PCPL: Predicate-Correlation Perception Learning for Unbiased Scene Graph Generation

Shaotian Yan[1,*], Chen Shen[2,†], Zhongming Jin[2],
Jianqiang Huang[2], Rongxin Jiang[1,3], Yaowu Chen[1,4,†], Xian-Sheng Hua[2]

[1] Zhejiang University [2] DAMO Academy, Alibaba Group

[3] Zhejiang University Embedded System Engineering Research Center, Ministry of Education of China

[4] Zhejiang Provincial Key Laboratory for Network Multimedia Technologies

{yanshaotian, huaxiansheng}@gmail.com, {jason.sc, zhongming.jinzm, jianqiang.hjq}@alibaba-inc.com

ABSTRACT

Today, scene graph generation (SGG) task is largely limited in realistic scenarios, mainly due to the extremely long-tailed bias of predicate annotation distribution. Thus, tackling the class imbalance trouble of SGG is critical and challenging. In this paper, we first discover that when predicate labels have strong correlation with each other, prevalent re-balancing strategies (e.g., re-sampling and re-weighting) will give rise to either over-fitting the tail data (e.g., *bench sitting on sidewalk* rather than *on*), or still suffering the adverse effect from the original uneven distribution (e.g., aggregating varied *parked on/standing on/sitting on* into *on*). We argue the principal reason is that re-balancing strategies are sensitive to the frequencies of predicates yet blind to their relatedness, which may play a more important role to promote the learning of predicate features. Therefore, we propose a novel Predicate-Correlation Perception Learning (PCPL for short) scheme to adaptively seek out appropriate loss weights by directly perceiving and utilizing the correlation among predicate classes. Moreover, our PCPL framework is further equipped with a graph encoder module to better extract context features. Extensive experiments on the benchmark VG150 dataset show that the proposed PCPL performs markedly better on tail classes while well-preserving the performance on head ones, which significantly outperforms previous state-of-the-art methods.

CCS CONCEPTS

• **Computing methodologies** → **Scene understanding**.

KEYWORDS

Scene Graph Generation, Long-tailed Bias, Correlation Perception
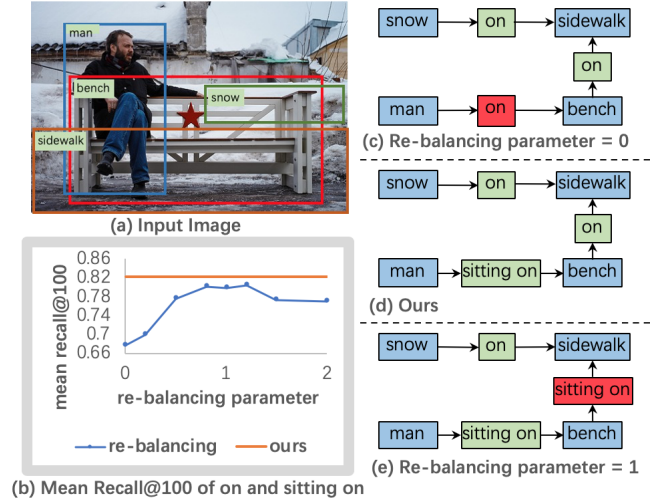
(a) Input Image

(b) Mean Recall@100 of on and sitting on

Figure 1: A comparison of re-balancing(e.g., re-weighting) methods and our PCPL. The training weight of re-balancing is set to nth power of the inverse of class sample frequencies where n is the value of re-balancing parameter. (a) An input image with bounding boxes and object labels. (b) The blue curve depicts the mean recall@100 of re-balancing for "on" and "sitting on" under different settings of parameter while the orange curve indicates the performance of our PCPL. It should be noted that there is no intersection between the PCPL and re-balancing training process. (c) SGG with re-balancing parameter = 0. (d) SGG from the proposed PCPL. (e) SGG with re-balancing parameter = 1. Red boxes in (c) and (e) denote wrong predication.

1 INTRODUCTION

Scene graph generation (SGG)[13], which is a visual task to detect objects and recognize semantic relationships between different objects in an image, can serve as a powerful structural representation

---

[*]This work was done during research intern at Alibaba.
[†]Corresponding authors.

of images and benefit other high-level Vision-and-Language tasks such as image generation[12, 33, 40], image retrieval[13, 24, 28, 34], visual question answering[6, 18, 39] and image captioning[7, 19, 38]. Taking advantage of the remarkable feature representations of convolutional neural networks (CNNs)[16] and diverse contextual feature fusion strategies (e.g., message passing[20, 37], lstm[41]), a variety of methods have made significant progress to improve the recall evaluation metric performance of SGG tasks. Some other works[3, 32] further utilize the co-occurring language regularity of typical subject-predicate-object relationship triplets as prior information to enhance overall performance. However, in practice the SGG benchmark datasets such as Visual Genome 150 (VG150)[15, 37] always have extremely long-tailed predicate label distributions (i.e., imbalanced annotation bias in training data, dominating by a few classes which occupy most of the data). Although achieving encouraging performance on head classes (e.g., on/has), these previous efforts are not feasible to obtain outstanding accuracy on predicting fewer but more meaningful predicate samples(e.g., sitting on/riding/looking at/eating/parked on), making them largely limited for supporting high-level tasks in real-world scenarios.

