key factor that influences our detection. The spatial feature is learned by a convolution neural network. In an image I, an object pair $(o_{c,I,k}, o_{c',I,k'})$ contains two bounding boxes $b_{o_{c,I,k}}$ and $b_{o_{c',I,k'}}$. First, we apply dual spatial masks for bounding boxes to get two binary masks, one for object $o_{c,I,k}$ and another for object $o_{c',I,k'}$. Then the masks are down-sampling to a predefined square (32×32) [5]. Finally, a convolution net of three convolution layers and a 300-D fully-connected layer take the masks as inputs to obtain the 300-D spatial feature v_{spa}.

Features mentioned above are visual features and express the relation between two objects. To consider the semantic feature and independence of objects, we represent an object class as a word vector. In this work, $Word2vec$ [13] is used to learn the word vectors. For an image I, each object $c_{c,I,k}$ has an object class c, then we can find the word vector corresponding to the name of c (*e.g.*, $person$). The relation between two words is an inherent semantic relationship instead of the mathematics distance with one-hot vector. Obviously, similar object pairs may have similar relationships. For example, the relationship between "$person$" and "$sand$" is normally "on". "$horse$" and "$person$" are similar in semantic space. Then it can reason that $(horse, on, sand)$. Similarly, some infrequent relations can be learned by the normal relation. For a pair of object $(o_{c,I,k}, o_{c',I,k'})$ in the image I, we generate two feature vectors, $v_{w(o_{c,I,k})}$ and $v_{w(o_{c',I,k'})}$, for subject and object, simplify as $v_{w(s)}$ and $v_{w(o)}$.

Before feeding features into a Bi-RNN, we concatenate appearance feature v_{app} and spatial feature v_{spa} and make the concatenated vector through a fully-connected layer to obtain visual feature v_{vis}. Then we combine the visual feature and semantic feature. Applying Bi-RNN to predict relationships, we feed feature vector $v_{w(s)}$, v_{vis} and $v_{w(o)}$ to input x_1, x_2, and x_3 (shown in equation (1) and (2)), respectively. The Bi-RNN structure is shown in Figure 3.

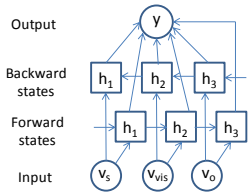

Fig. 3. Bi-RNN has three inputs in sequence ($v_{w(s)}$, v_{vis} and $v_{w(o)}$) and one output (predicate prediction y).

The output y is a $|\mathcal{C}|$-dimension vector. We use softmax function on y to compute the normalized probabilistic prediction to form the predicate f_F. The feature vector, including appearance feature, spatial feature and word vector, construct the memory $\mathcal{M}_F$ to store the visual and semantic information.

3.2 Commonsense Knowledge Module

In this part, we introduce how to construct the commonsense knowledge graph $\mathcal{G} = (\mathcal{C}, \mathcal{P}, \lambda)$ from a given image set $\mathcal{I}$ and how to use it in our framework. As defined before, the commonsense knowledge graph is a directed edge-labeled graph. First of all, we collect common sense from the training annotations and count the conditional probability that encodes the correlation between the pair of objects and the predicate. Each node $c \in \mathcal{C}$ represents an object class, *e.g.*, "$person$" in Figure1. Each edge (c_s, p, c_o) represents a relationship between two node c_s and c_o, *e.g.*,$(sand, under, person)$ in Figure1. λ is the labeling function that shows the conditional probability of the relationship. The conditional

probability can be formulated as:

$$P(p|c_s, c_o) = \frac{P(p, c_s, c_o)}{P(c_s, c_o)} \tag{4}$$

The commonsense knowledge graph is a universal graph, while visual relationship detection task is closely related to a particular image. To tackle this problem, we construct the subgraph of the commonsense knowledge graph $\mathcal{G}' = (\mathcal{C}', \mathcal{P}', \lambda)$. For each image, we use an pre-train object detector (Faster R-CNN) to detect objects, then according to the classes of those objects, we select all relative relationships from the global CKG. The nodes $\mathcal{C}'$ consist of the set of all the detected object classes and the edges $\mathcal{P}'$ are the corresponding relationships. This subgraph $\mathcal{G}'$ is only for the specific image with all the detected objects in the image and the connected edges. Then we set a threshold and distill the prediction f_C according to the subject node and the object node. When the prediction f_C updates, the subgraph will update the weights of the corresponding edges. Then we construct memory $\mathcal{M}_C$ to store the updated subgraph.

