# **One-shot Scene Graph Generation**

Yuyu Guo

Center for Future Media, University of Electronic Science and Technology of China, Chengdu, China yuyuguo1994@gmail.com

Lianli Gao

Center for Future Media, University of Electronic Science and Technology of China, Chengdu, China lianli.gao@uestc.edu.cn

# ABSTRACT

As a structured representation of the image content, the visual scene graph (visual relationship) acts as a bridge between computer vision and natural language processing. Existing models on the scene graph generation task notoriously require tens or hundreds of labeled samples. By contrast, human beings can learn visual relationships from a few or even one example. Inspired by this, we design a task named One-Shot Scene Graph Generation, where each relationship triplet (e.g., "dog-has-head") comes from only one labeled example. The key insight is that rather than learning from scratch, one can utilize rich prior knowledge. In this paper, we propose Multiple Structured Knowledge (Relational Knowledge and Commonsense Knowledge) for the one-shot scene graph generation task. Specifically, the Relational Knowledge represents the prior knowledge of relationships between entities extracted from the visual content, e.g., the visual relationships "standing in", "sitting in", and "lying in" may exist between "dog" and "yard", while the Commonsense Knowledge encodes "sense-making" knowledge like "dog can guard yard". By organizing these two kinds of knowledge in a graph structure, Graph Convolution Networks (GCNs) are used to extract knowledge-embedded semantic features of the entities. Besides, instead of extracting isolated visual features from each entity generated by Faster R-CNN, we utilize an Instance Relation Transformer encoder to fully explore their context information. Based on a constructed one-shot dataset, the experimental results show that our method significantly outperforms existing state-ofthe-art methods by a large margin. Ablation studies also verify the effectiveness of the Instance Relation Transformer encoder and the Multiple Structured Knowledge.

# CCS CONCEPTS

• Computing methodologies → Artificial intelligence; Computer vision; Scene understanding;

\*Jingkuan Song is the corresponding author.

MM '20, October 12-16, 2020, Seattle, WA, USA

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-7988-5/20/10...\$15.00

https://doi.org/10.1145/3394171.3414025

Jingkuan Song\*

Center for Future Media, University of Electronic Science and Technology of China, Chengdu, China jingkuan.song@gmail.com

Heng Tao Shen

Center for Future Media, University of Electronic Science and Technology of China, Chengdu, China shenhengtao@hotmail.com

## **KEYWORDS**

scene graph generation, prior knowledge, vision and language

#### **ACM Reference Format:**

Yuyu Guo, Jingkuan Song, Lianli Gao, and Heng Tao Shen. 2020. Oneshot Scene Graph Generation. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3414025

## **1 INTRODUCTION**

As essential tasks of vision understanding, image classification [10, 30, 36], image retrieval [28, 29, 34, 40], and object detection [7, 25, 26] are booming with the development of deep neural networks. However, general attributes of objects, such as category or location, are not adequate to understand image contents. A scene graph, which is a structured representation of the image content, contains not only the semantic and spatial information of objects in images but also relationships between instances. Since the scene graph possesses a wealth of visual contents, the study of scene graph generation facilitates other high-level tasks in the multimedia field [4, 5], such as visual captioning [9, 31, 32] and visual question answering (VQA) [6, 18, 19, 33].

In general, previous methods on scene graph generation focus on the following aspects. 1). How to propose an efficient messagepassing mechanism between object features to get the local or global context [20, 41, 43–45]? 2). How to effectively map visual relationships to a semantic space [23, 24, 46]? 3). How to design a multi-task network to enhance the scene graph generation task [8, 21]? Because the semantic space of visual relationships is tremendous, these methods usually require a large number of labeled samples as supervision information. However, based on rich prior knowledge in the brain, humans can easily overcome this difficulty and learn visual relations from few or even one example.

In order to equip models with the ability to learn visual relationships from one example, we design a new task called One-Shot Scene Graph Generation, where each relationship triplet contains only one annotated example, as shown in Figure 1. Due to a lack of sufficient supervision information, this task is difficult for the previous work, as illustrated in Figure 2. Directly applying existing scene graph generation models on this task leads to a significant performance drop.

As mentioned above, rich prior knowledge helps humans to learn visual relationships from few examples. This suggests that models should pay attention to not only visual information but also

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.



Figure 1: In this paper, we focus on the one-shot scene graph generation task, where each relationship triplet (e.g., "dog-has-head") comes from only one labeled example. For clarity, bounding boxes in the figure are not shown.



Figure 2: MotifNet [45] is applied to the one-shot scene graph generation task. The performances are evaluated on the PredCls (Recall@K) setting. The blue bars denote the performances of MotifNet on the scene graph generation task. The orange bars denote the performances of MotifNet on the one-shot scene graph generation task.

other information from prior knowledge [12, 22, 35]. In this paper, the Multiple Structured Knowledge (Relational Knowledge and Commonsense Knowledge) is introduced into the one-shot scene graph generation task. The Relational Knowledge represents prior knowledge of relationships between entities. For example, there is a high probability that the relationship "play" exists between "person" and "dog", even if the image is not visible. The Commonsense Knowledge precisely locates entities in the commonsense and helps models to reason effectively. When we have known the facts "horse is an animal" and "man can raise horses", it is natural to infer "man can raise animals", even if we have not seen other animals.

In order to handle the one-shot scene graph generation task, the Multiple Structured Knowledge is introduced into our method as following steps: 1) Encoding instance features with an Instance Relation Transformer; 2) Extracting the Relational Knowledge and the Commonsense Knowledge from knowledge bases; 3) Organizing the prior knowledge into graph structures; 4) Encoding the graph-structured knowledge with GCNs; and 5) Combining the GCNs and the Instance Relation Transformer for relationship predicate classification. Specifically, motivated by the Transformer [38] structure, we propose the Instance Relation Transformer encoder to capture the relational context among instances in an image. Then the Relational Knowledge is extracted from a relation knowledge base (Visual Genome [16]), and the Commonsense Knowledge is obtained from a commonsense knowledge base (ConceptNet [35]). These large knowledge bases consist of many loose triplets, and it is unwieldy to obtain knowledge features from these triplets. In this paper, the triplets in the knowledge bases are organized into graph structures. Graph Convolutional Networks (GCNs) [14] encode the graph structures to extract knowledge features. Finally, the outputs of the Instance Relation Transformer and the GCNs are combined to predict relationships between instances.

