Fast Contextual Scene Graph Generation with Unbiased Context Augmentation

Tianlei Jin[1] Fangtai Guo[1] Qiwei Meng[1] Shiqiang Zhu[1] Xiangming Xi[1]
Wen Wang[1] Zonghao Mu[1] Wei Song[1*]
[1]Research Center for Intelligent Robotics, Zhejiang Lab
{jtl,guofangtai,mengqw,zhusq,xxm21,wangwen,muzonghao,weisong}@zhejianglab.com

Abstract

Scene graph generation (SGG) methods have historically suffered from long-tail bias and slow inference speed. In this paper, we notice that humans can analyze relationships between objects relying solely on context descriptions, and this abstract cognitive process may be guided by experience. For example, given descriptions of cup *and* $table$ *with their spatial locations, humans can speculate possible relationships* $< cup, on, table >$ *or* $< table, near, cup >$*. Even without visual appearance information, some impossible predicates like* $flying\ in$ *and* $looking\ at$ *can be empirically excluded. Accordingly, we propose a contextual scene graph generation (C-SGG) method without using visual information and introduce a context augmentation method. We propose that slight perturbations in the position and size of objects do not essentially affect the relationship between objects. Therefore, at the context level, we can produce diverse context descriptions by using a context augmentation method based on the original dataset. These diverse context descriptions can be used for unbiased training of C-SGG to alleviate long-tail bias. In addition, we also introduce a context guided visual scene graph generation (CV-SGG) method, which leverages the C-SGG experience to guide vision to focus on possible predicates. Through extensive experiments on the publicly available dataset, C-SGG alleviates long-tail bias and omits the huge computation of visual feature extraction to realize real-time SGG. CV-SGG achieves a great trade-off between common predicates and tail predicates.*

1. Introduction

SGG is a challenging technology that identifies triplet relationships $< subject, predicate, object >$ between objects from images. With the development of artificial intelligence, SGG has gradually become a bridge from image recognition to image understanding. Scene graphs are an indispensable part of complex visual understanding tasks, such as visual question answering [24], visual grounding [21] and visual-language navigation [40]. However, the researches [3, 19, 36]on SGG suffer from two insurmountable obstacles. The first is the long-tail bias derived from datasets. More common predicates such as *on* and *near* have more samples than tail predicates such as *from* and *above*, causing the model to prefer to classify the common predicates. The second is the low-speed inference in practical applications. Analyzing the predicate between each objects-pair to generate a scene graph is a quadratic time complexity problem, making real-time inference difficult.

On the one hand, some SGG methods [3, 26, 31, 34] are dedicated to solving the long-tail bias. Tang [26] and Chiou [2] introduce the causal graph and the label frequency to reason tail predicates and attempt unbiased SGG inference based on biased training. Li [11] and Desai [3] propose to optimize label distribution and rebalance category sampling, which realize unbiased SGG training. However, these methods increase the recall of tail predicates, but inevitably reduce the recall of common predicates. The current SGG methods are difficult to consider and balance the recall of common predicates and tail predicates simultaneously. We think the internal reason is that there are not enough data samples for each predicate.

On the other hand, some SGG methods [18, 33] focus on improving the inference speed of the SGG task. In detail, Yang [33] designs all objects in a fully connected graph structure and prunes the connections between objects. It can reduce the time complexity of SGG inference. Liu [18] transforms the predicate inference into an integral on relationship affinity fields. Although the time complexity is not reduced, the computation amount of integral operation is much less than that of deep learning calculation of visual features. These methods improve the inference speed, but sacrifice the recall performance.

We reflect on the human cognitive process of predicate analysis between objects and discover two overlooked phenomena. First, humans can roughly infer the predicate between objects based on the context descriptions only includ-

*Corresponding Author

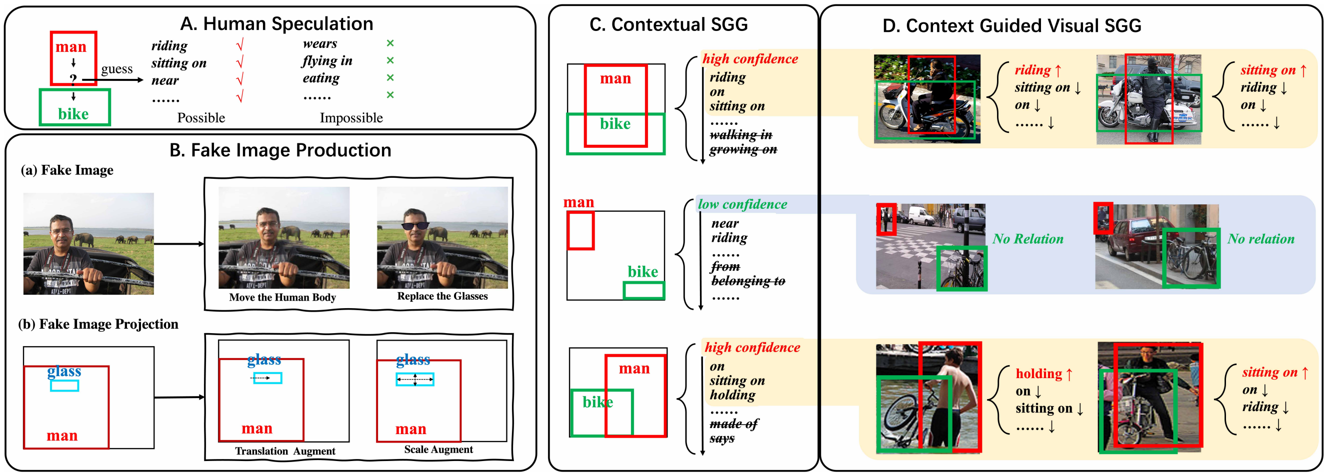

Figure 1. A. The example of human speculate predicates based on the context description. B. (a) Use software to change objects in the image to produce fake images; (b) Project object pairs in the fake image to the context level. C. Examples of C-SGG outputs in different context descriptions. D. Examples of CV-SGG to further analyze high-confidence relationships and possible predicates.

ing categories and positions. In other words, humans can speculate and analyze possible relationships through context descriptions even without seeing objects. As shown in Fig.1 A, when humans know the subject man and the object $bike$, they can speculate and analyze which predicates are possible ('*riding*', '*sitting on*' , '*near*') and which predicates are impossible ('*wears*', '*flying in*', '*eating*') based on past experience. Second, when humans analyze the predicate between the objects-pair, the apparent features of the objects themself are not important. As shown in Fig.1 B (a), we use adobe photoshop software to move the human body and replace the style of the glasses, but the relationship $<man, wears, glass>$ remains the same. Therefore, we argue that context features may be more important than visual features in rough predicates judgment.

