Zero-Shot Scene Graph Generation via Triplet Calibration and Reduction

JIANKAI LI, YUNHONG WANG, and WEIXIN LI*, State Key Laboratory of Software Development Environment, School of Computer Science and Engineering, Beihang University; and Shanghai Artificial Intelligence Laboratory, China

Scene Graph Generation (SGG) plays a pivotal role in downstream vision-language tasks. Existing SGG methods typically suffer from poor compositional generalizations on unseen triplets. They are generally trained on incompletely annotated scene graphs that contain dominant triplets and tend to bias toward these seen triplets during inference. To address this issue, we propose a Triplet Calibration and Reduction (T-CAR) framework in this paper. In our framework, a triplet calibration loss is first presented to regularize the representations of diverse triplets and to simultaneously excavate the unseen triplets in incompletely annotated training scene graphs. Moreover, the unseen space of scene graphs is usually several times larger than the seen space since it contains a huge number of unrealistic compositions. Thus, we propose an unseen space reduction loss to shift the attention of excavation to reasonable unseen compositions to facilitate the model training. Finally, we propose a contextual encoder to improve the compositional generalizations of unseen triplets by explicitly modeling the relative spatial relations between subjects and objects. Extensive experiments show that our approach achieves consistent improvements for zero-shot SGG over state-of-the-art methods. The code is available at https://github.com/jkli1998/T-CAR.

CCS Concepts: • **Computing methodologies** → **Scene understanding**; **Image representations**.

Additional Key Words and Phrases: scene analysis and understanding, scene graph generation, compositional zero-shot learning

1 INTRODUCTION

Scene Graph Generation (SGG), which aims to detect object instances and their pairwise visual relationships, is crucial to visual comprehension [21]. Such objects and visual relationships provide a compact and structured description of scenes, which can be used for high-level Vision-Language tasks, *e.g.* visual question answering [2, 23, 30, 41], image captioning [5, 51, 56, 64], image retrieval [7, 11, 54, 55], *etc.*

In the literature, SGG is typically formulated as predicting a triplet tuple <subject-predicate-object> [46, 52]. As shown in Fig. 1, methods for zero-shot SGG need to learn from training triplets

---

*Corresponding author.

Authors' address: Jiankai Li, lijiankai@buaa.edu.cn; Yunhong Wang, yhwang@buaa.edu.cn; Weixin Li, weixinli@buaa.edu.cn, State Key Laboratory of Software Development Environment, School of Computer Science and Engineering, Beihang University; and Shanghai Artificial Intelligence Laboratory, China, 100191.
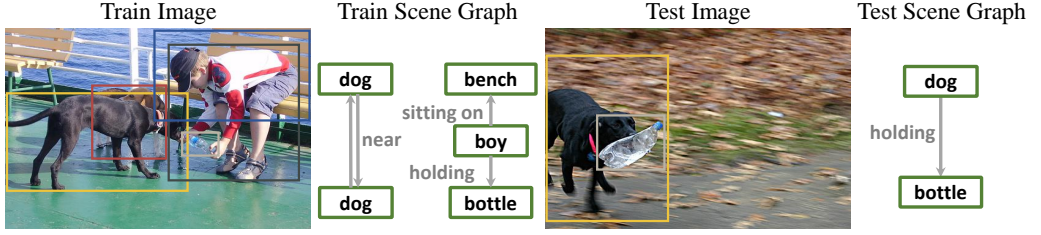
Fig. 1. Zero-shot scene graph generation aims to learn from triplets <subject-predicate-object> in the training set and infer their unseen compositions at test time.

and then infer unseen triplets from test images. It is easy for us humans to recognize <dog-holding-bottle> from <boy-holding-bottle> and <dog-near-dog> since we acknowledge the concepts of "boy", "dog", "bottle", and "holding". But it is extremely challenging for SGG models when facing the unseen composition <dog-holding-bottle> [20, 44] because they suffer from the problem of biased seen triplets and are insensitive to unseen compositions. Since less common triplets typically contain more information (according to the information entropy), this failure greatly impacts downstream tasks.