Prevalent class re-balancing strategies are introduced into SGG recently, examined by Tang et al. [31] at training stage in order to tackle the challenging long-tailed training data bias problems. In general, the prominent class re-balancing methods are roughly summarized as two types, which are adjusting the sample proportion within a mini-batch (i.e., re-sampling) or assigning relatively higher costs to tail samples (i.e., re-weighting). These two categories share the same connotation of manually tuning sampling frequencies or classifier weights based on the numbers of different class samples during training process to simulate the test distributions. These effective strategies indeed promote the overall mean recall evaluation metric for SGG benchmark datasets, however, when going deeper to examine specific predicate cases, we unexpectedly find that the performance is not satisfactory under the circumstances that predicate labels have strong correlation with each other. Fig. 1 comprehensively illustrates our observation. Taking a semantically closely related predicate pair — *on* (i.e., head class, occupying a large proportion of annotations) and *sitting on* (i.e., tail class, having rarely few samples) — as example, it can be seen from Fig. 1 that, class re-balancing strategies, which merely rely on the manual tuned classifier weights based on the numbers of samples, give rise to either over-fitting the tail data when re-balancing parameter is relative high (*bench sitting on sidewalk* rather than *on*, shown in Fig. 1(e)) or still suffering the side effect from the original uneven distribution when re-balancing parameter is relative low (aggregating *man sitting on bench* into *man on bench*). As shown in Fig. 1(b), the optimal point that maximizes the recall of both classes is hardly reachable by manually tuning the re-balancing parameter. We argue the principal reason is that re-balancing strategies merely utilize the frequencies of classes yet neglect their semantic relatedness, which may play a more important role to catalyze the learning of predicate features. To the contrary of other classification tasks, SGG essentially involves complex semantic correlations among different ground truth predicate annotations, which are insensitive to the class frequencies.

Consequently, we naturally put forward an assumption that the performance of predicates having strong correlations with multi classes will benefit from the learning of correlated ones, as a consequence of which, smaller loss weights are acceptable, otherwise those predicates tend to dominate other classes by severely degrading their recall, and vice versa. In view of that, we propose a novel Predicate-Correlation Perception Learning (PCPL for short) scheme aiming to tackle the class imbalance trouble of SGG, having the benefit of adaptively seeking out optimal loss weights by directly perceiving and explicitly utilizing the implicit correlations among predicates. Equipped with PCPL, the model is able to markedly improve the predicting results on tail classes and well preserve the performance on head predicates simultaneously, thus the optimum mean recall can be obtained as illustrated in Fig. 1(b). Specifically, we construct an iteratively updated class graph to perceive the correlations between predicates and the loss weights of classes are appropriately inversed with their relatedness derived from the graph.

Morever, we propose a graph encoding module (GE for short) to encode global context through a series of stacked encoders in a graph manner. A variety of methods have been adopted by previous works to fuse global context into relationship representations. Dual graph message passing[37] is relatively out-of-date while BiLSTM[9] employed by Neural Motifs[41] achieves better results yet suffers a drawback that different input orders will bring different results. Without introducing additional information, the performance of GGNN [3] is not satisfying. Compared with previous methods, our graph encoding module can better capture the relationships between object classes and benefit from being permutation invariant, thus can obtain more robust contextual features, setting a higher baseline.

In summary, the contributions of this paper are threefold:

- We propose a novel Predicate-Correlation Perception Learning (PCPL for short) scheme that is able to alleviate the long-tailed bias of SGG by directly perceiving and explicitly utilizing the implicit correlations between predicate classes, opening up new ideas to tackle the imbalance issue of SGG or other tasks involving correlated classes.
- PCPL can significantly promote the predicting results of tail classes while well preserving the performance of head predicates, by adaptively assigning optimal loss weights, which are appropriately inversed with the degrees of relatedness, to different predicates.
- Extensive experiments show the effectiveness of PCPL and demonstrate that PCPL achieves a new state-of-the-art.

2 RELATED WORK

2.1 Scene Graph Methods

Reasoning about relationships is the major challenge for generating scene graph. There are mainly two approaches in previous works making efforts to improve the performance of relationship prediction. The first approach is to make better use of visual features. Xu et al. [37] finds that relationship prediction can be greatly improved by jointly reasoning with contextual information. Message Passing model proposed by Xu iteratively refines its prediction by passing contextual messages along the topological structure of a scene graph. Zellers et al. [41] emphasizes the importance of context by introducing BiLSTM to encode global context that can

directly inform the local predictors. Second approach is to involve additional information such as semantic labels and statistical correlations to help prediction. Zellers et al. [41] embeds GloVe word vectors, statistical correlations of object pairs and relationships to visual features to obtain better results. Chen et al. [3] makes further use of statistical correlations. They facilitate scene graph generation by explicitly unifying the statistical knowledge with the architecture of graph neural network.

Chen et al. [3], Tang et al. [32] both take a notice on the class imbalanced issue of SGG by proposing the mean recall@K metric but their works are still confined to better feature extracting. In recent works, the criticalness of long-tailed bias of SGG is addressed [31, 36]. Tang et al. [31] employ causal inference in the prediction stage in an effort to remove the training bias while Wen et al. [36] make use of a pseudo-siamese network to pursue extracting balanced visual features. In contrast, our method perceives and
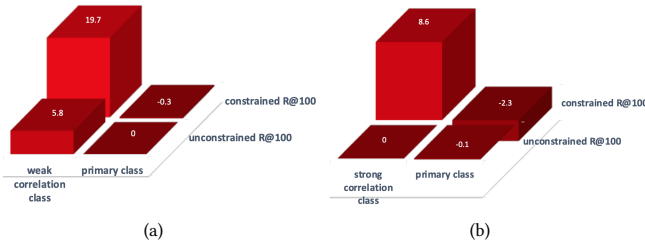

(a)   (b)

Figure 2: The improvements of constrained and unconstrained recall@100 of re-balancing over cross-entropy in %. (a) The results of weakly correlated group. (b) The results of strongly correlated group.

utilizes the implicit correlations among predicate classes based on an innovative observation, aiming to generate unbiased scene graph representations.

2.2 Class Imbalance

Real-world large-scale datasets often have long-tailed data distributions. Neural networks trained on these datasets tend to perform poorly on less presented classes. It has become a critical issue for model training. A lot of works has been done to resolve the class imbalance problem. Existing methods can be categorized as re-sampling [1, 5, 11, 29, 42] and re-weighting[4, 14, 25]. Re-sampling methods are simple yet effective. They often over-sample (e.g.,[2, 23]) less presented classes or under-sample (e.g.,[8, 17]) frequent classes to make the data distribution more balanced. However, they have their downsides. Under-sampling frequent classes will discard a large amount of data, causing waste of data. And it is not practicable when the dataset is extremely imbalanced. Over-sampling less presented classes can lead to over-fitting of the repeatedly sampled classes.