3.3 Iteration

The key component of the proposed model is to iteratively build up estimates. To deliver information from one iteration to another, we construct memories to store the information. The feature module uses feature memory $\mathcal{M}_F$ and the commonsense module use another memory $\mathcal{M}_C$.

At iteration i, the commonsense module distills the prediction $f_{C,i}$ and the feature module creates the prediction $f_{F,i}$ for a pair of candidate objects. Because $f_{C,i}$ ranges from 0 to 1, and the codomain of $f_{F,i}$ is uncertain, it is unreasonable to combine the two predictions directly. So the combination $f_{C,i+1}$ of the two predictions is given by

$$f_{C,i+1} = W_1 \circ f_{C,i} + W_2 \circ f_F \tag{5}$$

where W_1 and W_2 are weights, $f_{C,i+1}$ is the updated probability, $f_{C,i}$ denotes the result from the commonsense knowledge module, and f_F denotes the prediction from the feature module. Then $f_{C,i+1}$ can be used to get the updated memories, $\mathcal{M}_C^{i+1}$ and $\mathcal{M}_F^{i+1}$.

Then we update the feature memory $\mathcal{M}_F$ by a convolutional gate recurrent unit (GRU) [4]. F denotes a memory for a pair of candidate objects. F_{up} denotes a memory that we construct to update memory. We extract the appearance feature and the spatial feature from memory F. Then we combine $f_{C,i+1}$ with addition and convolution layers to form memory F_{up}. We update the feature memory as the following formula:

$$F_{i+1} = u \circ F_i + (1 - u) \circ \sigma(W_u F_{up} + W_F(r \circ F_i) + b) \tag{6}$$

where u denotes the update gate, r denotes the reset gate, F_{i+1} is the updated memory. W_f ,W_F, and b are convolutional weights and bias, and $\circ$ is entry-wise

product. $\sigma()$ is an activation function. After that, F_{i+1} is used to update memory $\mathcal{M}_F^{i+1}$.

The new memories, $\mathcal{M}_C^{i+1}$ and $\mathcal{M}_F^{i+1}$, will lead to another round of updated f_C and f_F and the iteration goes on. In this way, the feature memory can benefit from commonsense knowledge graph. At the same time, the subgraph of commonsense knowledge graph can get a better sense of the particular image.

3.4 Attention

To modify the model output, we generate the final prediction f by the combination of each iteration prediction instead of the last iteration prediction. To combine the predictions from each iteration, we introduce attention mechanism [3] to our framework. It means that the final output is a weighted version of all predictions using attentions. Mathematically, if the model iterate n times, then outputs $N = 2n$ (including n times feature module and n times commonsense module) prediction f_n by attention a_n, the final output f is represented as:

$$f = \sum_{n}^{N} w_n f_n \tag{7}$$

$$w_n = \frac{exp(a_n)}{\sum_{n'} exp(a_{n'})} \tag{8}$$

$$a_n = ReLU(W f_n + b) \tag{9}$$

where f_n is the logits before softmax w_n denotes the weight of each prediction, a_n is produced by f_n with an activation function $ReLU$. The introduction of attention mechanism enables the model to select feasible predictions from different modules and iterations.

3.5 Training

The total loss function consists of the feature module loss $\mathcal{L}_F$, the commonsense module loss $\mathcal{L}_C$ and the final prediction loss $\mathcal{L}_f$. To take more attention on the harder examples, we give different weights for the loss examples, based on the predictions from previous iterations. Then the cross-entropy loss is represented as:

$$w_{\mathcal{L}} = \frac{max(1.0 - p_{i-1}, 0.5)}{\sum max(1.0 - p_{i-1}, 0.5)} \tag{10}$$

$$\mathcal{L}_i = -w_{\mathcal{L}} log(p_i) \tag{11}$$

where p_i denotes the softmax output of the prediction for iteration i.