The contributions of this paper can be summarized as: 1) To imitate the way human beings understand visual relationships, this work first defines the one-shot scene graph generation task, where the supervision information of each relationship triplet only comes from one labeled example; 2) Relational Knowledge and the Commonsense Knowledge are introduced into the one-shot scene graph generation task. The Relational Knowledge provides the prior knowledge about the relationships of entities, and the Commonsense Knowledge encodes "sense-making" knowledge. An Instance Relation Transformer encoder is utilized to explore the context information of visual entity for scene graph generation; and 3) We collect a new dataset for the one-shot scene graph generation task, where each relationship (subject, predicate, object) contains only one annotated example. Experiments show that our method significantly outperforms existing state-of-the-art methods by a large margin.

#### 2 RELATED WORK

### 2.1 Scene Graph Generation

The scene graph defined by Johnson et al. [13] is composed of a series of nodes and edges. The nodes are represented by entities in images, which contain categories and locations of entities. The edges are represented by visual relationship predicates between entities, such as "on", "in" and "under". As mentioned above, the previous methods for the scene graph generation task are roughly based on the following three perspectives: 1). Building a semantic space for visual relationships from a language model [23] or a Translation Embedding [46]; 2). Extracting contextual information from bipartite sub-graphs [45], global architectures [45], or tree structures [37]; 3). Combining with multi-tasks, such as dense captioning [21] and image reconstruction [8]. Most of these efforts need a large-scale annotated dataset, which requires much labor. However, human beings can understand visual relationships from few examples. To imitate such ability, we design a one-shot scene graph task and introduce rich knowledge from human beings to handle this task.



Figure 3: The framework of our method. Given an image, we first detect the instances in the image with Faster R-CNN. Then the Instance Relation Transformer network is proposed to explore the contextual information among instances. Next, the Relational Knowledge Extractor extracts the relational knowledge features from Visual Genome, and the Commonsense Knowledge Extractor extracts the commonsense knowledge features from ConceptNet. Finally, the relationship predicates are predicted with outputs of the Instance Relation Transformer, the Relational Knowledge Extractor, and the Commonsense Knowledge Extractor.

#### 2.2 One-Shot learning

Many works [17, 27, 39] have shown that the machine can understand a wealth of information from one example. Feifei et al. [17] explored general knowledge from learned categories and utilized a Bayesian method to implement one-shot learning of object categories. However, in their work, the prior knowledge contains only three categories. This weakens the generalization of the method. To alleviate this issue, Salakhutdinov et al. [27] collected a set of super-categories to represent different priors for new categories and constructed a hierarchical Bayesian model for learning from one example. Both of these Bayesian methods lack powerful image features. Due to the success of deep learning in image representation, Koch et al. [15] proposed a siamese neural network that combines the convolutional networks and the metric learning strategy. Orthogonal to the above methods, Wang et al. [39] designed a multi-attention network that generates the image features from the category semantic embedding. Different from the above works, our work focuses on the scene graph generation task under the oneshot environment and adopts the rich knowledge from knowledge bases to support this task.

# 3 APPROACH

In this paper, we introduce the Multiple Structured Knowledge into the one-shot scene graph generation task. The framework of our method is depicted in Figure 3. The proposed method contains four main components: 1) An object detector; 2) An Instance Relation Transformer encoder; 3) A Relational Knowledge extractor; and 4) A Commonsense Knowledge extractor. In this section, we first define the one-shot scene graph generation task and then introduce each part from inputs to outputs.

# 3.1 Problem Definition of One-Shot Scene Graph Generation

Given an image, a scene graph is defined as a set of nodes and edges, where the nodes represent instances in the image, while the edges represent the relationships between instances. The scene graph can be divided into relationship triplets <subject, predicate, object>, where the subject and object are the instances detected by an object detector. This task requires a model to predict relationship predicates between instances. For the one-shot scene graph generation task, the ground truth of each relationship triplet <subject, predicate, object> contains only one labeled example as supervision information. Specifically, a one-shot dataset from the Visual Genome dataset is built to support this task. During the construction process, we first initial the oneshot dataset D with none. Next, all images in the Visual Genome dataset are checked. An image faces two situations: 1). When an image contains relationship triplets that do not appear in dataset D yet, we add the image and the corresponding annotations of the relationship triplets to dataset D. 2). When all relationship triplets of an image have appeared in D, the dataset D skips the image. The one-shot dataset D is collected to verify our method on the one-shot scene graph generation task.

## 3.2 Object Detector

In this paper, we adopt Faster R-CNN to generate *n* instances, which include the following information:

- Label probabilities L = {l<sub>1</sub>, ..., l<sub>n</sub>}, where L ∈ ℝ<sup>n×d<sup>a</sup></sup> and d<sup>a</sup> is the number of instance categories;
- Bounding boxes  $B = \{b_1, ..., b_n\}$ , where  $B \in \mathbb{R}^{n \times 4}$ ;
- Object features  $F = \{f_1, ..., f_n\}$ , where  $F \in \mathbb{R}^{n \times d^c}$  and  $d^c$  is the feature dimension;
- Union object features  $U = \{u_{1,1}, ..., u_{n,n}\}$ , where  $U \in \mathbb{R}^{n \times n \times d^c}$ .

The object feature  $f_i$  is extracted from bounding box  $b_i$ . The label probability  $l_i$  is generated with Faster R-CNN from bounding box  $b_i$ . The union object feature  $u_{i,j}$  is extracted from the union bounding box of  $b_i$  and  $b_j$ .

#### 3.3 Instance Relation Transformer

Generating a complete scene graph requires not only the visual features of instances but also the contextual information. For example, when we know "people feed horses", we should also increase the confidence of "horses on the ground". However, the isolated features *F* from Faster R-CNN ignore the surrounding context, and it is necessary to use an effective strategy to get the context in an image for the scene graph generation task. Most of the previous methods [37, 41, 45] utilize RNNs to obtain the global or local context. Nevertheless, as stated by Vaswani *et al.* [38], RNNs have defects in parallelization, computational complexity, and long-term dependence. In this paper, in order to understand the context of relationships effectively, the Transformer network is utilized for encoding the instance features. Since the Transformer structure can explore the relationship among inputs, this structure is suitable for capturing the relational context in an image.