Re-weighting methods assign different weights for different classes to balance the loss. The simplest way of re-weighting is to set weights of classes as the inverse of their frequency[10, 35], but this causes poor performance on frequent classes. Cui et al. [4] proposes the definition of effective number of samples and re-weights the loss by the inverse of effective number to address this

issue. Another widely used re-weighting method is focal loss proposed by Lin et al. [22]. Focal loss down-weights the loss assigned to well-classified examples and focuses training on a sparse set of hard examples.

While most of traditional re-balancing methods merely rely on sample frequencies to manually tune the loss weights or sample ratios of different classes, our proposed method is able to adaptively assign optimal training costs to classes based on their relatedness.

3 METHODS

3.1 Problem definition

Scene graph[13], representing a visual scene's detailed semantics, is generated with:

- a set of bounding boxes $B = \{b_1, b_2, \text{fi}, b_n\}$, referring to the spatial locations of detected regions,
- a set of labels $O = \{o_1, o_2, \text{fi}, o_n\}$, containing object label o_i of the corresponding bounding box b_i,
- and $R = \{r_{1->2}, r_{1->3}, \text{fi}, r_{n->n-1}\}$, denoting the relationships of object pairs.

A triplet of a start object (o_i, b_i), an end object (o_j, b_j) and a predicate label $p_{i->j}$ connecting the former to the latter make up $r_{i->j} \in R$.

As shown in Fig. 3(d), we conduct a conventional two-stage pipeline which detects the locations and labels of objects first and then outputs the relationship representations. Given an input image containing two strong correlated predicates with great disparity in sample frequencies (i.e.,*parked on* and *on*), the ubiquitous cross entropy loss, employed by most of SGG methods to optimize the framework, causes aggregating *parked on* into *on*, as can be seen in Fig. 3(a). At the other extreme, Fig. 3(b) illustrates that a typical re-balancing strategy, which assign the fixed inverse of frequencies to the sample weights of predicate classes, surprisingly give rise to over-fitting *parked on*. Both strategies fail to achieve satisfactory performance when there exists strong correlation between predicate classes with long-tailed distribution. In view of that, we propose a novel PCPL scheme aiming to tackle the class imbalance trouble of SGG by directly perceiving and explicitly making use of the implicit correlations among predicates. Fig. 3(c) presents an overview of PCPL. We construct an iteratively updated class graph to represent the correlations between predicates. By utilizing the relatedness derived from the graph, PCPL has an advantage of adaptively seeking out optimal loss weights instead of manually tuning. Equipped with PCPL, the model is able to markedly improve the predicting results on tail classes and well preserve the performance on head predicates simultaneously. In subsequent sections, we will start with an innovative observation of re-balancing on SGG and then describe our method in detail.

3.2 An observation of re-balancing

As mentioned above, we find that re-balancing methods are of no avail when predicates are closely correlated to each other. In an effort to further explore the origins of this phenomenon, this section will discuss the influence of class correlations on the effectiveness of re-balancing for SGG before diving into our method.
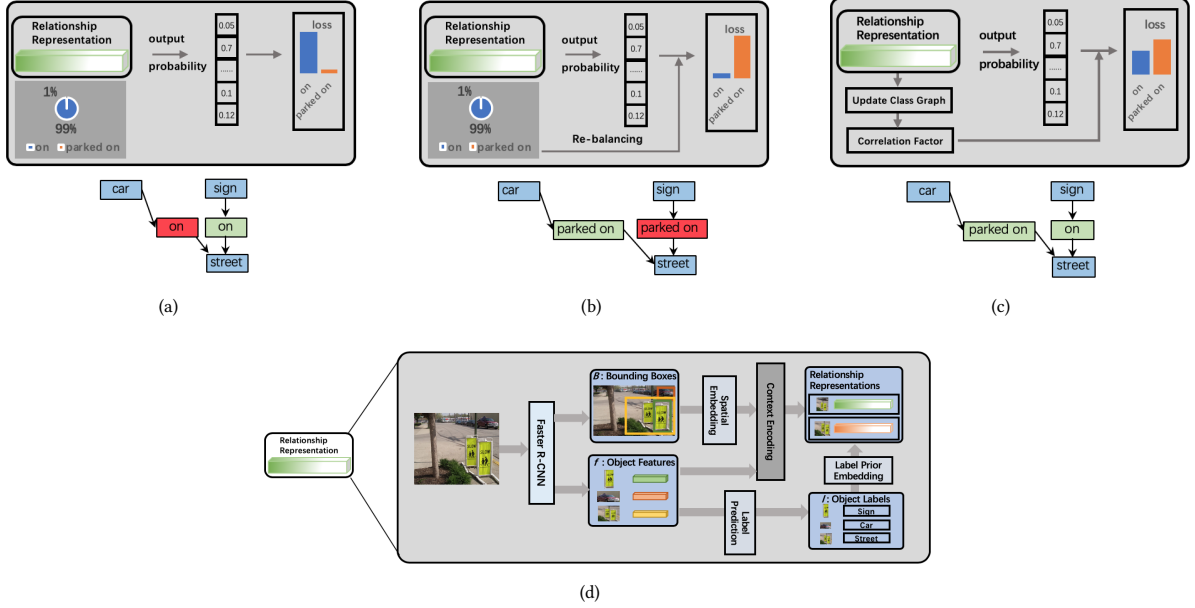
Figure 3: Illustration of how the baseline method (a), the re-balancing methods (b) and PCPL (c) generate scene graph from relationship representation and the corresponding output scene graph. Red boxes in (a) and (b) denote wrong predictions. (d) The pipeline used to acquire relationship representations.