For model initialization, we use a pre-trained VGG-16 ImageNet model to initialize the CNN parameters of the appearance module and randomly initialize the spatial feature [5]. For word vectors for classes, we train our Word2vec model based on the class set and the triples in CKG. For Bi-RNN, it has two hidden layers and each layer has 128 hidden states. We roll out the feature module and the commonsense module three times and update the subgraph of commonsense knowledge graph at each iteration.

4 Experiments

We evaluate the proposed method on two recently released datasets. We first introduce datasets and experimental settings, and then analyze the experimental results in detail.

4.1 Datasets

We evaluate our proposed model on Visual Relationship Datasets (VRD) [12] and Visual Genome(VG) [8] shown in Table 1.

Table 1. Statistics of datasets

Dataset	$\|\mathcal{I}\|$	$\|\mathcal{C}\|$	$\|\mathcal{P}\|$	$\|\mathcal{I}_{train}\|$	$\|\mathcal{I}_{test}\|$
VRD	5,000	100	70	4,000	1,000
VG	99,659	200	100	73,801	25,857

VRD contains 5000 images with 100 object classes and 70 predicates. VRD contains 37,993 relation annotations with 6,672 type triples in total. Following the same train/test split as in [12], we split images into two sets, 4,000 images for training and 1,000 for testing.

VG contains 99,658 images with 200 object classes and 100 predicates. Totally, VG contains 1,174,692 relation annotations with 19,237 type triples. Following the experiments in [19], we split the data into 73,801 for training and 25,857 for testing.

4.2 Experimental Settings

According to [12], we use Recall@K as the major performance metric. Recall@K computes the fraction of times the correct relationships are predicted in the top K confident relationship predictions, as the following formula:

$$Recall@K = \frac{\sum_{i=1}^{\mathcal{I}} n}{\sum_{i=1}^{\mathcal{I}} m} \tag{12}$$

where n denotes the number of correct relationships in the top K confidence in i image, m denotes the number of the relationships labeled in ground truth in $i-th$ image. Following [12], we use Recall@50 (R@50) and Recall@100 (R@100) as evaluation metrics for our experiments. The reason using R@K is that the relationships in ground-truth are incomplete, in other words, some true relationships are missing.

Like [12], we evaluate our proposed method for the following tasks:

- **Predicate detection**: this task focuses on the accuracy of predicate prediction. The input includes the object classes and the bounding boxes of both the subject and object. In this condition, we can learn how difficult it is to predict relationships without the limitations of object detection.

– **Phrase/Union detection**: the task treats the whole triple $(sub, pred, obj)$ as a union bounding box which contains the subject and object. A prediction is considered correct if all the three elements in a triple are correct and the *Intersection over Union (IoU)* between the detected box and the ground truth bounding box is greater than 0.5.
– **Relationship detection**: this task treats a triple $(sub, pred, obj)$ as three components. A prediction is considered correct if three elements in a triple are correct and the IoU of subject and object are both above 0.5 with the ground-truth bounding box.

4.3 Comparative Results

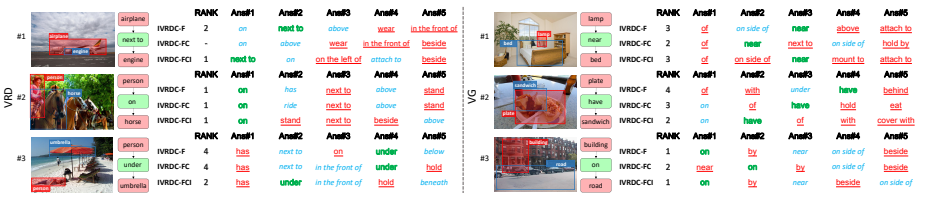

Fig. 4. Qualitative examples of relation prediction. We show the correct relation rankings and the top-5 answers from IVRDC-F, IVRDC-FC and IVRDC-FCI on VRD and VG. Relations in **bold**, *italic*, and underline fonts denote the correct, plausible, and wrong answers respectively.