Based on this thought, we weaken the role of vision in the SGG task and propose a contextual SGG (C-SGG) method with context augmentation. As we did in Fig.1 B (a), software such as photoshop can be used to modify the image by moving the position and replacing the like objects without changing the predicate. Different from traditional image augmentation, HSV variation and size scale changing the entire image, this kind of image modification will change the shape and position of a certain object. However, using software to modify images is an extremely complex task. We project the modified image to the context level, as shown in Fig.1 B (b), which is the slight translation and scale of the object position with a cheap cost, and we named it context augmentation. In our C-SGG method, we only use context descriptions to predict predicates, and context augmentation can increase context description samples of any tail predicate for unbiased training. To some extent, through the context augmentation, during our training for C-SGG, there are no two identical context descriptions. In addition, since there are no visual image features, we do not need complex computational models. Although it is still a quadratic time complexity task, the computation amount per object pair is extremely cheap.

Certainly, the C-SGG lacks the analysis of the visual interaction information between objects. We also propose a context guided visual SGG method (CV-SGG) to confirm truth predicates between object pairs further. As shown in Fig.1 C, C-SGG can roughly analyze the confidence in the existence of relationships between objects-pairs and the possible types of predicates. Our CV-SGG focuses on those high-confidence relationships and the high possible predicates. We use a simple visual model to extract visual features and fuse them with contextual features. During the training, we apply a ReLuL1 function and only calculate the loss on high possible predicates. In this way, CV-SGG only pays attention to possible predicates from C-SGG and ignores impossible predicates. As shown in Fig.1 D, context guided visual SGG is used to boost the truth predicate and suppress other possible but false predicates.

We validate our methods on the most common SGG dataset VG [10] and the latest SGG dataset PSG [32]. Our methods achieve the best balance between common predicates and tail predicates, and accomplish real-time SGG. The contributions of this paper can be summarized as:

1) Inspired by the human cognitive process, we propose context augmentation to produce diverse context descriptions at the context level for unbiased training, which weakens the role of vision.

2) We propose two methods for SGG: C-SGG which only uses context descriptions and CV-SGG which guides vi-

sual attention based on C-SGG results.

3) Based on extensive experiments on two SGG datasets VG and PSG, our methods have obvious advantages in dealing with long-tail bias and inference speed.

2. Related Works

Traditional research about SGG is also called visual relationship detection. VRD [19] first proposes the SGG task based on visual object proposals from RCNN [4, 25]. Researchers have gradually realized the importance of SGG in image understanding, and many subsequent works including IMP [30], Motifs [36], VCTree [27] follow this task. These works respectively introduce message passing structures, such as IMP [30], MSDN [14], GPS-Net [17], GB-Net [35], CISC [29], tree structures including VC-Tree [27] and CogTree [34], graph structures including G-RCNN [33], KERN [1] and GCN-SGG [39]. Pixels2Graphs [22] and FCSGG [18] directly predict object pairs and relationships from images, without relying on RCNN results. Seq2Seq-RL [20] introduces using the global context and the seq2seq transformer to estimate the scene graph. SS-RCNN [28] achieves one-stage SGG through triple query based on Sparse R-CNN. OpenPSG [32] combines the panoptic segmentation and the SGG, and uses the transformer structure to simultaneously predict panoptic masks and relationships. However, in these methods, visual features always play a dominant role in SGG and context features are often used as auxiliary information. For example, RelDN [38] predicts the predicate in the spatial, semantics and visual three channels respectively, and designs the contrastive losses. GPS-Net [17] concats visual features, class scores and spatial features as node features and predicts predicates between nodes based on node features. Motifs [36] has proposed to use the global bounding boxes and labels for edge prediction, but global context information cannot effectively deal with long-tail bias, and the inference speed of bidirectional LSTM is slow. In our methods, we only use local context and visual features are discarded. Our methods extract object pairs for contextual augmentation training, and uses the results of the contextual scene graph results to guide visual SGG.

In recent years, due to the extreme imbalance of predicate categories in the SGG dataset, some works have focused on the long-tail bias to improve the performance of tail predicate predictions. These works can be divided according to whether the training is biased or not. For biased SGG training, extra information is often learned to help remove bias during inference. TDE [26] proposes the causal graph and tries to make the model recognize the deep mean of object features. Cogtree [34] proposes a coarse-to-fine method and debris from biased predictions, while BPL-SA [6] introduces the confusion matrix. DLFE [2] proposes the label frequency estimation and learns the label frequencies in biased training to remove reporting bias.

For unbiased SGG training, additional data processing steps help the model to train unbiased. PCPL [31] proposes the predicate correlation and enables the model to distinguish similar predicates, such as '*on*' and '*parked on*'. GFAL [9] introduces the graph density-aware losses for unbiased training. DT2-ACBS [3] introduces rebalanced sampling strategy and discusses the impact of different sampling strategy on the SGG task. NICE [11] analyzes the samples in the dataset to optimize more accurate labels and generate pseudo-labels that are not labeled. IETrans [37] proposes internal transfer and external transfer to enhance SGG dataset. BGNN [13] introduces a bipartite graph network with bi-level data sampling that can account for the overall recall and the mean recall of predicates. We believe small changes in objects for producing fake images do not change predicates between objects, so we project fake images to the context level to increase the number of context samples. Then we can obtain diverse context samples for unbiased training of the contextual SGG.

3. Method

Notation. Given an SGG dataset χ, we donate its corresponding images I, bounding box locations B, objects O, and relationships R. For SGG, giving an image I_i, we can get a graph G_i, which is made up of a set of bounding box locations $B_i = \{b_{i1}, b_{i2}, ...b_{in}\}, b_{ij} \in \mathbb{R}^4$, objects $O_i = \{o_{i1}, o_{i2}, ...o_{in}\}$, relationships $R_i = \{r_{i1}, r_{i2}, ..., r_{im}\}$. Therefore, the task of SGG can be expressed as:

$$Pr(B, O, R|I) = Pr(B, O|I)Pr(R|B, O, I), \quad (1)$$

Following previous works [35, 36, 38], $Pr(B, O|I)$ is always realized with the help of object detection methods [25], and the SGG task pays more attention to relationships generation $Pr(R|B, O, I)$.