We construct the Instance Relation Transformer and generate the relation context features  $M \in \mathbb{R}^{n \times d^z}$  by applying the Transformer structure to the instance information, i.e., visual features, label embeddings, and position embeddings:

$$M = \text{Transformer}([F, E^g, E^v]; W^z), \qquad (1)$$

where  $W^z$  is a parameter set in the Transformer.  $E^g$  and  $E^v$  are the embedding vectors from label probabilities *L* and bounding boxes *B*, respectively. *F* is the instance feature mentioned in Section 3.2. [:, :, ] means the concatenate operation.

### 3.4 Relational Knowledge Extractor

As discussed in the Section 1, the Multiple Structured Knowledge (Relational Knowledge and Commonsense Knowledge) is introduced into the one-shot scene graph generation task. The Relational Knowledge extractor and the Commonsense Knowledge extractor are designed to capture the relational knowledge features and commonsense knowledge features, respectively.

We first introduce the Relational Knowledge, which contains the prior knowledge of the relationship between entities in the visual space. Specifically, the Relational Knowledge is extracted from a relation dataset: Visual Genome. Visual Genome bridges the gap between computer vision and natural language processing, and can be used for many tasks, such as VQA, image captions, and scene graphs. In particular, we use scene graph labels filtered by Xu et al. [41] as our knowledge base. A series of triplets <subject, predicate, object>, e.g., <pillow, on, bed>, represent the scene graph in Visual Genome. All triplets in the training dataset are organized into a Relational Knowledge base  $K^v$ , which contains 200 entity categories  $C^q$  that include subjects, objects, and predicates. The structured knowledge is represented as a set of adjacency matrices and entity vectors. In this work, two boolean adjacency matrices  $A^o \in \mathbb{R}^{200 \times 200}$  and  $A^p \in \mathbb{R}^{200 \times 200}$  are constructed to represent the Relational Knowledge. The boolean adjacency matrix A<sup>o</sup> represents whether there are triplets between entity categories. To capture predicate information, the adjacency matrix  $A^p$  focuses on the relationship between objects/subjects and predicates. For example, the relationship triplet <pillow, on, bed> is contained in the Relational Knowledge base  $K^v$ , and x, y and z are the indexes of "pillow", "bed", and "on" in  $C^q$ , respectively. Then the element  $a^o_{x,y}$  of the boolean adjacency matrix  $A^o$  is set to 1, and the elements  $a_{x,z}^p$  and  $a_{z,y}^p$  of the boolean adjacency matrix  $A^p$  are both set to 1.

After obtaining the boolean adjacency matrices, the Word2Vector method is adopted to extracts the entity vectors  $E^p$  according to the corresponding vocabularies in the categories  $C^q$ . In order to capture the structured information in the adjacency matrices  $A^o$  and  $A^p$ , Graph Convolutional Networks (GCNs) [14] are utilized for encoding the entity vectors  $E^p$ :

$$\begin{array}{rcl}
O^{v1} &= & \text{GCNs}(E^{p}, A^{o}; W^{g1}) , \\
O^{v} &= & \text{GCNs}(O^{v1}, A^{p}; W^{g2}) , 
\end{array}$$
(2)

where  $W^{g1}$  and  $W^{g2}$  are parameters in GCNs. We get the relational knowledge features  $O^v \in \mathbb{R}^{200 \times d^z}$  of the category set  $C^q$ . Finally, we find the indexes of label *L* (mentioned in Section 3.2) in  $C^q$ , and then use these indexes to select the corresponding features from  $O^v$  to form the Relation Knowledge features  $P^v \in \mathbb{R}^{n \times d^z}$  of label *L*.

### 3.5 Commonsense Knowledge Extractor

The Commonsense Knowledge defines the exact meaning of entities, which can assist the cognition and reasoning of the model. Inspired by [35], a commonsense knowledge base (ConceptNet) is used to obtain the Commonsense Knowledge. ConceptNet contains many loose triplets <head, relation, tail>, such as <dog, desires, play> and <frisbee, usedfor, play>. The head and the tail represent a head concept and a tail concept in ConceptNet, respectively. The relation represents a semantic relationship, such as "desires", "has property" and "is used for". The commonsense information in ConceptNet facilitates a model to understand the definitions of objects. However, the ConceptNet dataset is large and hard to be used directly. It is necessary to refine and filter the ConceptNet dataset. In order to increase the density of ConceptNet, the original relation categories are deleted and merged following the approach from Lin *et al.* [22].

As mentioned above, it is difficult to mine valuable information from loose triplets in ConceptNet. Previous work [8] utilizing LSTM to directly encode the loose triplets can not effectively extract the structured knowledge among triplets. In our work, we build a subgraph from simple paths constructed by the loose triplets and use GCNs on the subgraph to extract the knowledge features. To construct the subgraph, the method [22] is adopted in ConceptNet to find and prune simple paths constructed by triplets. We first locate the instance labels of L (mentioned in Section 3.2) in the concepts of ConceptNet. Then, for each label pair of *i*-th label  $l_i$  and *j*-th label  $l_i$ , all simple paths between  $l_i$  and  $l_i$  along triplets in ConceptNet are checked. If the length of a simple path is shorter than five, it is reserved. Otherwise, it is discarded. For path pruning, each triplet in a path is rated with the TranSE [1] method first. The score of each path is the product of the scores of triplets in the path. Then the paths with scores less than 0.15 are filtered out. Finally, for an image, the concepts in the filtered paths of all label pairs are organized into a new category set  $C^c$ . A boolean adjacency matrix  $A^{c}$  is used to indicate whether two concepts are adjacent in the filtered paths.

Similar to Relational Knowledge Extractor, the Word2Vector method extracts entity vectors  $E^c$  from the category set  $C^c$ , and GCNs capture commonsense knowledge features from  $A^c$  and  $E^c$ :

$$O^c = \operatorname{GCNs}(E^c, A^c; W^{g3}).$$
(3)

 $O^c \in \mathbb{R}^{|C^c| \times d^z}$  is the common sense knowledge feature of  $C^c$ , and  $|C^c|$  is the number of elements in the category set  $C^c$ . The common sense knowledge features  $P^c \in \mathbb{R}^{n \times d^z}$  of the instance labels are extracted from  $O^c$  according to the indexes of L in  $C^c$ .

Until now, we can obtain the semantic features of the Multiple Structured Knowledge ( $P^v$  and  $P^c$ ).