Compare with other works. As mentioned above, visual relationship detection methods can be categorized into two branches: one takes external knowledge into consideration, such as LK [18] and VRL [10]; one only considers the internal knowledge of the dataset, which includes our model. So we do not compare our model with them and only compare with the state-of-the-art methods in the second branch. (1) **LP** [12] is a visual relationship detection model with appearance features and language prior. (2) **VTransE** [19] applies translation embedding to visual relationship detection and it is an end-to-end model with only visual features. (3) **PPR-FCN** [20] uses a parallel FCN architecture and a position-role-sensitive score map to tackle the task of "subject-predicate-object". (4) **DR-Net** [5] constructs appearance feature and spatial feature and makes full use of the statistical dependencies between objects and their relationships to predict visual relationships. (5) **DSL** [21] is a deep structured model that learns relationships by using the feature-level prediction and the label-level prediction. (6)**DSR** [9] is a newly designed model that can both facilitate the co-occurrence of relationships and mitigate the relation-incomplete problem.

Since the task of visual relationship detection is proposed, only VRD dataset is publicly released. All of proposed works conduct experiments in this dataset, and select data from the whole VG dataset [8] by themselves. Recently, VTransE has released their VG dataset. In VG dataset, we compare our proposed model

with the model applying the same implement methods, which use the same dataset to train and test, *e.g.*,VTransE, PPR-FCN, DSL, and DSR.

Table 2. Performances of predicate detection, phrase detection and relationship detection on VRD, comparing with several state-of-the-art methods. We use "-" to indicate that the performance has not been reported in the original paper.

Model	Predicate Det.		Phrase Det.		Relation Det.	
	R@50	R@100	R@50	R@100	R@50	R@100
LP	47.87	47.87	10.11	12.64	0.08	0.14
VTransE	44.76	44.76	19.42	22.42	14.07	15.20
PPR-FCN	47.43	47.43	19.62	23.15	14.41	15.72
DR-Net	80.78	81.9	19.93	23.45	17.73	20.88
DSL	-	-	**22.61**	**23.92**	17.27	18.26
DSR	**86.01**	**93.18**	-	-	**19.03**	**23.29**
IVRDC-F	86.21	94.00	21.67	28.88	14.57	19.22
IVRDC-FC	87.71	94.61	**22.92**	**30.20**	**15.52**	**20.16**
IVRDC-FI	86.34	93.78	21.83	28.63	14.67	19.02
IVRDC-FCI	**88.34**	**94.69**	22.28	28.73	15.06	18.97

We used Recall@50 (R@50) and Recall@100 (R@100) as evaluation metrics for predicate detection, phrase detection and relation detection. From the result on VRD in Table 2, we can see that our proposed model outperforms others in predicate detection. Our proposed model works best on predicate detection, which improved 2.33 and 1.51 for R@50 and R@100 respectively by previous models. As for phrase detection, our method achieved 30.20 at R@100, which is over 25% relative improvement over the previous best result. At relation detection, our model achieved the average level of previous state-of-the-art models.

From the result on VG in Table 3, it is clear that our method surpasses all other methods at predicate detection, our best result achieved 85.40 and 85.26 for R@50 and R@100 respectively, which outperforms other methods by over 23.5% and 18.6%. As for phrase detection and relation detection, our model still outperforms most of other state-of-the-art models.

Due to the long tail distribution of relationships, it is hard to collect enough training images for all the relationships, especially for infrequent relationships. So it is crucial for a model to have the generalizability on detecting zero-shot relationships. The performances on zero-shot predicate, phrase and relationship detection are reported in Table 4. We only compare our proposed model with the models using the same input, *e.g.*, LP, VTransE, and DSR. From the result on VRD in zero-shot learning demonstrated in Table 4, we can see that our proposed model works best on phrase detection. Our best result achieved 6.92 and 8.73 for R@50 and R@100 respectively. As for predicate detection, our method outperforms DSR for R@50. And our proposed model achieved the average level of the pervious models.

Table 3. Performances of predicate detection, phrase detection and relationship detection using various methods on VG. We use "-" to indicate that the performance has not been reported in the original paper.

Model	Predicate Det.		Phrase Det.		Relation Det.	
	R@50	R@100	R@50	R@100	R@50	R@100
VTransE	62.63	62.87	9.46	10.45	5.52	6.04
PPR-FCN	64.17	64.86	10.62	11.08	6.02	6.91
DSL	-	-	**12.07**	**14.35**	**6.37**	**7.50**
DSR	**69.06**	**74.37**	-	-	-	-
IVRDC-F	80.71	82.29	10.90	13.46	6.02	7.31
IVRDC-FC	83.26	**88.44**	10.55	13.85	6.04	7.32
IVRDC-FI	72.76	74.40	9.57	11.98	5.34	6.54
IVRDC-FCI	**85.40**	88.26	**11.12**	**14.00**	**6.13**	**7.60**

Table 4. Performances of zero-shot predicate detection, phrase detection and relationship detection using various methods on VRD. The methods without reporting the performance on zero-shot setting are excluded from comparison.