3.1. Contextual SGG

In our C-SGG, contextual relationships are learned only from context descriptions. The context descriptions include objects O and bounding box locations B, and the learned possible predicate knowledge, which we donate R^c. The process of SGG from the context descriptions can be expressed as follows:

$$Pr(R^c|B, O). \quad (2)$$

Before C-SGG training, we preprocess the context to augment the context description. Traditional image augmentation enriches the color and size of the entire image, but the size and relative position of objects stay unchanged. For the prediction of predicates between objects, we believe that the apparent features of the image are not important, but

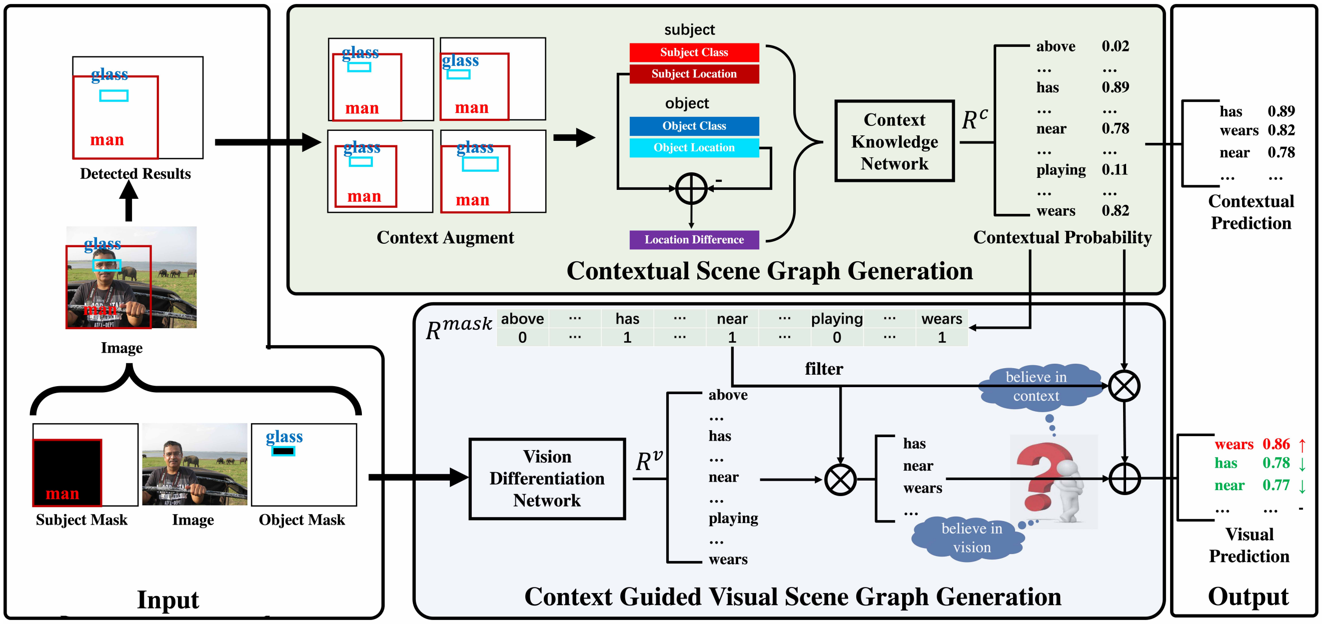

Figure 2. Illustration of our C-SGG and CV-SGG methods. We employ other object detection models to obtain categories and bounding boxes as the context description. For C-SGG, our context augmentation method is used to generate diverse context descriptions and these context descriptions are input into the simple CKN network to estimate possible predicates. For CV-SGG, the image with masks are input into the VDN network, then the contextual mask R^{mask} guide the VDN focus on those possible predicates.

the size of the objects themselves and the positional relationship between objects are more critical. For example, in Fig.2, the relationship is $<man, wearing, glasses>$. The body of the man may be tall or short, fat or thin. The style of the $glasses$ may be large or small, and the location of the $glasses$ may move with the head. However, the predicate $wearing$ between the man and $glasses$ has never changed. Therefore, we attempt to produce fake images by changing the position of the object and replacing the style of the object, but it is extremely labor-intensive. At the context level, the process of producing fake images can be viewed as perturbing the position of the bounding box of objects with a cheap cost.

As shown in Fig.2, we obtain the category and bounding box of the object in the image through the common object detection algorithm. For the j object in the i image, the normalized location can be represented as: $b_{ij} = [x1_{ij}, y1_{ij}, x2_{ij}, y2_{ij}]$. Then we add random context augmentations to the position, denoted as $\widetilde{b_{ij}} = [\widetilde{x1}_{ij}, \widetilde{y1}_{ij}, \widetilde{x2}_{ij}, \widetilde{y2}_{ij}] = [x1_{ij} + \varepsilon_1, y1_{ij} + \varepsilon_2, x2_{ij} + \varepsilon_3, y2_{ij} + \varepsilon_4]$, ε is random augment factor. For the category of the object o_{ij}, we use the glove word2vector model [23] to convert the category of object o_{ij} into semantic word vector $\vec{o}_{ij}, \vec{o}_{ij} \in \mathbb{R}^{50}$. The location vector $\vec{b}_{ij}$ consists of $\widetilde{b}_{ij}$, $\vec{b}_{ij} = [\widetilde{x1}_{ij}, \widetilde{y1}_{ij}, \widetilde{x2}_{ij}, \widetilde{y2}_{ij}, \widetilde{xc}_{ij}, \widetilde{yc}_{ij}] \times 5, \vec{b}_{ij} \in \mathbb{R}^{30}$. $\widetilde{xc}_{ij}$ and $\widetilde{yc}_{ij}$ are the center of the bounding box, and we repeat the location by 5 times to enhance location features. For the two objects j_1 and j_2, the final vector of context description can be expressed:

$$\vec{D} = [\vec{o}_{ij_1}, \vec{b}_{ij_1}, \vec{o}_{ij_2}, \vec{b}_{ij_2}, \vec{b}_{ij_1} - \vec{b}_{ij_2}], \vec{D} \in \mathbb{R}^{190}. \quad (3)$$

We construct a simple and effective context knowledge network (CKN) to generate possible contextual predicates based on context description vectors $\vec{D}$. In detail, we use three fully connected network layers with a sigmoid layer. The output dimension of the CKN corresponds to the number of predicates in the dataset. The loss consists of two parts, the confidence loss L^{ckn}_{conf} and the predicate loss L^{ckn}_{rel}. Both two loss are calculated by Binary Cross Entropy (BCE) function. The CKN predicts the likelihood and possible predicates of relationships between two contextually described objects. In this way, based on raw data samples in the dataset, we generate diverse context descriptions for each predicate through random context augmentations, and achieve C-SGG through CKN without vision.