#### 3.6 Predicate Classification

In order to represent the detected instance, we sum the outputs from the Instance Relation Transformer (M), the Relational Knowledge extractor ( $P^v$ ), and the Commonsense Knowledge extractor ( $P^c$ ):

$$E^r = P^v + P^c + M . ag{4}$$

Because the same instance is inconsistent in the subject space and the object space,  $E^r$  is mapped to the subject space and object space with fully connected (FC) layers:

$$E^{s} = FC(E^{r}; W^{s}),$$
  

$$E^{o} = FC(E^{r}; W^{o}),$$
(5)

where  $W^{s}$  and  $W^{o}$  are parameters.  $E^{s} = \{e_{1}^{s}, e_{2}^{s}, ..., e_{n}^{s}\}$  and  $E^{o} = \{e_{1}^{o}, e_{2}^{o}, ..., e_{n}^{o}\}$ .

The DisMult [42] method predicts relation predicate between *i*-th instance and *j*-th instance:

$$r_{i,j,k} = (e_i^s \circ u_{i,j})^T W_k^r (e_j^o \circ u_{i,j}) + b_{i,j,k}^r , \qquad (6)$$

where  $r_{i,j,k}$  is the probability that the *k*-th relation predicate exists between the *i*-th instance and the *j*-th instance.  $W_k^r$  is a diagonal

parameter matrix for the *k*-th relation predicate.  $b_{i,j,k}^r$  is a frequency baseline proposed by [45]. In addition,  $E^s$  and  $E^o$  are also used to predict the probability  $r'_{i,j}$  that indicates the probability of non-background relationship between the *i*-th instance and the *j*-th instance:

$$r'_{i,j} = (e^s_i \circ u_{i,j})^T W^{r'}(e^o_j \circ u_{i,j}) + b^{r'}_{i,j} .$$
<sup>(7)</sup>

If  $r'_{i,j} = 1$ , there is a non-background relationship between the *i*-th instance and the *j*-th instance. Finally, we apply the softmax function to  $r_{i,j,k}$  and  $r'_{i,j}$ , and use the cross entropy function to learn the parameters of the model.

### 4 EXPERIMENTS

In this section, we conduct experiments on two datasets: the Visual Genome dataset and the One-Shot Visual Genome dataset. Firstly, in order to verify whether a model can learn each visual relationship from one example, our method and some existing methods are evaluated on the one-shot scene graph task. Secondly, a detailed ablation study is conducted on the one-shot scene graph generation task to verify the effectiveness of each component. Then, we show that our method can also handle the scene graph generation task properly. Finally, some visual results are shown for qualitative analysis.

# 4.1 Datasets

**Visual Genome.** For modeling relationships in our visual world, Krishna *et al.* [16] collected a dense annotation dataset called Visual Genome, where each image is annotated with objects, attributes, and relationships. The Visual Genome dataset contains over 100*K* images. Each image contains about 21 instances, 18 attributes, and 18 relationship triplets. However, it is difficult for models to learn stable information due to a lot of low-quality annotations in this dataset. Therefore, Xu *et al.* [41] brought forward an approach to filter the low-quality annotations, which is widely used in other works [2, 37, 43, 45].

Each image contains about 12 instances and 6 relationship triplets in the filtered Visual Genome dataset. The dataset contains 150 instance categories and 50 predicate categories in total. The training set and testing set account for 70% and 30% of the entire dataset, respectively. Moreover, following previous works [8, 45], the validation set consists of 5k images from the training set. The previous works [8, 21, 44] also proposed other different filtering strategies. This paper ignores the different filtering strategies and employs the filtering strategy proposed by Xu *et al.* [41] to verify our method.

**One-Shot Visual Genome.** To handle the one-shot scene graph generation task, we collect an extreme training dataset from the Visual Genome dataset, where each relationship triplet only appears once. A relationship triplet set is constructed from the training set proposed by [41]. The triplet set contains about 29K relationship triplets in total. For each triplet, we randomly select an image for forming the One-Shot Visual Genome dataset. Finally, the dataset contains about 18K images. Since an image may contain several different triplets, the number of images is less than the number of triplets. It is worth emphasizing that we only use this dataset to train models, while the test dataset remain the same as the test set of Visual Genome.

Table 1: Existing methods decline significantly on the one-shot scene graph generation task. SGG denotes scene graph generation. OSSGG denotes one-shot scene graph generation. MSK denotes the Multiple Structured Knowledge. IRT denotes the Instance Relation Transformer.

Dataset		PredCls				SCCla		SCDat		
	Method				SGCIS			SGDet		
	Method	R@20	R@50	R@100	R@20	R@50	R@100	R@20	R@50	R@100
SGG	FREQ+OVERLAP [45]	53.6	60.6	62.2	29.3	32.3	32.9	20.1	26.2	30.1
	IMP+ [41]	52.7	59.3	61.3	31.7	34.6	35.4	14.6	20.7	24.5
	MotifNet [45]	58.5	65.2	67.1	32.9	35.8	36.5	21.4	27.2	30.3
	IRT (Ours)	60.3	66.8	68.5	33.9	36.9	37.5	21.9	27.8	31.0
OSSGG	FREQ+OVERLAP <sup>1</sup>	5.0	8.8	11.3	3.3	5.1	5.9	1.4	2.8	4.0
	IMP+ <sup>1</sup>	36.4	45.3	48.3	19.7	23.8	25.0	4.5	8.6	12.6
	MotifNet <sup>1</sup>	33.5	43.6	47.1	17.4	22.0	23.4	6.2	9.4	11.8
	IRT (Ours)	37.7	46.1	48.9	21.2	25.2	26.4	6.8	11.1	15.0
	IRT+MSK (Ours)	41.3	49.5	52.2	23.0	26.9	28.1	7.1	11.5	15.5

### 4.2 Implementation Details

For the object detector, Faster R-CNN with RoI Align provided by Zellers [45] is used to detect instances in images, and its parameters are frozen. Besides the Instance Relation Transformer mentioned above for predicting relationship predicates, we utilize another Instance Relation Transformer to refine the instance labels on the scene graph generation task. The depth and width of the Instance Relation Transformers are both set to 6, and the dimension is set to 768. For experiments on the one-shot scene graph generation task, the depth and width of the Instance Relation Transformer are both set to 12 and the dimensions to 768. The numbers of layers of GCNs in Equation 2 and Equation 3 are 2, and the dimensions of GCNs are 768.

We use the SGD method to learn the parameters. The learning rate and the batch size are set as  $5 \times 10^{-3}$  and 16, respectively. The maximum number of epochs is 50. Python and Pytorch are adopted to build our model. All the experiments are carried out on the Ubuntu system with 256 GB RAM, a Titan Xp (12 GB) GPU, and 4 Intel(R) Xeon(R) E5-2650 CPUs.