Considering that it is hard to judge the degree of correlations by human intuition, we utilize the distance between class centers in feature space as a quantitatively measurement for the relatednesses, as the feature clusters of two correlated classes tend to be closer than two independent classes. Concretely, we employ a learnable variable $\mathbf{v}_k$, jointly trained with the SGG model, to represent the center of predicate class k in the feature space. The dimension of $\mathbf{v}_k$ is the same as the output relationship feature before the last fully connected layer of SGG model. For output features $\{f_1, f_2...f_N\}$ and predicate labels $\{l_1, l_2...l_N\}$, the loss to update $\mathbf{v}$ is defined as:

$$L_{center} = \frac{1}{N} \sum_{i=1}^{N} (f_i - \mathbf{v}_{l_i})^2 \qquad (1)$$

where N is the count of ground truth predicate annotations in the mini-batch and $\mathbf{v}_{l_i}$ is the corresponding center variable for output feature f_i. Notably, the gradient of $\mathbf{v}$ will not pass to feature f in backward propagation in order not to mess up the training procedure of SGG model.

Directly following the training process, we acquire the correlations between predicate classes:

$$\mathbf{e}_{kj} = ||\mathbf{v}_j - \mathbf{v}_k||_2 \qquad (2)$$

Classes with larger $\mathbf{e}$ are more independent while smaller $\mathbf{e}$ means stronger correlation. $\mathbf{e}$ between class k and it self equals 0.

To provide a more intuitive illustration, comparative experiments are designed for two observation predicate groups with opposed level of correlations. Specifically, The first group consists of two predicates, a primary class, occupying a large proportion of annotations, and a strong correlation class, with far fewer samples but closely correlated with the primary one. The same primary class

and a weak correlation class, having the same sample frequency with the strong one yet relatively independent, make up the other group. To get rid of the influence of other classes, we remove irrelevant annotations from the dataset for each group separately, thus to make the results clearer and more concise. Here we regard "has", "with" and "looking at" as the primary predicate, strong correlation predicate and weak correlation predicate respectively. Obtained from Eq. 2, $\mathbf{e}$ between *has* and *with* is 4.57 while that between *has* and *looking at* is 20.96, which means that the relatedness between *has* and *with* is strong and that between *has* and *looking at* is weak. For a fair comparison, we randomly down sample the frequency of *with* to the same scale of *looking at*.

Examined with a same baseline model on each group, the constrained and unconstrained R@100 improvements of re-weighting over cross-entropy are revealed in Fig. 2. The results demonstrate that both the constrained and unconstrained R@100 of the weak correlation predicate increase notably with almost no impact on the primary predicate. In the other group, though the constrained R@100 of the strong correlation predicate occurs a minor rise, that of the primary predicate happens a relatively significant decrease, while there is no obvious change on the unconstrained R@100 of both predicates. The contrast results of the two groups indicate that re-balancing, to some extent, is able to alleviate the class imbalance trouble when classes are independent. However, when it comes to classes closely correlated with each other, these strategies, sensitive to class frequency but blind to the correlations between classes, result in over-fitting to tail classes. Scene graph generation task, predicating relationship between instances, involves critically complex correlations between predicates, which fully exposes the

shortcoming of re-balancing. In stark contrast, our proposed PCPL scheme can achieve a satisfying performance on both head and tail classes by adaptively assigning optimal training costs, which are appropriately inversed with the degrees of relatedness, to predicates, opening up new ideas to tackle the imbalance issue of SGG or other tasks involving correlated classes. Although predicates having strong correlations with multi classes are assigned with relatively smaller loss weights, their performance will benefit from the learning process of correlated ones, while other predicates gain improvements on account of higher training costs. The detailed process of PCPL will be described in the next section.

3.3 Learning Process

The class correlations are dynamically changing along with the optimization of the feature extracting network. For this reason, as is shown in Fig. 4, we construct a learnable class graph to dynamically perceive the relatedness between predicates. As the network gradually converges, the graph we built is also achieving a relatively stable state. In this way, we are able to guide the learning of the model throughout the whole training process. The graph consists of a set of nodes and edges connecting every pair of them. Each node represents the center of one predicate class while the edges connecting nodes represent their degrees of correlation. Given output features, we first update the corresponding $\mathbf{v}_i$ using Eq. 1 and update the edges with Eq. 2 afterwards, as presented in Fig. 4(b,c,d). Then the global correlations $\mathbf{u}_i$ of predicate class i is defined as:

$$\mathbf{u}_i = \sum_{j=1}^{N} \mathbf{e}'_{ij} \tag{3}$$

where N is the number of predicate classes in the dataset and $\mathbf{e}'_{ij}$ is the updated value of edge connecting node i and node j. Following this, we perform a normalization for $\mathbf{u}_i$ to obtain the correlation factor τ:

$$\tau_i = \frac{\mathbf{u}_i - min(\mathbf{u}) + \epsilon}{max(\mathbf{u}) - min(\mathbf{u})} \tag{4}$$

where ϵ denotes a minimal value to prevent τ_i from being zero. The correlation factor τ_i can be seen as a measure for the independence degree of class i. After that, τ_i is assigned to the classification loss weight of the SGG network, thus to correct the learning process and alleviate the training bias:

$$p'_{l_i} = \frac{e^{p_{l_i}}}{\sum_{j=1}^{N} e^{p_j}}, \tag{5}$$

$$L = -\sum_{i=1}^{N} \frac{\tau_{l_i}}{\sum_{k=1}^{N_r} (\tau_{l_k})} * \log p'_{l_i} \tag{6}$$

where N_r is the count of ground truth predicate classes present at the current mini-batch, p is the probability of each predicate output by the model and l_i is the ground truth label of feature i.

Moreover, the dynamic graph makes it possible to alleviate the influence of noisy labels. Models are easy to be distracted by noisy labels because their losses are usually higher than normal samples. With the graph, we are able to distinguish and abandon noisy labels, thus to make the learning process more stable to some extent.