3.2. Context Guided Visual SGG

We are able to pick out possible predicates through the C-SGG, but since no visual information is used, the prediction is empirical. We further propose a CV-SGG, combining visual and contextual. The process of learning relationships from CV-SGG can be expressed as:

$$Pr(R^v | B, O, I, R^c). \quad (4)$$

Based on object detection results, we can get the location of objects. We make the $subject$ mask and the $object$ mask according to the location of the object pair. We compress the $subject$ mask, the original image and the $object$ mask together to form visual pair information, and feed it to the vision differentiation network (VDN). The VDN is constructed by a ResNet [7] for extracting visual features, followed by a flattened layer and a fully connected layer with a sigmoid for predicate prediction.

From C-SGG, the CKN predicts the confidence scores and possible predicates. We expect that VDN can focus on possible predicates, ignore impossible predicates (e.g. $< human, above, glass >$), and differentiate the truth predicate relationship R^v based on vision. Based on R^c, we generate an R^{mask} for the most possible N_{mask} predicates. Then we design a ReLuL1 loss including L^{vdn}_{boost} and $L^{vdn}_{suppress}$ to boost or suppress R^v.

$$L^{vdn}_{boost} = ReLu(R^c_{p=p^t} - R^v_{p=p^t} + \eta), \tag{5}$$

$$L^{vdn}_{suppress} = ReLu(R^v_{p\neq p^t} - R^c_{p\neq p^t} + \eta) \times R^{mask}, \tag{6}$$

For the truth predicate p^t in eq.5, we suppose that visual understanding R^v can further boost contextual probability R^c. For the false predicate in eq.6, we suppose the visual understanding R^v can suppress contextual probability R^c, and only high possible predicates based on contextual mask R^{mask} will be calculated, η is a boost factor. For example, in Fig.2, the model learns from the C-SGG that has, $near$, $wears$ are high possible predicates under the current context description which can generate a R^{mask}. During CV-SGG, the visual information only focuses on and analyzes these possible predicates. Just like analysis pattern of human beings, relationships that are beyond the scope of empirical cognition are not considered.

During the final inference, context and vision are both considered:

$$R = (\alpha R^c + (1 - \alpha)R^v) \times R^{mask}, \tag{7}$$

Where α is an empirical factor. The larger α is, the model more believes in the inherent context experience. The smaller α is, the model more believes in the visual analysis. Similarly, only high possible predicates can be imagined in inference through R^{mask}.

3.3. Implementation detail

For C-SGG, we perform context description augments during training. We set the context augmentation factor ε below 0.05. We also adopt a similar alternating class balanced sampling [3] strategy to make the samples of each predicate as equal as possible, the difference is that our samples are enhanced by context descriptions. Even for the same sample, the context description of the input model after context augmentation is different. We trained it on an RTX2070 SUPER with 256 batch size, which only takes up 1.8G GPU memory without visual information. The epoch is 2000 for 8 hours of training. The initial learning rate is set to 0.04 and drops during training.

For CV-SGG, the inputs size of VDN are resized to 224×224×5, including two masks and an image. As for the R^{mask}, we count the output of C-SGG and find that in the test samples of the VG dataset, the probability of the truth predicate being included in the top 3, 5, and 10 possibilities is 89%, 95%, and 98%, respectively, so we set the $N_{mask} = 10$. The boost factor η is set to 0.1, and the empirical factor α is set to 0.7 for balance context experience and vision analysis. We trained it on an RTX3090Ti with 64 batch size. The epoch is 100 for 60 hours of training. The initial learning rate is set to 0.002 and drops during training.

4. Experiments

4.1. Dataset and Metrics

We train and evaluate our method on the challenging SGG dataset VG [10,26]. VG contains approximately 108k images, with 70% for training and 30% for testing from the Visual Genome dataset [10]. The relationships include the most frequent 150 object categories and 50 predicate categories. In total, the number of original object pair context descriptions in the VG training set is 342,363. There are 101,843 and 54,317 samples for the common predicates on and has, while only 121 and 260 samples for the tail predicates $playing$ and $across$. The task requires outputting the results of object detection and the scene graph.

We also evaluate our method on the latest SGG dataset PSG [32]. PSG contains 46697 images for training, and 1989 images for validation and testing from the COCO dataset [16]. Each image has a corresponding panoptic segmentation label. For relationships, it includes 133 objects (i.e., things plus stuff) and 56 predicates with appropriate granularity and minimal overlaps. The number of original object pair context descriptions in the PSG training set is 261,666. There are 52,974 and 45,032 samples for the most common predicates on and $beside$, while only 7 and 8 samples for the tail predicates $falling\ off$ and $picking$. The task requires outputting the results of panoptic segmentation and the scene graph.

This paper focuses on the scene graph. We evaluate our method on two standard SGG tasks: Predicate Classification (PredCls) and Scene Graph Generation (SGGen). For PredCls, given the ground-truth objects O^t and locations B^t (or panoptic segmentation mask M^t), we only need to predict the predicate category of relationships, $P(R|B^t(M^t), O^t, I)$. For SGGen, only given the image I, we need to generate the scene graph, $P(B(M), O, R|I)$.