Model	Predicate Det.		Phrase Det.		Relation Det.	
	R@50	R@100	R@50	R@100	R@50	R@100
LP	-	-	**3.36**	3.57	3.13	3.52
VTransE	-	-	2.65	**3.75**	1.71	2.14
DSR	**60.90**	**79.81**	-	-	**5.25**	**9.20**
IVRDC-F	53.81	74.94	2.30	2.74	1.54	1.79
IVRDC-FC	60.05	**78.69**	3.76	5.39	2.22	2.82
IVRDC-FI	56.54	75.79	3.00	5.13	1.53	2.48
IVRDC-FCI	**61.76**	78.52	**6.92**	**8.73**	**3.93**	**4.53**

Compare different configs. We also compare different variants of the proposed model, in order to identify the contributions of individual components listed :

- IVRDC-F : Part of our model. We only use the feature module to predict the visual relationship without iteration.
- IVRDC-FI : Part of our model. We use the feature module and roll-out iteratively to obtain the predictions.
- IVRDC-FC : Part of our model. We combine the feature module and the commonsense knowledge module without an iteration.
- IVRDC-FCI : Our model introduced in Figure 2.

From Table 2 and Table 3, we observe that our best results outperform pervious best state-of-the-art results by up to 25%, and even our worst result achieved the average level. Moreover, our method with different components performs differently on the three detection tasks. IVRDC-FC is relatively strong in phrase detection and relation detection, while IVRDC-FCI performs better in predicate detection.

The results in Table 2, Table 3 and Table 4 show: (1) The joint model IVRDC-FC significantly performs better in phrase detection and relation detection, which means that CKG is very useful in visual relation detection. The combination of feature module and commonsense knowledge module considerably outperforms the model IVRDC-F with only feature module. (2) The model IVRDC-FCI performs best in predicate detection. It indicates that iteratively using image features and CKG have benefit on enhancing predication detection by making use of image feature information and commonsense knowledge. (3) Relation detection has achieved the average level. Since the relation detection depends a lot on the accuracy of the object detector, our result is probably limited by the performance of the object detector. By using the same object detector of VTransE, our result outperforms VTransE by 32.6% in R@100.

Figure 4 further shows the predicted relationships on serval example images. As the example $(plate, have, sandwich)$ in image VG#2 shown in Figure 4, IVRDC-F with image features performs better in predict predicate according images, *e.g.*, *under*. IVRDC-FCI is able to learn the meaning of *have* from combining CKG and iterations, bringing it to a higher ranking. Since commonsense knowledge is a statistical result, the more a predicate occurs, the higher the probability of the predicate will be, *e.g.*, *on*.

5 Conclusion and Future Work

In this paper, we present a model of iterative visual relationship detection where commonsense captured in the form of commonsense knowledge graph. In our model, the feature memory and the commonsense knowledge graph facilitate each other iteratively. The experimental results show that our model surpasses the state-of-the-arts methods. It is illustrated that the commonsense knowledge graph is capable of enhancing the visual relationship detection task.

So in the future work, we will focus on the representation of common sense and consider completing the commonsense knowledge graph using knowledge graph completion technics.

6 ACKNOWLEDGMENT

This paper was supported by the National Natural Science Foundation of China (No. 61375056, 61876204, 61976232, and 51978675), Guangdong Province Natural Science Foundation (No. 2017A070706010 (soft science), 2018A030313086), All-China Federation of Returned Overseas Chinese Research Project (17BZQK216), Science and Technology Program of Guangzhou (No. 201804010496, 201804010435).