To improve the compositional generalization ability, previous works employ causal inference for unbiased prediction [44] or devise a Generative Adversarial Network (GAN) to generate unseen triplets [20]. Despite their remarkable progress in SGG, their performances in the zero-shot recall are still far from satisfactory. We attribute this weakened compositional generalization ability to the seen triplet bias, which mainly stems from two aspects.

On the one hand, dominant triplets lead to poorly discriminative representations of diverse triplets and bias the SGG model toward predicting frequent triplets. As shown in Figs. 2 (a) and 2 (b), several triplets dominate the dataset, even in the top three frequent predicates (*i.e.*"on", "has", and "wearing"). Moreover, given different pairs of subjects and objects, distributions of triplets are also dominated by several predicates, which may coincide with the distribution of predicates in the whole dataset or the opposite. On the other hand, recent works point out that the large-scale SGG benchmark contains many unlabeled, relatively rare, and meaningful relationships [9, 22]. These unseen triplets in the incompletely annotated training set are suppressed by classical cross-entropy loss, which forces the SGG model to bias toward predicting seen triplets. Previous works focus on the predicate-granularity re-balance and unseen sample mining rather than the triplet-granularity [9, 22, 44], resulting in unsatisfactory compositional generalization ability to zero-shot scene graph generation. We consider a calibration method to regularize the discrimination of diverse triplets and excavate unseen triplets to address the above two issues. However, exploring reasonable unseen triplets in an enormous unseen space is difficult. As shown in Fig. 2 (c), the unseen space is almost 37 times larger than the seen space in the Visual Genome dataset [21]. Thanks to the realistic meaning of the scene graph, most of the compositions of unseen triplets are unlikely to be present in the real world, *e.g.* <seat-eating-dog> [20]. Thus we further consider reducing the triplet space to shift the attention of our model to reasonable unseen compositions.

Based on the aforementioned observations, we propose a Triplet Calibration and Reduction (T-CAR) framework for zero-shot SGG to improve the model's generalization ability to unseen triplets. To this end, we first introduce a Triplet Calibration Loss (TCL) to regularize the discriminative representations of diverse triplets and excavate unseen triplets. TCL assigns triplet-specific calibrations on seen triplets to mitigate the bias toward frequent triplets and excavates unseen triplets
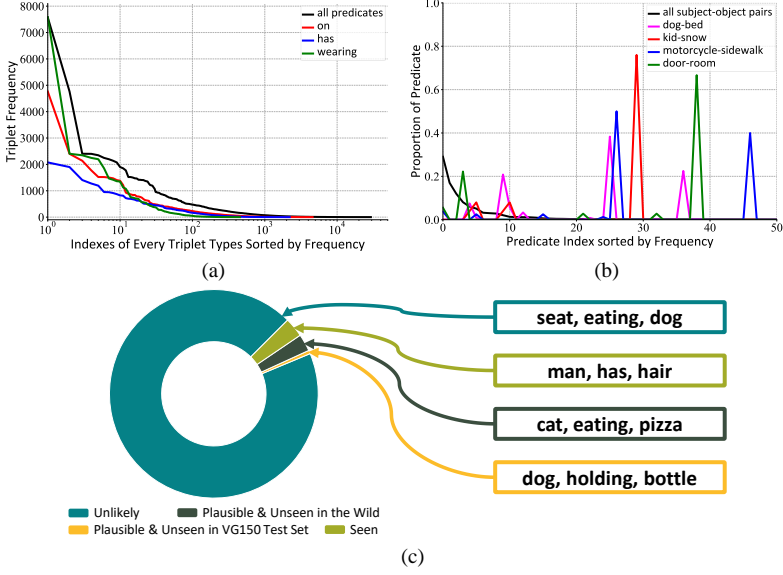
Fig. 2. (a) Several triplets dominate the Visual Genome [21] benchmark, and (b) the distribution of predicates may be opposite to the overall distribution of the dataset with the same subject and object. (c) Unseen triplet space in Visual Genome is very large, but most of them are unlikely to be present.