The metrics of SGG including Recall@K (**R@K**) [19], mean Recall@K (**mR@K**) [27], **Mean@K** [11], **F@K**

Method	PredCls				SGGen			
	R@50/100	mR@50/100	Mean@50/100	F@50/100	R@50/100	mR@50/100	Mean@50/100	F@50/100
IMP [30]CVPR'2017	61.1/63.1	11.0/11.8	36.1/37.4	18.6/19.9	25.9/31.2	4.2/5.3	15.1/18.3	7.2/9.1
FREQ [36]CVPR'2018	60.6/62.2	13.0/16.0	36.8/39.1	21.4/25.5	26.2/30.1	6.1/7.1	16.2/18.6	9.9/11.5
G-RCNN [33]ECCV'2018	64.8/66.7	16.4/17.2	40.6/42.0	26.2/27.4	29.7/32.8	5.8/6.6	17.8/19.7	9.7/11.0
KERN [1]CVPR'2019	65.8/67.6	17.7/19.2	41.2/43.4	27.9/29.9	27.1/29.8	6.4/7.3	16.8/18.6	10.4/11.7
GB-NET [35]ECCV'2020	**66.6/68.2**	22.1/24.0	44.4/46.1	33.2/35.5	26.3/29.9	7.1/8.5	16.7/19.2	11.2/13.2
BGNN [13]CVPR'2021	59.2/61.3	30.4/32.9	**44.8/47.1**	40.2/42.8	31.0/35.8	10.7/12.6	20.9/24.2	15.9/18.6
DT2-ACBS [3]ICCV'2021	23.3/25.6	**35.9/39.7**	29.6/32.7	28.3/31.1	15.0/16.3	**22.0/24.4**	18.5/20.4	**17.8**/19.5
PCPL [31]ACM MM'2021	50.8/52.6	35.2/37.8	43.0/45.2	**41.6/44.0**	14.6/18.6	9.5/11.7	12.1/15.2	11.5/14.4
FCSGG [18]CVPR'2021	41.0/45.0	6.3/7.1	23.7/26.1	10.9/12.3	21.3/25.1	3.6/4.2	12.4/14.7	6.0/7.2
SGTR [12]CVPR'2022	-	-	-	-	24.6/28.4	12.0/15.2	18.3/21.8	16.1/**19.8**
SS-RCNN [28]CVPR'2022	-	-	-	-	**33.5/38.4**	8.6/10.3	**21.0/24.4**	13.7/16.2
Motifs [36]CVPR'2018	**65.5/67.2**	16.5/17.8	41.0/42.5	26.4/28.1	**32.1/36.9**	5.5/6.8	18.8/21.9	9.4/11.5
+TDE [26]CVPR'2020	46.2/51.4	25.5/29.1	35.9/40.3	32.9/37.2	16.9/20.3	8.2/9.8	12.6/15.1	11.0/13.2
+CogTree [34]IJCAI'2021	35.6/36.8	26.4/29.0	31.0/32.9	30.3/32.4	20.0/22.1	10.4/11.8	15.2/16.9	13.7/15.4
+DLFE [2]ACM MM'2021	52.5/54.2	26.9/28.8	39.7/41.5	35.6/37.6	25.4/29.4	11.7/13.8	18.6/21.6	16.0/18.8
+BPL-SA [6]ICCV'2021	50.7/52.5	29.7/31.7	40.2/42.1	37.5/39.5	23.0/26.9	13.5/15.6	18.3/21.3	17.0/19.8
+NICE [11]CVPR'2022	55.1/57.2	29.9/32.3	**42.5/44.8**	38.7/41.3	27.8/31.8	12.2/14.4	**20.0/23.1**	17.0/19.8
+IETrans [37]ECCV'2022	48.6/50.5	**35.8/39.1**	42.2/44.8	**41.2/44.1**	23.5/27.2	**15.5/18.0**	19.5/22.6	**18.7/21.7**
VCTree [27]CVPR'2019	**65.9/67.5**	17.1/18.4	41.5/43.0	27.2/28.9	**32.0/36.2**	7.2 / 8.4	19.6/22.3	11.8/13.6
+TDE [26]CVPR'2020	47.2/51.6	25.4/28.7	36.3/40.2	33.0/36.9	19.4/23.2	9.3/11.1	14.4/17.2	12.6/15.0
+CogTree [34]IJCAI'2021	44.0/45.4	27.6/29.7	35.8/37.6	33.9/35.9	18.2/20.4	10.4/12.1	14.3/16.3	13.2/15.2
+DLFE [2]ACM MM'2021	51.8/53.5	25.3/27.1	38.6/40.2	34.0/35.9	22.7/26.3	11.8/13.8	17.3/20.1	15.5/18.1
+BPL-SA [6]ICCV'2021	50.0/51.8	30.6/32.6	40.3/42.2	38.0/40.0	21.7/25.5	**13.5/15.7**	17.6/20.6	**16.6/19.4**
+NICE [11]CVPR'2022	55.0/56.9	30.7/33.0	**42.9/45.0**	39.4/41.8	27.0/30.8	11.9/14.1	**19.5/22.5**	16.5/19.3
+IETrans [37]ECCV'2022	48.0/49.9	**37.0/39.7**	42.5/44.8	**41.8/44.2**	23.6/27.8	12.0/14.9	17.4/21.4	15.2/19.4
C-SGG(Ours)	55.2/59.2	32.9/36.0	44.1/47.6	41.2/44.7	26.7/30.5	14.8/17.1	20.8/23.8	19.0/21.9
CV-SGG(Ours)	58.2/62.4	32.6/36.2	**45.4/49.3**	**41.8/45.8**	27.8/32.0	14.6/17.0	**21.2/24.5**	**19.2/22.2**

Table 1. Comparison results of SOTA SGG methods on the VG dataset. The excellent result of each group has been marked in blue, while the best result is marked in red.

[37]. R@K calculates the proportion of top-K confident triplets contained in the ground truth, and each triplet only counts the highest score predicate. mR@K calculates the R@K for each predicate category separately. Mean@K is the arithmetic average of R@K and mR@K. F@K is the harmonic average of R@K and mR@K. Since there are more common predicate samples in the dataset, R@K is more suitable for evaluating the recall of common predicates. While a few tail predicate samples lead to dramatic influence on mR@K, mR@K is more concerned with the recall of tail predicates. Mean@K and F@K are proposed to analyze the balance performance of R@K and mR@K. For fairness, we use the same evaluation system as TDE [26].

4.2. Evaluation on VG dataset

We first compare our results with the previous SOTA methods on the VG dataset in Table 1. We roughly divide previous methods into 3 groups. The first group methods do not depend on the previous SGG methods, the second group methods are modifications based on Motifs, and the third group methods are improvements based on VCTree.