#### 4.3 Evaluation Strategies

Following previous works, we use three setups (PredCls, SGCls and SGDet) to evaluate our method. The PredCls (predicate classification) task predicts relationship predicates with the ground truths of bounding boxes and categories in the test phase. The SG-Cls (scene graph classification) task allows models to employ the ground truths of bounding boxes in the test phase. The SGDet (scene graph detection) setup requires the model to generate bounding boxes, categories, and relationship predicates in the test phase without any ground truths. All three cases are evaluated by Recall@K (R@K, K=20,50,100). In this paper, we show all the results with graph constraint, i.e., each instance pair produces a relationship triplet.

# 4.4 Experimental Results on the One-Shot Scene Graph Generation task

As mentioned above, human beings can learn stable information from few samples or even one sample. In order to check whether models possess such ability, we show the experimental results of our method and existing methods on the one-shot scene graph generation (OSSGG) task.

Due to the lack of sufficient supervision information, the OSSGG task is more difficult than the scene graph generation task. Directly applying existing methods to the OSSGG task can lead to a significant performance drop, as shown in Table 1. It can be seen that the decline of FREQ+OVERLAP is the most severe since it just relies on the bias of the dataset. It is noteworthy that the decline rate of our IRT is lower than that of MotifNet. For example, on R@20 of PredCls, the rate of decline for our IRT is 37.5% ((60.3 – 37.7)/60.3), and the rate of decline for MotifNet is 42.7% ((58.5 – 33.5)/58.5). This shows that the contextual information extracted by our IRT is more robust than MotifNet, which uses Bi-LSTM to extract the global context.

Moreover, the Multiple Structured Knowledge enhances the Instance Relation Transformer on the OSSGG task. Under the conditions of PredCls and SGCls, the Multiple Structured Knowledge notches up high growth rates ((41.3 - 37.7)/37.7 = 9.5% and (23.0 - 21.2)/21.2 = 8.5%). On the SGDet task, the growth rates brought by the Multiple Structured Knowledge are marginal, because the SGDet setting depends heavily on object detection. However, the emphasis of our work is predicate prediction rather than object detection. These experiments verify the validity of the Multiple Structured Knowledge on the OSSGG task.

In general, compared with the above methods, our method achieves the best on the scene graph generation task as well as the one-shot scene graph generation task.

#### 4.5 Ablation Study

In order to profoundly analyze our method and illustrate the effectiveness of each component, we conduct a set of detailed ablation experiments on the one-shot scene graph generation task.

The role of Multiple Structured Knowledge. To further illustrate the role of Multiple Structured Knowledge, we test the effects of the Multiple Structured Knowledge and visual features, as shown in Table 2. Using ResNet enhances the results of IRT with an increase of 2.8% on R@20 compared with VGG. This means that a powerful visual feature can improve the robustness of IRT on the OSSGG task. Moreover, the results of IRT(V)+MSK are better than IRT(R) (41.3 vs. 40.5). These results illustrate the effectiveness of our Multiple Structured Knowledge. When we use both ResNet and Table 2: Ablation study of visual features and Multiple Structured Knowledge. V denotes VGG-16. R denotes ResNet-101.

Method	PredCls						
Methou	R@20	R@50	R@100				
IRT(V)	37.7	46.1	48.9				
IRT(R)	40.5	48.6	51.2				
IRT(V)+MSK	41.3	49.5	52.2				
IRT(R)+MSK	41.8	51.3	54.3				

Table 3: Ablation study of the Commonsense Knowledge and the Relational Knowledge. CK denotes the Commonsense Knowledge from the ConceptNet dataset. RK denotes the Relational Knowledge from the Visual Genome dataset.

СК	RK	IRT	PredCls						
			R@20	R@50	R@100				
		$\checkmark$	37.7	46.1	48.9				
	$\checkmark$	$\checkmark$	39.9	48.2	51.0				
$\checkmark$	$\checkmark$	$\checkmark$	41.3	49.5	52.2				

Table 4: Ablation study of the structured knowledge from Visual Genome.  $A^o$  is the adjacency matrix with instance categories mentioned in Section 3.4.  $A^p$  is the adjacency matrix with predicate categories.

A <sup>p</sup>	Ao	IRT	PredCls						
			R@20	R@50	R@100				
			37.7	46.1	48.9				
	$\checkmark$		38.7	47.5	50.3				
	$\checkmark$		39.9	48.2	51.0				

Multiple Structured Knowledge, the model achieves the best. This shows the adaptability of our Multiple Structured Knowledge.

The roles of the Commonsense Knowledge and the Relational knowledge. In order to further investigate the effect of the structure knowledge, we gradually add different types of knowledge to IRT, as shown in Table 3. Without any additional knowledge, IRT achieves presentable results, which indicate that IRT can take advantage of the contextual information to predict relationships. After adding the Relational Knowledge (RK) for IRT, the results are improved, e.g., the result increases by 2.2% on R@20. This shows that the relational knowledge features are effectively extracted and utilized in our model. Finally, the model achieves the best, when the Commonsense Knowledge (CK) is further added. These results indicate that providing the Relational Knowledge and the Commonsense Knowledge for IRT can make up for the lack of supervision information of the OSSGG task to some extent.

**The roles of**  $A^o$  **and**  $A^p$  **in Visual Genome.** Furthermore, we explore the influence of the structured knowledge from Visual Genome. As mentioned in Section 3.4, two adjacency matrices ( $A^o$  and  $A^p$ ) are used to encode the structured knowledge from Visual Genome.  $A^o$  is the adjacency matrix with instance categories mentioned in Section 3.4.  $A^p$  is the adjacency matrix with predicate categories. Table 4 shows that both matrices improve the effectiveness of the model, e.g.,  $A^o$  improves the result of IRT by 1.0% on R@20, and  $A^p$  further improves the result by 1.2%. This shows that the model requires not only the prior knowledge from instance categories ( $A^o$ ) but also the prior knowledge from relationship predicates ( $A^p$ ).

# 4.6 Experimental Results on the Scene Graph Generation task

In this section, we evaluate our method on the scene graph generation task to illustrate the universality of our approach. We compare our method with previous methods, including: methods adopting supervised learning [3, 11, 23, 24, 41, 43, 45], and methods adopting reinforcement learning [2, 37], as shown in Table 5.