Given an output feature f_i with ground-truth label i, we distinguish whether it is noisy or not by D_{drop}:

$$D_{drop_j} = ||f_i - \mathbf{v}_i||_2 - ||f_i - \mathbf{v}_j||_2 - \frac{\mathbf{e}_{ij}}{\lambda} \tag{7}$$

where λ is a hyper-parameter and D_{drop_j} means D_{drop} with class j. Here we set λ as 2. If any of D_{drop_j} is great than zero, we consider f_i as noisy and abandon the corresponding sample.

3.4 Context Encoding

Given an image I, bounding boxes are first detected using Faster R-CNN as described in Fig. 3(d). Besides, for each b_i in the proposal region set B, it also outputs a corresponding feature vector and a possible label l_i, which are of non-context, causing relatively low performance in object and predicate classification. Thus, as shown in Fig. 5, we design a graph encoding module to obtain contextualized representations. Taking the pooled feature vectors as a set of nodes, we use an input network implemented by fully connected layers to expand bounding box coordinates to the same dimension of node features. Following that, we perform an element-wise sum to acquire new representations of nodes, containing spatial information which is also crucial when inferring relationships. Afterwards, we can construct a fully connected undirected graph G by connecting all the nodes together. The edges between nodes represent to what extent nodes can interact with its neighbors. Then we iteratively process the graph with stacked encoders. As Fig. 5 illustrates, each encoder consists of a self-attention layer and a Feed Forward network (FF). Every encoder calculates the attention coefficients between nodes and obtains the hidden state of each node by attending over its neighbors:

$$\hat{\mathbf{H}}_{i-1} = \mathbf{H}_{i-1} + \mathbf{Attention}(\mathbf{H}_{i-1}) \tag{8}$$

$$\mathbf{H}_i = \hat{\mathbf{H}}_{i-1} + \mathbf{FF}(\hat{\mathbf{H}}_{i-1}) \tag{9}$$

where $\mathbf{H}_i$ is the hidden state of graph G output by ith encoder. In this way, messages can be propagated through the whole graph. Eventually, we obtain the final contextual representation of each region after processing several encoders.

4 EXPERIMENTS

4.1 Experiment Settings

4.1.1 Implementation Details. To keep consistent with previous works [3, 32, 37, 41], we adopt the Faster-RCNN detector[26], pretrained on ImageNet[27] and refined on VG150[37], with VGG16[30] being the backbone to generate region proposals. The numbers of stacked encoders and attention heads in graph encoding modules are set to 6 and 12 respectively. All our experiments are conducted using a NVIDIA P100 GPU.

4.1.2 Dataset. We evaluate our methods and all the comparison models on Visual Genome[15], a large-scale dataset commonly used in vision-and-language tasks. Following prior works[3, 32, 37, 41], we adopt the most popular preprocessed split, VG150[37], which contains the most frequent 150 object categories and 50 predicate classes.
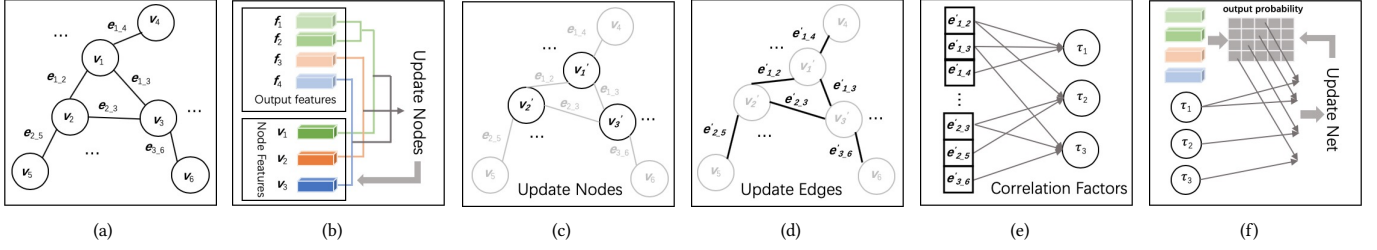
Figure 4: Illustration of the proposed Predicate-Correlation Perception Learning (PCPL) scheme. (a) A learnable class graph is constructed with each node representing the center of one predicate class and edges representing their correlations. (b) (c) (d) The graph is jointly trained with SGG model. (e) Correlation factors are derived from the graph. (f) We utilize correlation factors to adaptively assign optimal loss weights to predicate classes.

Table 1: Performance comparison with state-of-the-art methods on VG150 dataset. The constrained and unconstrained mR@50/100 in % on PredCls, SGCls and SGGen tasks are presented. As VCTree and TDE do not report the unconstrained mR@K metric, they are not listed in unconstrained results.

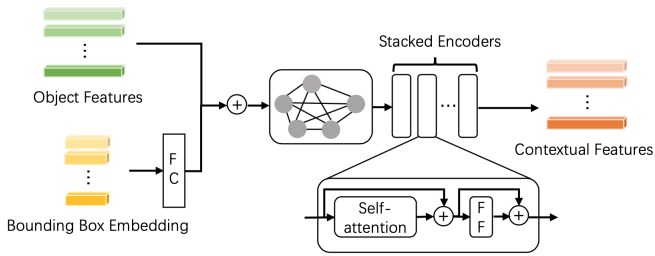|	Methods	PredCls		SGCls		SGGen		Mean
		mR@50	mR@100	mR@50	mR@100	mR@50	mR@100	
unconstrained	IMP+[37]	20.3	28.9	12.1	16.9	5.4	8.0	15.3
	FREQ[41]	24.8	37.3	13.5	19.6	5.9	8.9	18.3
	SMN[41]	27.5	37.9	15.4	20.6	9.3	12.9	20.6
	KERN[3]	36.3	49.0	19.8	26.2	**11.7**	**16.0**	26.5
	Ours	**50.6**	**62.6**	**26.8**	**32.8**	10.4	14.4	**32.9**
constrained	IMP+[37]	9.8	10.5	5.8	6.0	3.8	4.8	6.8
	FREQ[41]	13.3	15.8	6.8	7.8	4.3	5.6	8.9
	SMN[41]	13.3	14.4	7.1	7.6	5.3	6.1	9.0
	KERN[3]	17.7	19.2	9.4	10.0	6.4	7.3	11.7
	VCTree[31]	17.9	19.4	10.1	10.8	6.9	8.0	12.2
	SMN+TDE[31]	25.5	29.1	13.1	14.9	8.2	9.8	16.8
	Ours	**35.2**	**37.8**	**18.6**	**19.6**	**9.5**	**11.7**	**22.1**

Figure 5: A diagram of the Graph Encoding Module (GE). We fuse object features as well as their corresponding spatial information to construct a graph and obtain contextual features by processing the graph with stacked encoders. Each encoder is permutation invariant by consisting of a self-attention layer and a Feed Forward network (FF).