We pay more attention to achieving the balance between recall R@K and mean recall mR@K, and want to optimize long-tail bias while maintaining a high overall recall. So Mean@K and F@K metrics are more critical. The PredCls task only focuses on predicates based on known objects. Although our method is not the best in the metrics of R@K and mR@K, our method can find the optimal balance and achieve the SOTA result on Mean@K and F@K. Our C-SGG method has achieved excellent performance without using vision, while CV-SGG method has further improved the R@K without reducing the mR@K by using visual information. FREQ from [36] is a method of generating relationships by statistical frequency without visual information. Compared with it, our C-SGG method has obvious advantages. The SGGen task needs to detect objects and generate predicates, forming triple relationships. Our method still achieves SOTA results on balanced metrics Mean@K and F@K. Our method is based on local context for reasoning, and it can flexibly combine different object detection models to achieve the SGGen task.

For the SGG task, we verify that the apparent features of objects are not important, and the contextual description of object categories and locations is sufficient to infer predicates between objects. In Figure 3, we manifest the recall for each predicate on the SGGen task by the results from our CV-SGG and Motifs. From the trend, due to the long-tail bias, as the occurrences of the predicate category in the training set decrease, the corresponding recall in the test set decreases. Motifs with biased training performs bet-

Method	PQ	PredCls				SGGen			
		R@20/100	mR@20/100	Mean@20/100	F@20/100	R@20/100	mR@20/100	Mean@20/100	F@20/100
IMP [30]CVPR'2017	40.2	31.9/38.9	9.55/11.6	20.7/25.3	14.7/17.9	16.5/18.6	6.52/7.23	11.5/12.9	9.3/10.4
MOTIFS [36]CVPR'2018	40.2	44.9/52.4	20.2/22.9	32.6/37.7	27.9/31.9	20.0/22.0	9.10/9.69	14.6/15.8	12.5/13.5
VCTree [27]CVPR'2019	40.2	**45.3/52.7**	20.5/23.3	32.9/38.0	28.2/32.9	**20.6/22.5**	9.70/10.2	15.2/16.4	13.1/14.0
GPSNet [17]CVPR'2020	40.2	31.5/44.7	13.2/18.4	22.4/31.5	18.6/26.0	17.8/20.1	7.03/7.67	12.4/13.9	10.1/11.1
C-SGG(Ours)	40.2	36.5/46.5	**32.5/36.4**	**34.5/41.5**	**34.4/40.8**	18.1/21.6	**16.6/17.8**	**17.4/19.7**	**17.3/19.5**
PSGTR [32]ECCV'2022	13.9	-	-	-	-	**28.4/36.3**	16.6/22.1	22.5/**29.2**	20.9/27.5
PSGFormer [32]ECCV'2022	36.8	-	-	-	-	18.0/20.1	14.8/17.6	16.4/18.9	16.2/18.8
C-SGG*(Ours)	55.4	36.5/46.5	32.5/36.4	34.5/41.5	34.4/40.8	24.0/29.0	**24.1/26.3**	24.0/27.7	24.0/27.6
CV-SGG*(Ours)	55.4	38.0/49.1	30.0/33.8	34.0/41.5	33.5/40.0	25.3/29.8	23.0/25.8	**24.2**/27.8	**24.1/27.7**

Table 2. Comparison results of SOTA SGG methods on the PSG dataset. The excellent results based on the same panoptic segmentation model has been marked in blue, and the best results has been marked in red. * indicates that using the newer panoptic segmentation results [15].

ter in common predicates but falls off a cliff in tail predicates. For these tail predicates, our context augmentation method can evolve different context description samples for conducting unbiased training, which eliminates long-tailed bias. In addition, for tail predicate categories, such as $says$ and $flying\ in$, the recall performs well. It is due to the strong correlation between predicates and object categories, and our method may learn some fixed collocation from context. For example, the tail predicate $flying\ in$ always appears with $airplane$, and $says$ always appears with $sign$ at the same time.

4.3. Evaluation on PSG Dataset

We also perform our method on the PSG dataset and the results are shown in Table 2. For a fair comparison, we divide the previous methods into two groups. The first group used the same panoptic segmentation results from [32] for the SGG task. In the second group we substitute the newer panoptic segmentation results [15] to generate scene graphs and compare with recent end-to-end models.

Panoptic quality (PQ) [8] is an evaluation metric for panoptic segmentation, and in the first group, the two-stage SGG methods all make predictions based on the same panoptic segmentation results. Due to the PSG dataset being relatively new, there is almost no research on the long-tailed bias in the PSG dataset. Our method with cheap context augmentation is optimal on the mR@K, Mean@K and F@K metrics on the PSG dataset. As for the second group, the end-to-end methods PSGTR [32] and PSGFormer [32] estimate panoramic segmentation results and scene graphs directly from images. Our method is a flexible two-stage method, in the first stage, we can utilize the SOTA panoptic segmentation method [15], and the second stage uses our methods for scene graph inference. From the results, our CV-SGG can improve on the R@K while slightly decreasing on the mR@K, which is similar to the VG results. In fact, PSGTR and PSGFormer output both panoramic segmentation and scene graph, and it is difficult to both account for the PQ of panoptic segmentation and the recall of SGG.

4.4. Model Size and Speed

In our opinion, inference speed is also an important evaluation metric for the SGG task. Although current deep-learning algorithms can achieve object detection in real-time (FPS>30), no SGG algorithm that can infer in real-time due to the quadratic time complexity. Our CSGG method does not need to extract visual features and can be embedded into the backend of any real-time object detection method to achieve real-time SGG. Here we choose the yolov5l [5] model to cooperate with our method for real-time SGG, and the comparison with the previous method is shown in Table 3. For a fair comparison in inference speed [18], we also test these mentioned methods on a RTX2070 SUPER GPU device based on open codes [18, 26, 32].

In terms of model size, C-SGG merely has three fully connected layers with few model parameters, while yolov5l has more parameters. As for FPS, C-SGG has great advantages, with the floating point operations (FLOPs) for each object pair only 0.2M. Even if there are 10000 object pairs (100 objects) in the image to be detected, only 2 GFLOPs are required, far less than once image feature extraction. Our C-SGG method may be the first high-performance SGG method capable of running in real-time. In table 3, based on C-SGG, only the most possible 100 object pairs are selected for CV-SGG, which greatly reduces the time complexity and accelerates CV-SGG inference.