From Table 5, our IRT outperforms the state-of-the-art methods with the supervised learning strategy, such as KERN [3], MotifNet [45], and IMP [41]. This shows that our IRT can fully extract and utilize the contextual information among instances to assist the relationship prediction. After adding Multiple Structured Knowledge, the performances are also improved. However, the improvement is marginal. Because the task already has rich annotations that allow the model to learn stable knowledge, the additional knowledge is trivial for the scene graph generation task.

We also compare our method with the methods using reinforcement learning [2, 37] and observe the following results. Firstly, our method achieves comparable results on the SGDet setup. The setup depends heavily on the results of object detection. Because the focus of our method and the previous methods is not the object detection task, we get similar results on the SGDet setup. Secondly, the methods based on the reinforcement learning strategy get better results on the SGCls because the SGCls setup focuses more on the instances classification results than the PredCls setup. [2, 37] use the reinforcement learning strategy to enforce a high penalty on the misclassification of prominent instances, resulting in a better performance on the SGCls setup. Thirdly, on the PredCls setup, our method is better than reinforcement learning based models since the PredCls setup ignores the effect of instance categories. These results show the superiority of our method for the relationship predicate prediction. Moreover, compared with these methods, the training process of our method is more concise.

#### 4.7 Qualitative Results

In this section, we show some qualitative results in Figure 4. With the Relational Knowledge and the Commonsense Knowledge, our IRT predicts some spatial relationships correctly, e.g., <shirt, above, shot> and <racker, on, hand>, in the first example, and some commonsense relationships, e.g., <dog, has, head> and <dog, has nose>, in the second example. We also show some notable failures in the third and fourth examples. In the third example, due to the excessive overlap between *man2* and *man3*, the model outputs that *man3* wears the T-shirt of *man2*. In the fourth example, the model outputs <br/> <br/> <br/> shird, on, window> because the bounding box of the window completely contains the bird, which is a strong signal for the relationship predicate "on". These two failure examples are due to the bias of spatial information. Properly reducing the dependence on spatial information may help to alleviate this problem.

<sup>&</sup>lt;sup>1</sup> We use the codes provided by Zellers *et al.* [45] and train the MotifNet model on the One-Shot Visual Genome dataset. Github: https://github.com/rowanz/neural-motifs

Table 5: Experimental results on the Visual Genome dataset. SL denotes supervised learning. RL denotes reinforcement learning.

Mathada	PredCls			SGCIS			SGDet		
Methous	R@20	R@50	R@100	R@20	R@50	R@100	R@20	R@50	R@100
VRD [23]	-	27.9	35.0	-	11.8	14.1	-	0.3	0.5
IMP [41]	-	44.8	53.0	-	21.7	24.4	-	3.4	4.2
IMP+ [41, 45]	52.7	59.3	61.3	31.7	34.6	35.4	14.6	20.7	24.5
TFR [11]	40.1	51.9	58.3	19.6	24.3	26.6	3.4	4.8	6.0
AE [24]	47.9	54.1	55.4	18.2	21.8	22.6	6.5	8.1	8.2
FREQ+OVERLAP [45]	53.6	60.6	62.2	29.3	32.3	32.9	20.1	26.2	30.1
Graph R-CNN [43]	-	54.2	59.1	-	29.6	31.6	-	11.4	13.7
KERN [3]	-	65.8	67.6	-	36.7	37.4	-	27.1	29.8
MotifNet [45]	58.5	65.2	67.1	32.9	35.8	36.5	21.4	27.2	30.3
IRT (Ours)	60.3	66.8	68.5	33.9	36.9	37.5	22.0	27.9	31.1
IRT+MSK (Ours)	60.4	67.0	68.6	34.2	37.1	37.7	22.2	28.0	31.2
TreeLSTM+RLrefine [37]	60.1	66.4	68.1	35.2	38.1	38.8	22.0	27.9	31.3
CMAT+RLrefine [2]	60.2	66.4	68.1	35.9	39.0	39.8	22.1	27.9	31.2
	Methods           VRD [23]           IMP [41]           IMP+ [41, 45]           TFR [11]           AE [24]           FREQ+0VERLAP [45]           Graph R-CNN [43]           KERN [3]           MotifNet [45]           IRT (Ours)           IRT+MSK (Ours)           TreeLSTM+RLrefine [37]           CMAT+RLrefine [2]	Methods         R@20           VRD [23]         -           IMP [41]         -           IMP+ [41, 45]         52.7           TFR [11]         40.1           AE [24]         47.9           FREQ+OVERLAP [45]         53.6           Graph R-CNN [43]         -           KERN [3]         -           MotifNet [45]         58.5           IRT (Ours)         60.3           IRT+MSK (Ours)         60.4           TreeLSTM+RLrefine [37]         60.1           CMAT+RLrefine [2]         60.2	Methods         PredCls           R@20         R@50           VRD [23]         -         27.9           IMP [41]         -         44.8           IMP+ [41, 45]         52.7         59.3           TFR [11]         40.1         51.9           AE [24]         47.9         54.1           FREQ+OVERLAP [45]         53.6         60.6           Graph R-CNN [43]         -         55.2           IRT (Ours)         60.3         66.8           IRT+MSK (Ours)         60.4         67.0           TreeLSTM+RLrefine [37]         60.1         66.4	Methods         PredCls           R@20         R@50         R@100           VRD [23]         -         27.9         35.0           IMP [41]         -         44.8         53.0           IMP [41]         -         44.8         53.0           IMP [41, 45]         52.7         59.3         61.3           TFR [11]         40.1         51.9         58.3           AE [24]         47.9         54.1         55.4           FREQ+OVERLAP [45]         53.6         60.6         62.2           Graph R-CNN [43]         -         54.2         59.1           KERN [3]         -         65.8         67.6           MotifNet [45]         58.5         65.2         67.1           IRT (Ours) <b>60.3 66.8 68.5</b> IRT+MSK (Ours) <b>60.4 67.0 68.6</b> TreeLSTM+RLrefine [37]         60.1         66.4         68.1           CMAT+RLrefine [2]         60.2         66.4         68.1	Methods         PredCls         R@20           R@20         R@50         R@100         R@20           VRD [23]         -         27.9         35.0         -           IMP [41]         -         44.8         53.0         -           IMP [41]         -         44.8         53.0         -           IMP [41]         -         44.8         53.0         -           IMP [41, 45]         52.7         59.3         61.3         31.7           TFR [11]         40.1         51.9         58.3         19.6           AE [24]         47.9         54.1         55.4         18.2           FREQ+OVERLAP [45]         53.6         60.6         62.2         29.3           Graph R-CNN [43]         -         54.2         59.1         -           KERN[3]         -         54.2         59.1         -           KERN[4]         -         58.5         65.2         67.1         32.9           IRT (Ours)         60.3         66.8         68.5         33.9           IRT+MSK (Ours)         60.4         67.0         68.6         34.2           TreeLSTM+RLrefine [37]         60.1         66.4         68	$\begin{tabular}{ c c c c c } \hline & & & & & & & & & & & & & & & & & & $	Methods         PredCls         SGCIS           R@20         R@20         R@100         R@20         R@100           VRD [23]         -         27.9         35.0         -         11.8         14.1           IMP [41]         -         44.8         53.0         -         21.7         24.4           IMP [41, 45]         52.7         59.3         61.3         31.7         34.6         35.4           TFR [11]         40.1         51.9         58.3         19.6         24.3         26.6           AE [24]         47.9         54.1         55.4         18.2         21.8         22.6           FREQ+OVERLAP [45]         53.6         60.6         62.2         29.3         32.9         31.6         31.7           Graph R-CNN [43]         -         55.2         59.1         -         29.6         31.6           KERN [3]         -         65.8         67.6         -         36.7         37.4           MotifNet [45]         58.5         65.2         67.1         32.9         35.8         36.5           IRT (Ours) <b>60.3 66.8 68.5 33.9 36.9 37.7</b>	Methods         Image: PredCls         SGCls         R@100         R@20         R@20 </td <td>Methods         PredCls         SGCls         SGDet           R@20         R@50         R@100         R@20         R@50         R@100         R@20         R@100         R@20         R@100         R@20         R@100         R@20         R@100         R@20         R@100         R@20         R@50           VRD [23]         -         27.9         35.0         -         11.8         14.1         -         0.3           IMP [41]         -         44.8         53.0         -         21.7         24.4         -         3.4           IMP+ [41, 45]         52.7         59.3         61.3         31.7         34.6         35.4         14.6         20.7           TFR [11]         40.1         51.9         58.3         19.6         24.3         26.6         3.4         4.8           AE [24]         47.9         54.1         55.4         18.2         21.8         22.6         66.2         8.1           FREQ+OVERLAP [45]         53.6         60.6         62.2         29.3         32.3         32.9         20.1         26.2           Graph R-CNN [43]         -         54.2         57.1         2.7         27.1         37.4         -</td>	Methods         PredCls         SGCls         SGDet           R@20         R@50         R@100         R@20         R@50         R@100         R@20         R@100         R@20         R@100         R@20         R@100         R@20         R@100         R@20         R@100         R@20         R@50           VRD [23]         -         27.9         35.0         -         11.8         14.1         -         0.3           IMP [41]         -         44.8         53.0         -         21.7         24.4         -         3.4           IMP+ [41, 45]         52.7         59.3         61.3         31.7         34.6         35.4         14.6         20.7           TFR [11]         40.1         51.9         58.3         19.6         24.3         26.6         3.4         4.8           AE [24]         47.9         54.1         55.4         18.2         21.8         22.6         66.2         8.1           FREQ+OVERLAP [45]         53.6         60.6         62.2         29.3         32.3         32.9         20.1         26.2           Graph R-CNN [43]         -         54.2         57.1         2.7         27.1         37.4         -