to resist their negative constraints imposed by cross-entropy. We then devise an Unseen Space Reduce Loss (USRL) to reduce the hindrance of mining unlabeled samples in such a huge unseen space with a large number of unreasonable compositions. USRL exploits the interchangeability of subjects, predicates, and objects to explore the rationality of unseen compositions based on seen triplet samples, which is reformulated as the positive-unlabeled learning (PU Learning) problem. It regards all the seen triplets in the training set as in-positive data and all background triplets as unlabeled data. Finally, a Contextual Encoding Network (CEN) is further proposed to encode the spatial relationships between subjects and objects. Compared with previous context models [24, 45, 58], it removes the linguistic priors and strengthens the relative position knowledge to reduce the seen triplet bias.

Extensive experiments on the Visual Genome dataset [21] are conducted to verify the effectiveness of our proposed modules. The experimental results demonstrate that our method outperforms state-of-the-art methods by a significant margin, *i.e.*, **12.5%**, **4.4%**, and **1.8%** of Zero-Shot Recall (zR@100) respectively for PredCls, SGCls, and SGDet tasks.

The main contributions of our work are summarized as follows:

(1) A triplet calibration loss is introduced to balance the constraints on unseen triplets that are incorrectly annotated as background and enhance the attention of the model on less frequent triplet types to improve the model capability in representing diverse triplet compositions.

(2) An unseen space reduction loss is proposed to reduce the huge unseen triplet space. The loss allows the model to search for reasonable unseen compositions in a small triplet space restricted by linguistic priors.

(3) A contextual encoding network is devised to explicitly encode the relative spatial features into triplet representations. Experiment results show that the relative spatial features are beneficial to distinguish unseen triplets.

2 RELATED WORK

2.1 Scene Graph Generation

Scene graph is a structured representation of image content, bridging vision and language. It is involved in and facilitates many visual-language tasks [7, 10].

Some early works on scene graph generation explore incorporating more knowledge from various modalities [26, 31, 40, 57, 61, 63], and other approaches study the context modeling of entities and relations [4, 8, 28, 32, 33, 48]. Besides, scene-parsing-based models [6, 49, 65] also demonstrate strong abilities in parsing relationships, *e.g.*, the cascaded scene parsing model which performs relation reasoning stage by stage, and achieves promising results [65]. A recently published survey [3] conducts a comprehensive investigation of current scene graph researches, showing the development of scene graph generation models. Although much progress has been made in the last few years in generating scene graphs on seen samples, existing methods still perform poorly in generating unseen triplets. The challenge of correctly predicting unseen triplets is not the same as the unbalanced predicate problem since frequent predicates and objects in the test set also dominate unseen triplets [20]. Various methods are proposed to address the zero-shot generalization problem. Some approaches try to increase the diversity of input scene graphs by augmenting scene graphs based on Generative Adversarial Network (GAN) [20]. Some methods mitigate training bias through the model designed, where they either directly generate triplets instead of following the two-stage paradigm [46] or draw the counterfactual causality [44]. Other approaches devise graph-normalized [19] or energy-based loss [43] to improve the compositional generalization. However, previous SGG works overlook the issues of large unseen triplet space containing lots of unrealistic triplets and unseen samples in the incompletely annotated training set. Their performance on zero-shot generalization is still far from satisfactory. Our work considers triplets calibration and unseen space reduction in a single framework. We propose a triplet calibration loss to regularize the triplet representations and excavate unseen triplets, and an unseen space reduction loss to reduce unseen space and shift the attention to reasonable unseen triplets.

2.2 Compositional Zero-Shot Learning

Compositional zero-shot learning (CZSL) is subordinate to zero-shot learning [36, 38, 53], which aims to transfer knowledge from seen to unseen compositions. And the goal of zero-shot SGG is to transfer knowledge from seen combinations of subjects, predicates, and objects to unseen combinations. Thus zero-shot SGG can be regarded as CZSL.