4.1.3 Evaluation Metrics. Most of previous works adopt the recall@K (R@K for short) metric that measures the fraction of ground-truth relationship triplets(subject-predicate-object) that appear among the top K most confident predictions in an image[37]. However, this metric is easily dominated by a few predicate classes accounting for absolute proportion of data due to the long-tail distribution of annotations. Thus we abandon R@K on most experiments and evaluate all the methods using the mean rcall@K (mR@K for short), proposed by Chen et al. [3] and Tang et al. [32], to give a more comprehensive assessment. It is defined as the average R@K of all the predicate classes, which gives a fair performance appraisal for both head and tail classes. Notably, we report R@K in Table 2, which compares different debiasing methods, to avoid over-fitting to tail classes. Both unconstrained and constrained [41] mR@K are presented on all experiments, which obtained from multi and single output relationships respectively.

4.1.4 Tasks. To comprehensively evaluate the performance on different stages of SGG, we adopt the following three tasks: Predicate classification (PredCls) predicts the predicate classes of a set of given object pairs with ground truth bounding boxes and object labels. Scene graph classification (SGCls) predicts the object classes for ground truth bounding boxes and predicts the predicate labels of each object pairs. Scene graph generation (SGGen) only takes
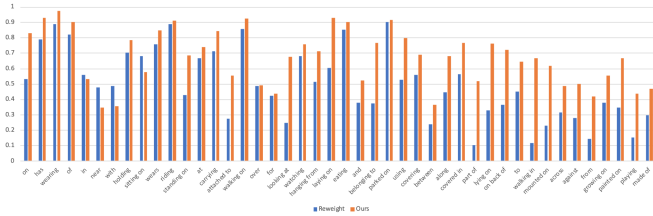
Figure 6: Performance comparison between re-weighting and our method on the VG150 dataset. The unconstrained R@100 for each predicate class on the PredCls task is presented.

the original image as input and sequentially detects the bounding boxes, object labels and then predicts the relationships between object pairs.

4.2 Compared methods

In this section, we first perform a thorough comparison between our proposed method and the existing state-of-the-art methods of scene graph generation, including Iterative Message Passing (IMP+)[37], Frequency baseline (FREQ)[41], Stacked Motif Networks (SMN)[41], Knowledge-Embedded Routing Network (KERN)[3], Visual Contexts Tree (VCTree)[32] and Stacked Motif Networks with TDE (SMN + TDE) [31]. As shown in Table. 1, we present the unconstrained and constrained mR@K on three tasks on the VG150 benchmark. KERN, which explicitly uses the statistical co-occurring prior outperforms SMN by 5.9% and 2.7% of the mean mR while VCTree, which mines the implicit relatedness between object pairs by learning a score matrix, gains a slight improvement over KERN. Though these methods achieve significant progress, TDE, as state-of-the-art on mR@K, is the first method focusing on the long-tailed trouble of SGG. While TDE is a prediction strategy, our proposed PCPL is a training scheme. Our method achieves higher constrained mR@K than others on all three tasks, gaining the mean mR of 22.1% and 32.9%, with a relative improvement of 31.5% and 24.2% compared with the previous state-of-the-art methods (i.e., SMN+TDE and KERN). Our model evidently outperforms others on PredCls and SGCls in the unconstrained mR@K metric, only slightly lower than KERN on the SGGen task, principally due to the statistical co-occurrence of objects KERN uses to promote the performance of object detection which is not the main concern of our discussion.

Secondly, we perform a more in-depth comparison between our method and several commonly used class imbalance handling strategies as well as the previous state-of-the-art debiasing method of SGG (SMN+TDE) to further demonstrate the effectiveness of PCPL. We retrain our baseline model (GE) with focal loss[22], class balanced loss[4] and weighted cross entropy loss respectively. The results of R@50/100 and mR@50/100 on three tasks are listed in Table. 2. Focal loss, which assigns larger training costs to hard samples, leads to a decline of mR@50/100 from GE. Re-balancing methods, i.e. class-balanced loss and re-weighting gain significant improvements on mR@50/100 but occur huge decrease on R@50/100, indicating over-fitting to tail classes. SMN+TDE[31] achieves a relatively balanced performance, promoting the mR@50/100 while preferably

keeping the performance of R50/100. Our method gains further increase on mR@50/100 from re-weighting and acquire comparable R@50/100 with SMN+TDE, though their detector is equipped with resnet[21], which is a more powerful backbone than the VGG16 we use. The comparison suggests that our proposed PCPL can supervise the model to learn a more unbiased representation of scene graph.

Fig. 6 presents a comparison between our method and re-weighting of the detailed recall@100 of PredCls task on each predicate class ranking by sample frequencies. PCPL performs better than re-weighting on almost all the predicate classes. While evidently promoting the performance of tail classes with few training samples like *walking in*, *mounted on* and *painted on*, PCPL obtains recall@100 on the four head predicates, which accounting of nearly 70% of the training data, as 83%, 93%, 97.5% and 90.2%, with improvements of 29.9%, 13.8%, 8.6% and 8.1% over those of re-weighting. Re-balancing strategies blindly restrain the training of head classes and encourage tail predicates while disregarding the correlations between them, gaining unworthy improvements on tail classes at the cost of massive decrease of the results on head predicates (e.g., *on*). On the contrary, PCPL adaptively assigns optimal loss weights appropriately inversed with their relatedness to predicate classes during the training process. The performance of predicates with weak correlations (e.g., *looking at*,*belonging to* and *playing*) improves on account of higher training costs. Although predicate classes having strong correlations with multi classes (e.g., *on*) are assigned with relatively smaller loss weights, their learning benefits from the training process of correlated ones (e.g., *parked on*,*standing on* and *walking on*), thus we are able to obtain relatively unbiased results on all predicates.