Figure 4: The visualization results of Instance Relation Transformer (VGG)+MSK on the one-shot scene graph generation task. These results are generated on the PredCls setup.

## 5 CONCLUSION

In this paper, to equip the model with the ability to learn the visual relationship from one labeled sample, we design a novel task, namely one-shot scene graph generation. Motivated by the way humans learn visual relationships, the Multiple Structured Knowledge (Relational Knowledge and Commonsense Knowledge) is introduced into the one-shot scene graph generation task. The Relational Knowledge extracted from Visual Genome represents the prior knowledge of relationships among entities in the visual space. The Commonsense Knowledge explores "sense-making" knowledge from ConceptNet. Besides, we propose the Instance Relation Transformer for capturing the relational context among instances. Detailed experiments validate the effectiveness of the Instance Relation Transformer and the Multiple Structured Knowledge.

## ACKNOWLEDGMENTS

This work is supported by the Fundamental Research Funds for the Central Universities (Grant No. ZYGX2019J073), the National Natural Science Foundation of China (Grant No. 61772116, No. 61872064, No.61632007, No. 61602049), Sichuan Science and Technology Program (Grant No. 2019JDTD0005), The Open Project of Zhejiang Lab (Grant No. 2019KD0AB05).

# REFERENCES

- Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. [n.d.]. Translating Embeddings for Modeling Multi-relational Data. In *NeurIPS*. 2787–2795.
- [2] Long Chen, Hanwang Zhang, Jun Xiao, Xiangnan He, Shiliang Pu, and Shih-Fu Chang. 2019. Counterfactual Critic Multi-Agent Training for Scene Graph Generation. In *ICCV*. 4612–4622.
- [3] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. 2019. Knowledge-Embedded Routing Network for Scene Graph Generation. In CVPR.
- [4] Chuang Gan, Deng Huang, Peihao Chen, Joshua B Tenenbaum, and Antonio Torralba. 2020. Foley Music: Learning to Generate Music from Videos. ECCV (2020).