Typical CZSL works [15, 17, 25, 34] focus on the combinations of objects and states, which is different from zero-shot SGG. Compared to these works, zero-shot SGG owns a larger label space since it has three degrees of freedom. Taking the MIT-States [16], one of the biggest and most commonly used datasets for CZSL, as an example, there are 1,262 seen and 400 unseen compositions in the training and test sets, respectively. In contrast, VG150 [21] is composed of about 29 thousand seen, 5 thousand unseen, and 1 million potentially existing compositions in the training and test sets. In this paper, our T-CAR narrows down the huge potential triplet space, removes the most unconventional combinations, and reduces the learning and inference difficulties.

3 OUR APPROACH

3.1 Problem Formulation and Overview

3.1.1 Problem Formulation. Given an image $\mathcal{I}$, the task of scene graph generation (SGG) is to predict a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the entity node set and $\mathcal{E}$ denotes the set of relationships of two ordered entities. Each entity node $v \in \mathcal{V}$ is composed of a bounding box $\boldsymbol{b}$, a visual feature
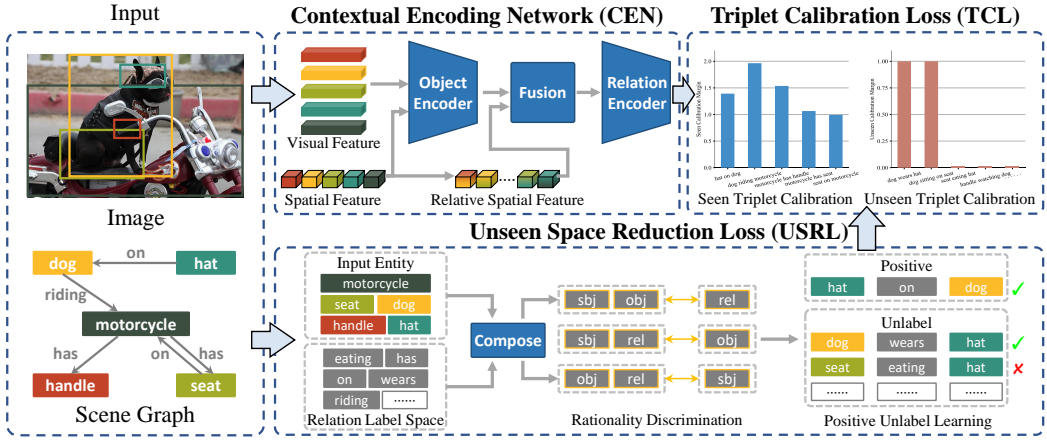
Fig. 3. The overall framework of our T-CAR model. Given an input image, T-CAR first detects its object proposals. Then the CEN encodes the entity and relation features. Meanwhile, USRL reduces a large number of impossible unseen triplets in the output space. TCL imposes calibration margins to each seen triplet based on its frequency, shifting attention to rare triplets and regularizing discriminative representations of diverse triplets. It also excavates unseen triplets in the training set according to the reasonable triplets provided by USRL.

v, and a class label $c_v \in C_e$. A relationship $r = (s, p, o) \in \mathcal{E}$ is a three-tuple, including a subject entity node s, an object entity node o, and its predicate label $p \in C_p$.

Scene graph generation is typically a three-stage task, which can be formulated as follows:

$$P(\mathcal{G}|\mathcal{I}) = P(\mathcal{B}|\mathcal{I})P(C_e|\mathcal{B}, \mathcal{I})P(\mathcal{G}|C_e, \mathcal{B}, \mathcal{I}), \tag{1}$$

where $P(\mathcal{B}|\mathcal{I})$ aims to detect bounding boxes of entities, $P(C_e|\mathcal{B}, \mathcal{I})$ works on predicting the entity categories, and $P(\mathcal{G}|C_e, \mathcal{B}, \mathcal{I})$ denotes the predicate classification.