4.3 Ablation Study

We consider several ablations in Table. 3 and Table. 4. Table. 3 reveals the different ways to obtain class center representations (i.e., AvgCenter and LearntCenter, using the average of all features of a class to represent the center in every epoch and learning a class center end-to-end,respectively) and normalize the correlation factor(i.e., SoftmaxNorm, ScalingNorm and MinMaxNorm, using softmax function, divided with the maximum value and employing Eq. 4 to obtain correlation factor τ from global correlation **u**, respectively). Results show that the composition we use (i.e., LearntCenter + MinMaxNorm) acquires best performance. An explanation for the low performance of SoftmaxNorm and ScalingNorm is that the global correlation **u** of each predicate is roughly at the same scale, causing the distribution of correlation factor τ obtained using SoftmaxNorm or ScalingNorm is too smooth to make enough impact on the loss weights while MinMaxNorm magnifies the difference.

The contributes of this paper can be summarized as PCPL, the graph encoder and the noisy label dropping method. To better verify the effectiveness of each components, we perform an ablation study as listed in Table. 4. The performance of model with PCPL on all three tasks occurs an evident rise from GE, which clearly shows that our proposed PCPL greatly improves the generalization ability of the model. Meanwhile, GE still markedly outperforms IMP+, FREQ and SMN, indicating the effectiveness of the graph encoders in encoding context and extracting better visual features. Equipped

Table 2: Performance comparison with other debiasing methods on VG150 dataset. The R@50/100 and mR@50/100 in % with and without constraints on PredCls, SGCls and SGGen tasks are presented.

	Methods	PredCls		SGCls		SGGen		Mean
		R@50/100	mR@50/100	R@50/100	mR@50/100	R@50/100	mR@50/100	
unconstrained	GE + focal loss[22]	**77.3/85.4**	26.4/36.2	**42.3/46.1**	14.8/19.8	**18.3/23.7**	3.6/5.4	33.3
	GE + class-balanced loss[4]	57.0/70.8	35.1/44.9	33.2/39.9	19.1/24.0	8.4/12.8	6.1/8.9	30.0
	GE + re-weighting	56.5/70.7	39.0/49.6	32.0/38.9	20.6/25.8	8.1/12.1	6.5/9.4	30.8
	Ours	72.1/81.5	**50.6/62.6**	39.9/44.5	**26.8/32.8**	15.2/20.6	**10.4/14.4**	**38.4**
constrained	GE + focal loss[22]	**64.4/66.8**	16.7/18.4	**35.0/36.0**	8.7/9.4	**18.1/22.9**	3.5/4.9	25.4
	SMN+TDE[31]	46.2/51.4	25.5/29.1	27.7/29.9	13.1/14.9	16.9/20.3	8.2/9.8	24.4
	GE + class-balanced loss[4]	43.4/48.1	29.7/33.6	24.9/26.8	15.9/17.9	8.4/12.6	6.0/8.8	23.0
	GE + re-weighting	40.4/44.6	32.1/35.9	22.4/24.2	16.5/18.3	8.1/11.9	6.5/9.3	22.5
	Ours	50.8/52.6	**35.2/37.8**	27.6/28.4	**18.6/19.6**	14.6/18.6	**9.5/11.7**	**27.1**

Table 3: Performance comparison of different compositions of PCPL on VG150. The constrained and unconstrained mR@50/100 in % on PredCls, SGCls and SGGen tasks are presented.

	Methods	PredCls		SGCls		SGGen		Mean
		mR@50	mR@100	mR@50	mR@100	mR@50	mR@100	
unconstrained	LearntCenter + SoftmaxNorm	34.1	45.8	18.7	24.5	4.4	6.7	22.4
	LearntCenter + ScalingNorm	37.2	49.1	20.6	26.4	5.1	7.4	24.3
	AvgCenter + MinMaxNorm	49.7	61.9	25.4	31.8	9.2	12.0	31.7
	LearntCenter + MinMaxNorm	**50.6**	**62.6**	**26.8**	**32.8**	**10.4**	**14.4**	**32.9**
constrained	LearntCenter + SoftmaxNorm	17.4	18.9	9.1	9.7	3.9	5.3	10.7
	LearntCenter + ScalingNorm	19.0	20.5	10.1	10.7	4.5	5.8	11.8
	AvgCenter + MinMaxNorm	34.1	36.9	17.8	18.9	8.6	10.6	21.2
	LearntCenter + MinMaxNorm	**35.2**	**37.8**	**18.6**	**19.6**	**9.5**	**11.7**	**22.1**

Table 4: Ablation study of our method.The constrained and unconstrained mR@50/100 in % on PredCls, SGCls and SGGen tasks are presented.

	Methods	PredCls		SGCls		SGGen		Mean
		mR@50	mR@100	mR@50	mR@100	mR@50	mR@100	
unconstrained	GE	32.7	44.0	18.3	23.8	8.3	11.6	23.1
	GE+PCPL	50.1	61.9	26.1	32.3	10.1	14.2	32.5
	Ours	**50.6**	**62.6**	**26.8**	**32.8**	**10.4**	**14.4**	**32.9**
constrained	GE	17.3	18.7	9.3	9.8	5.5	6.5	11.2
	GE+PCPL	34.5	37.4	18.1	19.2	9.3	11.2	21.6
	Ours	**35.2**	**37.8**	**18.6**	**19.6**	**9.5**	**11.7**	**22.1**

with the noisy label dropping schema, the performance of model gains a sight further improvement, demonstrating its efficiency.