4.5. Ablation study

We perform extensive ablation studies to explore in detail the impact of hyperparameter factors in C-SGG and CV-SGG. For our proposed context augmentation method, we have studied the influence of the random context augmentation factor ε in Table 4. The larger the ε is, the greater the change of context description. Too large ε may cause the model to learn some unreal predicates. We also show the performance of different boost factors η about ReLuL1 loss in Table 5. For our CV-SGG, ReLuL1 loss can bring substantial improvement compared with BCE loss, but extremely large η causes CV-SGG to lose performance on

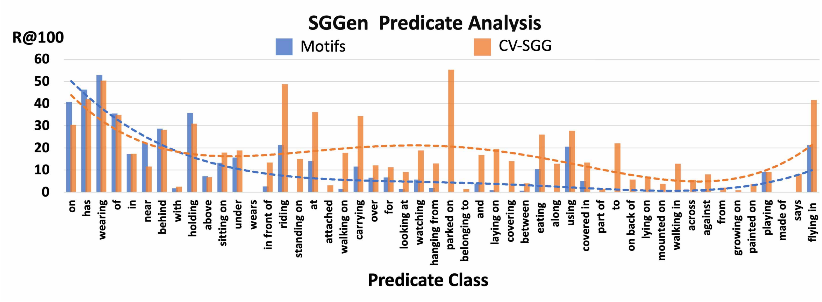

Figure 3. SGGen predicate analysis on the VG dataset. The predicate categories are ordered by the number of samples in the training set. Dotted lines represent trends.

Method	#Param (M)	Input Size	FPS
IMP	293.9	600×1000	4.05
VCTree	341.8	600×1000	4.24
Motifs	349.8	600×1000	4.00
FCSGG-W32-1S	31.8	512×512	10.3
FCSGG-W48-5S-FPN×2	83.0	640×1024	5.89
PSGFormer	50.4	800×1333	4.52
PSGTR	42.4	800×1333	3.2
Yolov5l+C-SGG	45+0.2	640×640	33.5
Yolov5l+CV-SGG	45+10	640×640	6.4

Table 3. Comparisons of Inference Efficiency.

Model	Augment	PredCls			
		R@50/100	mR@50/100	Mean@50/100	F@50/100
C-SGG	-	43.8/48.0	31.2/35.6	37.2/41.8	36.4/40.9
C-SGG	$\varepsilon = 0$	59.2/61.9	26.3/29.5	42.7/45.7	36.4/40.0
C-SGG	$\varepsilon = 0.01$	**59.3/62.4**	27.8/30.2	43.6/46.3	37.9/40.7
C-SGG	ε = **0.05**	55.2/59.2	32.9/36.0	**44.1/47.6**	41.2/44.7
C-SGG	$\varepsilon = 0.1$	52.2/56.4	**33.8/37.2**	43.6/45.1	**41.4/45.0**

Table 4. Ablation study about augment factor ε on the VG dataset. $-$ indicates no bounding box locations in C-SGG, and $\varepsilon = 0$ indicates the bounding box locations is used, but no context augmentation is done.

Model	ReLuL1	PredCls	
		R@50/100	mR@50/100
CV-SGG	-	60.2/62.5	23.1/26.4
CV-SGG	$\eta = 0.05$	57.9/61.5	31.8/35.1
CV-SGG	η = **0.1**	58.2/62.4	**32.6/36.2**
CV-SGG	$\eta = 0.2$	**59.3/63.2**	27.7/30.1

Table 5. Ablation study about ReLuL1 loss and boost factor η on the VG dataset. $-$ indicates that the BCE loss is used.

Model	Experience	SGGen			
		R@20/100	mR@20/100	Mean@20/100	F@20/100
CV-SGG*	$\alpha = 0$	25.0/30.1	12.6/13.1	18.8/22.1	16.8/19.2
CV-SGG*	$\alpha = 0.1$	26.2/31.2	14.8/16.3	20.5/23.8	18.9/21.4
CV-SGG*	$\alpha = 0.3$	**26.3/31.1**	16.3/18.2	21.3/24.7	20.1/23.0
CV-SGG*	$\alpha = 0.5$	25.7/30.2	22.3/24.8	24.0/27.5	23.8/27.1
CV-SGG*	α = **0.7**	25.3/29.8	23.0/25.8	**24.2/27.8**	**24.1/27.7**
CV-SGG*	$\alpha = 0.9$	24.0/28.4	**23.3/26.2**	23.7/27.3	23.6/27.3

Table 6. Ablation study about empirical factor α on the PSG dataset. When α=1, CV-SGG is equal to C-SGG.

Model	R^{mask}	SGGen	
		R@20/100	mR@20/100
(C)V-SGG*	$N_{mask} = 0$	**26.0/30.4**	16.5/18.2
CV-SGG*	$N_{mask} = 3$	24.4/29.2	22.8/25.9
CV-SGG*	$N_{mask} = 5$	24.9/29.8	**23.2/25.8**
CV-SGG*	N_{mask} = **10**	25.3/29.8	23.0/25.8
CV-SGG*	$N_{mask} = 20$	25.6/30.0	21.6/23.7

Table 7. Ablation study about the contextual mask R^{mask} on the PSG dataset. $N_{mask} = 0$ means no context R^{mask} guidance in the training, CV-SGG degenerates to visual SGG.

tail predicates. Our CV-SGG considers the final predicate comprehensively through an empirical factor α. In Table 6, when α is equal to 0, the model only believes in visual analysis. As α increases, the model more believes in context experience to judge predicates. We introduce a contextual mask R^{mask} to guide visual attention to possible predicates in CV-SGG. In Table 7, we show the effect of different mask size N_{mask} on the performance of CV-SGG. CV-SGG using R^{mask} guidance performs better in each predicate recall.

5. Conclusion

In this paper, we consider that visual appearance features may have an unessential effect on SGG and accordingly propose a C-SGG method solely using context descriptions. We notice that slight changes in the size and position of objects do not dramatically affect predicates between object pairs. Based on the local context samples of the dataset, we introduce the context augmentation method to produce diverse training samples at the context level, realizing unbiased training for C-SGG. Due to removing visual features, low computational cost allows C-SGG to achieve real-time SGG. Additionally, we also introduce a CV-SGG method that guides visual attention to possible predicates based on C-SGG results. Experiments demonstrate that our context-focused methods to SGG can alleviate long-tail bias and improve inference speed. We hope this phenomenon can inspire the following SGG research.