3.1.2 Method Overview. An overview of our Triplet Calibration and Reduction (T-CAR) model is illustrated in Fig. 3. A contextual encoding network (CEN) is first introduced to generate context-aware representations of entity nodes and relationships with less seen triplet bias. Compared to previous contextual encoders, CEN removes the linguistic prior and explicitly models the relative positions between subjects and objects to reduce seen triplet bias and improve compositional generalization ability. After that, we propose a triplet calibrate loss (TCL) to alleviate the bias effect of dominant seen triplets and mine the unseen compositions. However, the unseen triplet space is very large and typically contains a huge number of unrealistic compositions. It is difficult to excavate and infer unseen samples in such an enormous space. We observe the substitutability of semantically similar terms between seen triplets. Based on this observation, we propose the unseen space reduction loss (USRL) for training and inference, eliminating the most unrealistic triplets based on linguistic knowledge.

3.2 Contextual Encoding Network

It is a long-standing paradigm in SGG to initialize entity representations with visual and linguistic (from their text labels) features and initialize relation representations with visual and spatial features of the subject and object. However, we find that the linguistic features bring a bias to the seen compositions, while the relative position cues between subject and object are neglected. From our perspective, these linguistic features, fixed for a certain class of entities, cause the SGG

model to learn the conditional distribution from the composition priors of the subject and object [58]. These composition priors contribute to seen triplets but weaken the ability to represent unseen compositions. Moreover, relative spatial cues are also indispensable for robust relationship predictions. Most existing methods [24, 45, 58] in SGG simply concatenate or sum up the spatial features of subjects and objects, which may be insufficient to mine the underlying spatial relations. To address these deficiencies, we propose a contextual encoding network, which removes the linguistic priors and explicitly models the relative spatial features. Our contextual encoding network is composed of an entity encoder, a fusion layer, and a relation encoder.

The entity encoder is responsible for refining entity features by interacting with contextual entities:

$$\{x_i\}_{i=1,\dots,N} = Enc_{obj}(\{[v_i, FFN(b_i)]\}_{i=1,\dots,N}), \tag{2}$$

where x_i denotes the refined entity representations, N is the number of entities, $[\cdot,\cdot]$ represents the concatenation operation, $FFN(\cdot)$ is a two-layer Multi-Layer Perceptron (MLP) with LeakyReLU activation, and Enc_{obj} denotes the entity contextual modules which could be the multi-layer LSTM [13], GNN [12], or Transformer [47]. Here we apply a four-layer Transformer as the entity encoder. Note that the object features are initialized without linguistic features.

The fusion layer aims to initialize predicate representation x'_p with refined entity representations of its corresponding subject x_s and object x_o:

$$x'_p = x_s \star x_o \star FFN([v_u, b_{s,o}]), \tag{3}$$

where v_u is extracted from the union box of subject s and object o. The operation $\star$ denotes the fusion function defined in [45]: $x \star y = ReLU(W_x x + W_y y) - (W_x x - W_y y) \odot (W_x x - W_y y)$, where $\odot$ denotes the Hadamard product, and W_x and W_y are parameters used for projection.

The relative spatial feature $b_{s,o}$ is calculated from subject and object bounding boxes, i.e. $b_s = (x_s^1, y_s^1, x_s^2, y_s^2)$ and $b_o = (x_o^1, y_o^1, x_o^2, y_o^2)$, where (x^1, y^1) and (x^2, y^2) are the coordinates of corner points of the bounding box. The relative spatial feature is composed of the normalized union box location b_{ul}, relative size b_{sl}, and relative location b_{rl} since they imply potential categories of the relationships between subjects and objects:

$$b_{ul} = (\frac{x_u^1}{w}, \frac{y_u^1}{h}, \frac{x_u^2}{w}, \frac{y_u^2}{h}, \frac{x_u^1 + x_u^2}{2w}, \frac{y_u^1 + y_u^2}{2h}, \frac{w_u}{w}, \frac{h_u}{h}), \tag{4}$$