5 CONCLUSION

In this paper, we discover that the key challenge for generating unbiased scene graph lies in the complex relatedness among predicate classes. Thus, we propose a novel PCPL framework which can adaptively assign optimal loss weights to predicates by directly perceiving and explicitly utilizing the correlations among classes. PCPL is further equipped with a graph encoder module to better extract context features. Extensive experiments on the benchmark VG150 dataset show that PCPL performs markedly better on tail classes while well-preserving the performance on head ones, which significantly outperforms previous state-of-the-art methods in mean recall evaluation metric, demonstrating its effectiveness in removing the long-tailed bias of SGG.

ACKNOWLEDGMENTS

REFERENCES

[1] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. 2018. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks* 106 (2018), 249–259.

[2] Jonathon Byrd and Zachary Lipton. 2019. What is the Effect of Importance Weighting in Deep Learning?. In *International Conference on Machine Learning*. 872–881.

[3] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. 2019. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6163–6171.

[4] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9268–9277.

[5] Yonatan Geifman and Ran El-Yaniv. 2017. Deep active learning over the long tail. *arXiv preprint arXiv:1711.00941* (2017).

[6] Shalini Ghosh, Giedrius Burachas, Arijit Ray, and Avi Ziskind. 2019. Generating natural language explanations for visual question answering using scene graphs and visual attention. *arXiv preprint arXiv:1902.05715* (2019).

[7] Jiuxiang Gu, Shafiq Joty, Jianfei Cai, Handong Zhao, Xu Yang, and Gang Wang. 2019. Unpaired image captioning via scene graph alignments. In *Proceedings of the IEEE International Conference on Computer Vision*. 10323–10332.

[8] Guo Haixiang, Li Yijing, Jennifer Shang, Gu Mingyun, Huang Yuanyue, and Gong Bing. 2017. Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications* 73 (2017), 220–239.

[9] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[10] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. 2016. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5375–5384.

[11] Nathalie Japkowicz and Shaju Stephen. 2002. The class imbalance problem: A systematic study. *Intelligent data analysis* 6, 5 (2002), 429–449.

[12] Justin Johnson, Agrim Gupta, and Li Fei-Fei. 2018. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1219–1228.

[13] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. 2015. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3668–3678.

[14] Salman H Khan, Munawar Hayat, Mohammed Bennamoun, Ferdous A Sohel, and Roberto Togneri. 2017. Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE transactions on neural networks and learning systems* 29, 8 (2017), 3573–3587.

[15] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision* 123, 1 (2017), 32–73.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[17] Hansang Lee, Minseok Park, and Junmo Kim. 2016. Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning. In *2016 IEEE international conference on image processing (ICIP)*. IEEE, 3713–3717.

[18] Soohyeong Lee, Ju-Whan Kim, Youngmin Oh, and Joo Hyuk Jeon. 2019. Visual Question Answering over Scene Graph. In *2019 First International Conference on Graph Computing (GC)*. IEEE, 45–50.

[19] Xiangyang Li and Shuqiang Jiang. 2019. Know more say less: Image captioning based on scene graphs. *IEEE Transactions on Multimedia* 21, 8 (2019), 2117–2130.

[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 *Proceedings of the European Conference on Computer Vision (ECCV)*. 335–351.

[21] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2117–2125.

[22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*. 2980–2988.

[23] Enislay Ramentol, Yailé Caballero, Rafael Bello, and Francisco Herrera. 2012. SMOTE-RSB*: a hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using SMOTE and rough sets theory. *Knowledge and information systems* 33, 2 (2012), 245–265.

[24] Sahana Ramnath, Amrita Saha, Soumen Chakrabarti, and Mitesh M Khapra. 2019. Scene Graph based Image Retrieval–A case study on the CLEVR Dataset. *arXiv preprint arXiv:1911.00850* (2019).

[25] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. 2018. Learning to reweight examples for robust deep learning. *arXiv preprint arXiv:1803.09050* (2018).

[26] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*. 91–99.

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision* 115, 3 (2015), 211–252.

[28] Brigit Schroeder and Subarna Tripathi. 2020. Structured Query-Based Image Retrieval Using Scene Graphs. *arXiv preprint arXiv:2005.06653* (2020).

[29] Li Shen, Zhouchen Lin, and Qingming Huang. 2016. Relay backpropagation for effective learning of deep convolutional neural networks. In *European conference on computer vision*. Springer, 467–482.

[30] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[31] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. 2020. Unbiased scene graph generation from biased training. *arXiv preprint arXiv:2002.11949* (2020).

[32] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. 2019. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6619–6628.

[33] Subarna Tripathi, Anahita Bhiwandiwalla, Alexei Bastidas, and Hanlin Tang. 2019. Using scene graph context to improve image generation. *arXiv preprint arXiv:1901.03762* (2019).

[34] Sijin Wang, Ruiping Wang, Ziwei Yao, Shiguang Shan, and Xilin Chen. 2020. Cross-modal scene graph matching for relationship-aware image-text retrieval. In *The IEEE Winter Conference on Applications of Computer Vision*. 1508–1517.

[35] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. 2017. Learning to model the tail. In *Advances in Neural Information Processing Systems*. 7029–7039.

[36] Bin Wen, Jie Luo, Xianglong Liu, and Lei Huang. 2020. Unbiased Scene Graph Generation via Rich and Fair Semantic Extraction. *arXiv preprint arXiv:2002.00176* (2020).

[37] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. 2017. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5410–5419.

[38] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2019. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 10685–10694.

[39] Zhuoqian Yang, Zengchang Qin, Jing Yu, and Yue Hu. 2018. Scene Graph Reasoning with Prior Visual Relationship for Visual Question Answering. *arXiv preprint arXiv:1812.09681* (2018).

[40] LI Yikang, Tao Ma, Yeqi Bai, Nan Duan, Sining Wei, and Xiaogang Wang. 2019. Pastegan: A semi-parametric method to generate image from scene graph. In *Advances in Neural Information Processing Systems*. 3950–3960.

[41] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5831–5840.

[42] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. 2018. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*. 289–305.