Limitations. Since our our method significantly weakens the visual information, it is still difficult to discern those complex predicates such as *against* and *belong to*. Besides, we use the local context description of object pairs, which may cause SGG to lack consideration of the global scene.

Acknowledgement

This research is supported by Key Research Project of Zhejiang Lab (No. G2021NB0AL03) and Young Scientists Fund of Zhejiang Lab (No. K2023NB0AA02).

References

[1] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2019. 3, 6

[2] Meng-Jiun Chiou, Henghui Ding, Hanshu Yan, Changhu Wang, Roger Zimmermann, and Jiashi Feng. Recovering the unbiased scene graphs from the biased ones. In *Proceedings of the ACM International Conference on Multimedia, ACM MM*, 2021. 1, 3, 6

[3] Alakh Desai, Tz-Ying Wu, Subarna Tripathi, and Nuno Vasconcelos. Learning of visual relations: The devil is in the tails. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV*, 2021. 1, 3, 5, 6

[4] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2014. 3

[5] Glenn-jocher, AyushExel, and et al. yolov5. https://github.com/ultralytics/yolov5, 2022. 7

[6] Yuyu Guo, Lianli Gao, Xuanhan Wang, Yuxuan Hu, Xing Xu, Xu Lu, Heng Tao Shen, and Jingkuan Song. From general to specific: Informative scene graph generation via balance adjustment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, CVPR*, 2021. 3, 6

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2016. 5

[8] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2019. 7

[9] Boris Knyazev, Harm de Vries, Cătălina Cangea, Graham W Taylor, Aaron Courville, and Eugene Belilovsky. Graph density-aware losses for novel compositions in scene graph generation. *arXiv preprint arXiv:2005.08230*, 2020. 3

[10] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 2017. 2, 5

[11] Lin Li, Long Chen, Yifeng Huang, Zhimeng Zhang, Songyang Zhang, and Jun Xiao. The devil is in the labels: Noisy label correction for robust scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2022. 1, 3, 5, 6

[12] Rongjie Li, Songyang Zhang, and Xuming He. Sgtr: End-to-end scene graph generation with transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19486–19496, 2022. 6

[13] Rongjie Li, Songyang Zhang, Bo Wan, and Xuming He. Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2021. 3, 6

[14] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision, ICCV*, 2017. 3

[15] Zhiqi Li, Wenhai Wang, Enze Xie, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, Ping Luo, and Tong Lu. Panoptic segformer: Delving deeper into panoptic segmentation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2022. 7

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision, ECCV*, 2014. 5

[17] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2020. 3, 7

[18] Hengyue Liu, Ning Yan, Masood Mortazavi, and Bir Bhanu. Fully convolutional scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2021. 1, 3, 6, 7

[19] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *Proceedings of the European Conference on Computer Vision, ECCV*, 2016. 1, 3, 5

[20] Yichao Lu, Himanshu Rai, Jason Chang, Boris Knyazev, Guangwei Yu, Shashank Shekhar, Graham W Taylor, and Maksims Volkovs. Context-aware scene graph generation with seq2seq transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, CVPR*, 2021. 3

[21] Zongshen Mu, Siliang Tang, Jie Tan, Qiang Yu, and Yueting Zhuang. Disentangled motif-aware graph learning for phrase grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence, AAAI*, 2021. 1

[22] Alejandro Newell and Jia Deng. Pixels to graphs by associative embedding. *Advances in Neural Information Processing Systems, NIPS*, 2017. 3

[23] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP*, 2014. 4

[24] Tianwen Qian, Jingjing Chen, Shaoxiang Chen, Bo Wu, and Yu-Gang Jiang. Scene graph refinement network for visual question answering. *IEEE Transactions on Multimedia*, 2022. 1

[25] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems, NIPS*, 2015. 3

[26] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, CVPR, 2020. 1, 3, 5, 6, 7

[27] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2019. 3, 5, 6, 7

[28] Yao Teng and Limin Wang. Structured sparse r-cnn for direct scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19437–19446, 2022. 3, 6

[29] Wenbin Wang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Exploring context and visual pattern of relationship for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2019. 3

[30] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, CVPR*, 2017. 3, 6, 7

[31] Shaotian Yan, Chen Shen, Zhongming Jin, Jianqiang Huang, Rongxin Jiang, Yaowu Chen, and Xian-Sheng Hua. Pcpl: Predicate-correlation perception learning for unbiased scene graph generation. In *Proceedings of the ACM International Conference on Multimedia, ACM MM*, 2020. 1, 3, 6

[32] Jingkang Yang, Yi Zhe Ang, Zujin Guo, Kaiyang Zhou, Wayne Zhang, and Ziwei Liu. Panoptic scene graph generation. In *Proceedings of the European Conference on Computer Vision, ECCV*, 2022. 2, 3, 5, 7

[33] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision, ECCV*, 2018. 1, 3, 6

[34] Jing Yu, Yuan Chai, Yujing Wang, Yue Hu, and Qi Wu. Cogtree: Cognition tree loss for unbiased scene graph generation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI*, 2021. 1, 3, 6

[35] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Bridging knowledge graphs to generate scene graphs. In *Proceedings of the European Conference on Computer Vision, ECCV*, 2020. 3, 6

[36] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2018. 1, 3, 6, 7

[37] Ao Zhang, Yuan Yao, Qianyu Chen, Wei Ji, Zhiyuan Liu, Maosong Sun, and Tat-Seng Chua. Fine-grained scene graph generation with data transfer. In *Proceedings of the European Conference on Computer Vision, ECCV*, 2022. 3, 6

[38] Ji Zhang, Kevin J Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. Graphical contrastive losses for scene graph parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2019. 3

[39] Jingyi Zhang, Yong Zhang, Baoyuan Wu, Yanbo Fan, Fumin Shen, and Heng Tao Shen. Dual resgcn for balanced scene graphgeneration. *arXiv preprint arXiv:2011.04234*, 2020. 3

[40] Fengda Zhu, Xiwen Liang, Yi Zhu, Qizhi Yu, Xiaojun Chang, and Xiaodan Liang. Soon: Scenario oriented object navigation with graph-based exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2021. 1