$$b_{sl} = (\log(\frac{w_s}{w_o}), \log(\frac{h_s}{h_o}), \log(\frac{w_o}{w_s}), \log(\frac{h_o}{h_s})), \tag{5}$$

$$b_{rl} = (\frac{x_s^1 - x_o^1}{w_o}, \frac{y_s^1 - y_o^1}{h_o}, \frac{x_s^2 - x_o^2}{w_o}, \frac{y_s^2 - x_o^2}{h_o},$$
$$\frac{x_o^1 - x_s^1}{w_s}, \frac{y_o^1 - y_s^1}{h_s}, \frac{x_o^2 - x_s^2}{w_s}, \frac{y_o^2 - y_s^2}{h_s}), \tag{6}$$

where (w, h), (w_s, h_s), and (w_o, h_o) denote the widths and heights of image, subject, and object, respectively. (x_u^1, y_u^1) and (x_u^2, y_u^2) are the coordinates of the corner points of the union bounding box of subject b_s and object b_o. We concatenate these features and obtain the relative spatial feature $b_{s,o} = [b_{ul}, b_{sl}, b_{rl}]$.

The relation encoder aims to obtain the refined predicate feature z_j:

$$\{z_j\}_{j=1,\dots,N \times (N-1)} = Enc_{rel}(\{x'_j\}_{j=1,\dots,N \times (N-1)}), \tag{7}$$

where Enc_{rel} denotes the contextual module encoding the relation context. Here we apply a two-layer Transformer as the relation encoder. Finally, we decode entity class logits e_i and predicate

class logits r_j through two fully-connected layers:

$$e_i = FC_{obj}(x_i), \tag{8}$$

$$r_j = FC_{rel}(z_j), \tag{9}$$

where $FC_{obj}(\cdot)$ and $FC_{rel}(\cdot)$ denote the fully-connected layers.

Our CEN fuses the contextual feature with object features and models the relationships among entities, which shares a similar framework with the Object-centric Feature Alignment Module [50]. However, the key contribution of CEN lies in the relative spatial relationship modeling. It explicitly computes the relative spatial features between subjects and objects, while directly taking the spatial feature, *i.e.* bounding box, or implicitly modeling their spatial relationships through ROI features like OFAM [50] is insufficient for generating complicated unseen triplet compositions.

3.3 Triplet Calibration Loss

Existing SGG methods [24, 45, 58] typically utilize cross-entropy loss on relations to optimize their SGG models. However, optimizing cross-entropy loss on scene graphs consisting of a number of dominant triplets and unseen triplets is prone to suppressing the probabilities of unseen triplets and making the model biased towards these dominant seen triplets. In other words, given the fixed subject and object entities, these models will predict high probabilities for high-frequent seen triplets, obstructing the compositional generalization.

3.3.1 Unseen Triplet Calibration. To address these deficiencies, we first devise a calibration loss on unseen samples to resist cross-entropy constraints during training:

$$\mathcal{L}_{cal}(r_{s,o}) = -\log\left(\sum_{(s,c,o)\in C^u_{tpt}} \frac{\exp(r^c_{s,o})}{\sum_{c'\in C_p}\exp(r^{c'}_{s,o})}\right), \tag{10}$$

where C^u_{tpt} denotes the unseen triplet set, $r^c_{s,o}$ is the logit that the relationship category between the subject s and the object o belongs to class c, which is the c-th element in decoded $r_{s,o}$. Minimization of $\mathcal{L}_{cal}$ promotes these unseen samples to have a non-zero probability during training and improves the generalization ability of the SGG model toward unseen triplets. But for those predicates that are correctly annotated, very small logits on unseen triplets are capable of incurring a huge loss with the amplification of the function $log(\cdot)$ and affecting the correct label, which is not desired. Here we apply a hard margin to reduce the effect of $\mathcal{L}_{cal}$ on the annotated category:

$$\begin{aligned} \mathcal{L}^m_{cal}(r_{s,o}) &= \mathcal{L}_{cal}(r_{s,o} + \mathbb{I}_m(s,:,o)) \\ &= -\log\left(\sum_{(s,c,o)\in C^u_{tpt}} \frac{\exp(r^c_{s,o}+1)}{\sum_{c'\in C^s_{tpt}}\exp(r^{c'}_{s,o}-1) + \sum_{c''\in C^u_{tpt}}\exp(r^{c''}_{s,o}+1)}\right), \end{aligned} \tag{11}$$

where C^s_{tpt} denotes the seen triplets set, and $\mathbb{I}_m(s,:,o) \in \mathbb{R}^{|C_p|}$ is a margin vector. When (s,c,o) is an unseen composition, the value of $\mathbb{I}_m(s,c,o)$ is 1. Otherwise, it is -1. For those correctly labeled as background triplets, hard margins mitigate the problem of model training instability due to the large losses incurred by their small logits. For those unseen triplets incorrectly labeled as background, the calibration loss works inversely with the cross-entropy loss, reducing the constraint on unseen samples and generating reasonable confidence to the unseen triplets during inference.

3.3.2 Seen Triplet Calibration. The predicate distribution is diverse when given different subjects and objects, and it may coincide with the distribution of predicates in the whole dataset or maybe the opposite. Previous works [8, 58] consider the predicate distribution from the coarse granularity of the dataset rather than from the fine granularity of the subject-object combination, which weakens the ability of the model to represent various triplets. We propose a fine-grained method

for calibrating seen samples that collaborates with unseen calibration mitigating the bias towards frequently seen triplets while enhancing the attention on rare and diverse compositions. We consider adjusting the margins of seen triplets:

$$
\begin{aligned}
\mathcal{L}_{cal}^{m,\alpha}(\boldsymbol{r}_{s,o}) &= \mathcal{L}_{cal}(\boldsymbol{r}_{s,o} + \mathbb{I}_m^\alpha(s,:,o)) \\
&= -\log \Big(\sum_{(s,c,o)\in C_{tpt}^u} \frac{\exp(r_{s,o}^c + 1)}{\sum_{c'\in C_{tpt}^s} \exp(r_{s,o}^{c'} - \alpha_{s,c',o}) + \sum_{c''\in C_{tpt}^u} \exp(r_{s,o}^{c''} + 1)} \Big),
\end{aligned}
\tag{12}
$$

where $\mathcal{L}_{cal}^{m,\alpha}(\boldsymbol{r}_{s,o})$ replaces the fixed seen triplet margin in $\mathcal{L}_{cal}^m(\boldsymbol{r}_{s,o})$ with a dynamic margin α conditioning on different seen triplets. In Eq. 12, the dynamic margin becomes the coefficient $\exp(-\alpha_{s,c',o})$ of its corresponding term $\exp(r_{s,o}^{c'})$ of the seen triplet. A smaller $\alpha_{s,c',o}$ will impose a larger constraint on its corresponding triplet during optimization. We expect frequent triplets to be subject to large constraints and rare compositions to be subject to relatively small constraints to balance the attention to various compositions. Thus, seen triplet calibration is designed to provide a small α for dominant triplets and a large value for rare triplets:

$$
\alpha_{s,c,o} = \log \Big(\frac{n_{max}}{n_{s,c,o}} \Big) \times \frac{\sum_{i\in C_{tpt}} n_i}{\sum_{j\in C_{tpt}} n_j \log \Big(\frac{n_{max}}{n_j} \Big)},
\tag{13}
$$

where n_{max} and $n_{s,c,o}$ are the counts of triplet with the largest number and triplet (s,c,o), respectively. The first term in Eq. 13 is the margin weight, and the second term is used to normalize these margins. Margin $\alpha_{s,c,o}$ decreases as the count of (s,c,o) increases, which in turn adjusts constraints for the different counts of triplets. In addition, we also add this margin to the cross-entropy loss in seen triplet calibration to reduce the bias of dominant seen triplets:

$$
\mathcal{L}_{ce}^{m,\alpha}(\boldsymbol{r}_{s,o}) = -\log \frac{\exp(r_{s,o}^c - \alpha_{s,c,o})}{\sum_{c'} \exp(r_{s,o}^{c'} - \alpha_{s,c',o})}.
\tag{14}
$$