References

1. Anderson, P., Fernando, B., Johnson, M., Gould, S.: SPICE: semantic propositional image caption evaluation. In: Proceedings of ECCV, 2016. pp. 382–398 (2016), https://doi.org/10.1007/978-3-319-46454-1_24
2. Bordes, A., Usunier, N., García-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Proceedings of International Conference on Neural Information Processing Systems (NIPS2013). pp. 2787–2795 (2013)
3. Chen, L., Yang, Y., Wang, J., Xu, W., Yuille, A.L.: Attention to scale: Scale-aware semantic image segmentation. In: Proceedings of CVPR, 2016. pp. 3640–3649 (2016), https://doi.org/10.1109/CVPR.2016.396
4. Chung, J., Gülçehre, Ç., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR **abs/1412.3555** (2014), http://arxiv.org/abs/1412.3555
5. Dai, B., Zhang, Y., Lin, D.: Detecting visual relationships with deep relational networks. In: Proceedings of CVPR, 2017. pp. 3298–3308 (2017), https://doi.org/10.1109/CVPR.2017.352
6. Dong, L., Wei, F., Zhou, M., Xu, K.: Question answering over freebase with multi-column convolutional neural networks. In: Proceedings of ACL, 2015. pp. 260–269 (2015), http://aclweb.org/anthology/P/P15/P15-1026.pdf
7. Johnson, J., Krishna, R., Stark, M., Li, L., Shamma, D.A., Bernstein, M.S., Li, F.: Image retrieval using scene graphs. In: Proceedings of CVPR. pp. 3668–3678 (2015), http://dx.doi.org/10.1109/CVPR.2015.7298990
8. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L., Shamma, D.A., Bernstein, M.S., Fei-Fei, L.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. International Journal of Computer Vision **123**(1), 32–73 (2017), https://doi.org/10.1007/s11263-016-0981-7
9. Liang, K., Guo, Y., Chang, H., Chen, X.: Visual relationship detection with deep structural ranking. In: Proceedings of AAAI, 2018 (2018), https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16491
10. Liang, X., Lee, L., Xing, E.P.: Deep variation-structured reinforcement learning for visual relationship and attribute detection. In: Proceedings of CVPR, 2017. pp. 4408–4417 (2017), https://doi.org/10.1109/CVPR.2017.469
11. Liao, W., Lin, S., Rosenhahn, B., Yang, M.Y.: Natural language guided visual relationship detection. CoRR **abs/1711.06032** (2017), http://arxiv.org/abs/1711.06032
12. Lu, C., Krishna, R., Bernstein, M.S., Li, F.: Visual relationship detection with language priors. In: Proceedings of ECCV, 2016. pp. 852–869 (2016), https://doi.org/10.1007/978-3-319-46448-0_51

13. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR **abs/1301.3781** (2013), http://arxiv.org/abs/1301.3781
14. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Proceedings of NIPS, 2015. pp. 91–99 (2015), http://papers.nips.cc/paper/5638-faster-r-cnn-towards-real-time-object-detection-with-region-proposal-networks
15. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE Trans. Signal Processing **45**(11), 2673–2681 (1997), https://doi.org/10.1109/78.650093
16. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014), http://arxiv.org/abs/1409.1556
17. Wan, H., Luo, Y., Peng, B., Zheng, W.: Representation learning for scene graph completion via jointly structural and visual embedding. In: Proceedings of IJCAI, 2018. pp. 949–956 (2018), https://doi.org/10.24963/ijcai.2018/132
18. Yu, R., Li, A., Morariu, V.I., Davis, L.S.: Visual relationship detection with internal and external linguistic knowledge distillation. In: Proceedings of ICCV, 2017. pp. 1068–1076 (2017), https://doi.org/10.1109/ICCV.2017.121
19. Zhang, H., Kyaw, Z., Chang, S., Chua, T.: Visual translation embedding network for visual relation detection. In: Proceedings of CVPR, 2017. pp. 3107–3115 (2017). https://doi.org/10.1109/CVPR.2017.331, https://doi.org/10.1109/CVPR.2017.331
20. Zhang, H., Kyaw, Z., Yu, J., Chang, S.: PPR-FCN: weakly supervised visual relation detection via parallel pairwise R-FCN. In: Proceedings of IEEE, 2017. pp. 4243–4251 (2017), http://doi.ieeecomputersociety.org/10.1109/ICCV.2017.454
21. Zhu, Y., Jiang, S.: Deep structured learning for visual relationship detection. In: Proceedings of AAAI, 2018 (2018), https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16475