- [5] Chuang Gan, Deng Huang, Hang Zhao, Joshua B Tenenbaum, and Antonio Torralba. 2020. Music Gesture for Visual Sound Separation. In CVPR. 10478– 10487.
- [6] Lianli Gao, Pengpeng Zeng, Jingkuan Song, Xianglong Liu, and Heng Tao Shen. 2018. Examine before you answer: Multi-task learning with adaptive-attentions for multiple-choice VQA. In *Proceedings of the 26th ACM international conference* on Multimedia. 1742–1750.
- [7] Ross B. Girshick. 2015. Fast R-CNN. In ICCV. 1440-1448.
- [8] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. 2019. Scene Graph Generation With External Knowledge and Image Reconstruction. In CVPR. 1969–1978.
- [9] Yuyu Guo, Jingqiu Zhang, and Lianli Gao. 2019. Exploiting long-term temporal dynamics for video captioning. World Wide Web 22, 2 (2019), 735–749.
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In CVPR. 770–778.
- [11] Seong Jae Hwang, Sathya N. Ravi, Zirui Tao, Hyunwoo J. Kim, Maxwell D. Collins, and Vikas Singh. 2018. Tensorize, Factorize and Regularize: Robust Visual Relationship Learning. In CVPR. 1014–1023.
- [12] Y. Ji, Y. Zhan, Y. Yang, X. Xu, F. Shen, and H. T. Shen. 2020. A Context Knowledge Map Guided Coarse-to-Fine Action Recognition. *IEEE Transactions on Image Processing* 29 (2020), 2742–2752.
- [13] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. 2015. Image retrieval using scene graphs. In *CVPR*. IEEE Computer Society, 3668–3678.
- [14] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In ICLR.
- [15] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. 2015. Siamese neural networks for one-shot image recognition. (2015).
- [16] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2017. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision* 123, 1 (2017), 32–73.
- [17] Fei-Fei Li, Robert Fergus, and Pietro Perona. 2006. One-Shot Learning of Object Categories. IEEE Trans. Pattern Anal. Mach. Intell. 28, 4 (2006), 594-611.
- [18] Xiangpeng Li, Lianli Gao, Xuanhan Wang, Wu Liu, Xing Xu, Heng Tao Shen, and Jingkuan Song. 2019. Learnable aggregating net with diversity learning for video question answering. In *Proceedings of the 27th ACM International Conference on Multimedia*. 1166–1174.
- [19] Xiangpeng Li, Jingkuan Song, Lianli Gao, Xianglong Liu, Wenbing Huang, Xiangnan He, and Chuang Gan. 2019. Beyond rnns: Positional self-attention with co-attention for video question answering. In AAAI, Vol. 33. 8658–8665.
- [20] Yikang Li, Wanli Ouyang, Bolei Zhou, Jianping Shi, Chao Zhang, and Xiaogang Wang. 2018. Factorizable Net: An Efficient Subgraph-Based Framework for Scene Graph Generation. In ECCV. 346–363.
- [21] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. 2017. Scene Graph Generation from Objects, Phrases and Region Captions. In *ICCV*. 1270–1279.
- [22] Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. KagNet: Knowledge-Aware Graph Networks for Commonsense Reasoning. In *EMNLP-IJCNLP*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). 2829–2839.
- [23] Cewu Lu, Ranjay Krishna, Michael S. Bernstein, and Fei-Fei Li. 2016. Visual Relationship Detection with Language Priors. In ECCV. 852–869.
- [24] Alejandro Newell and Jia Deng. 2017. Pixels to Graphs by Associative Embedding. In NeurIPS. 2168–2177.
- [25] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. 2016. You Only Look Once: Unified, Real-Time Object Detection. In CVPR. 779–788.
- [26] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans.*

Pattern Anal. Mach. Intell. 39, 6 (2017), 1137-1149.

- [27] Ruslan Salakhutdinov, Joshua B. Tenenbaum, and Antonio Torralba. 2012. One-Shot Learning with a Hierarchical Nonparametric Bayesian Model. In Unsupervised and Transfer Learning - Workshop held at ICML. 195–206.
- [28] F. Shen, C. Shen, Q. Shi, A. van den Hengel, Z. Tang, and H. T. Shen. 2015. Hashing on Nonlinear Manifolds. *IEEE Transactions on Image Processing* 24, 6 (2015), 1839–1851.
- [29] Heng Tao Shen, Luchen Liu, Yang Yang, Xing Xu, Zi Huang, Fumin Shen, and Richang Hong. 2020. Exploiting subspace relation in semantic labels for crossmodal hashing. *IEEE Transactions on Knowledge and Data Engineering* (2020).
- [30] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. (2015).
- [31] Jingkuan Song, Yuyu Guo, Lianli Gao, Xuelong Li, Alan Hanjalic, and Heng Tao Shen. 2018. From deterministic to generative: Multimodal stochastic RNNs for video captioning. *IEEE transactions on neural networks and learning systems* 30, 10 (2018), 3047–3058.
- [32] Jingkuan Song, Zhao Guo, Lianli Gao, Wu Liu, Dongxiang Zhang, and Heng Tao Shen. 2017. Hierarchical LSTM with adjusted temporal attention for video captioning. arXiv preprint arXiv:1706.01231 (2017).
  [33] Jingkuan Song, Pengpeng Zeng, Lianli Gao, and Heng Tao Shen. 2018. From
- [33] Jingkuan Song, Péngpeng Zeng, Lianli Gao, and Heng Tao Shen. 2018. From Pixels to Objects: Cubic Visual Attention for Visual Question Answering.. In IJCAI. 906–912.
- [34] Jingkuan Song, Hanwang Zhang, Xiangpeng Li, Lianli Gao, Meng Wang, and Richang Hong. 2018. Self-supervised video hashing with hierarchical binary auto-encoder. *IEEE Transactions on Image Processing* 27, 7 (2018), 3210–3221.
- [35] Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. In AAAI, Satinder P. Singh and Shaul Markovitch (Eds.). 4444–4451.
- [36] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In CVPR. 1–9.
- [37] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. 2019. Learning to Compose Dynamic Tree Structures for Visual Contexts. In CVPR. 6619–6628.
- [38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NeurIPS*. 5998–6008.
- [39] Peng Wang, Lingqiao Liu, Chunhua Shen, Zi Huang, Anton van den Hengel, and Heng Tao Shen. 2017. Multi-attention Network for One Shot Learning. In CVPR. 6212–6220.
- [40] Jiwei Wei, Xing Xu, Yang Yang, Yanli Ji, Zheng Wang, and Heng Tao Shen. 2020. Universal Weighting Metric Learning for Cross-Modal Matching. In CVPR. 13005– 13014.
- [41] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. 2017. Scene Graph Generation by Iterative Message Passing. In CVPR. 3097–3106.
- [42] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In *ICLR*, Yoshua Bengio and Yann LeCun (Eds.).
- [43] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. 2018. Graph R-CNN for Scene Graph Generation. In ECCV. 690–706.
- [44] Guojun Yin, Lu Sheng, Bin Liu, Nenghai Yu, Xiaogang Wang, Jing Shao, and Chen Change Loy. 2018. Zoom-Net: Mining Deep Feature Interactions for Visual Relationship Recognition. In *ECCV*. 330–347.
- [45] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural Motifs: Scene Graph Parsing With Global Context. In CVPR. 5831–5840.
- [46] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. 2017. Visual Translation Embedding Network for Visual Relation Detection. In CVPR. 3107– 3115.