Combining the unseen and seen triplet calibrations, the final loss of our method is:

$$
\mathcal{L} = \mathcal{L}_{ce}^{m,\alpha} + \lambda \mathcal{L}_{cal}^{m,\alpha},
\tag{15}
$$

where λ is a pre-defined weighting hyper-parameter. During inference, we still calibrate the seen and unseen triplets as:

$$
predicate = \underset{c\in C_p}{\arg\max}\ r_{s,o}^c + \mathbb{I}_m(s,c,o).
\tag{16}
$$

So in the prediction, the model can regularize the discriminative representations of diverse triplets and well-consider unseen samples.

3.4 Unseen Space Reduction

Almost all of the existing SGG approaches treat the entire triplet space as the space of their potential generation results. They assume that all unseen compositions are potentially possible triplets that could exist in the real world and will output a certain probability to these triplets. However, compared to the count of seen triplet categories, the space of the whole composition is very large (which is 1,125,000 vs 29,283 for VG150). Only a small fraction of the unseen triplets could exist in the real world [20]. Moreover, most triplets are unrealistic compositions, *e.g.* <seat-eating-dog>. Hence, reducing the unseen triplet space is necessary to enhance the mining results of unseen triplets during training and inference.

It is observed that there is interchangeability between seen triplets, where subjects, predicates, and objects with similar properties can be replaced to form new compositions (*e.g.* <dog/elephant-walking on-street>, <man-using/holding-phone>, and <man-riding-house/elephant>). We propose

an Unseen Space Reduction Loss (USRL) to reduce the unseen triplet space by starting from this interchangeability among subjects, predicates, and objects. We first fuse any two elements of the triplet and then project them onto the same space as another element to explore the rationality of this triplet:

$$\begin{cases} \boldsymbol{d}_s = \sigma(\boldsymbol{t}_p \star \boldsymbol{t}_o) \odot \sigma(\boldsymbol{w}_s \boldsymbol{t}_s) \\ \boldsymbol{d}_p = \sigma(\boldsymbol{t}_s \star \boldsymbol{t}_o) \odot \sigma(\boldsymbol{w}_p \boldsymbol{t}_p) \\ \boldsymbol{d}_o = \sigma(\boldsymbol{t}_s \star \boldsymbol{t}_p) \odot \sigma(\boldsymbol{w}_o \boldsymbol{t}_o) \end{cases}, \tag{17}$$

where $\boldsymbol{t}_s$, $\boldsymbol{t}_p$, and $\boldsymbol{t}_o$ denote the linguistic embeddings for subject, predicate, and object, respectively. $\boldsymbol{w}_s$, $\boldsymbol{w}_p$, and $\boldsymbol{w}_o$ are trainable parameters, and σ represents the sigmoid function. Then we judge the reasonableness of the triplet d_{usrl} based on the results of the above three aspects:

$$d_{usrl} = \boldsymbol{w}_{usrl}[\boldsymbol{d}_s, \boldsymbol{d}_p, \boldsymbol{d}_o], \tag{18}$$

where $\boldsymbol{w}_{usrl}$ denote parameters that project the concatenated feature to one dimension. The process of learning knowledge from seen triplets and determining the rationality of unseen triplets can be formulated as the Positive-Unlabeled Learning problem [1, 62]. Specifically, based on the annotated compositions that are known to be positive, we classify which ones are negative from a bunch of unlabeled triplets. Here we apply the nnPU [18] to train the unseen space reduction:

$$\mathcal{L}_{usrl} = \frac{\pi}{n_{pos}} \sum_{h_i} \mathcal{L}_{bce}^+(h_i) + \max(0, \quad \frac{1}{n_u} \sum_{h_j} \mathcal{L}_{bce}^-(h_j) - \frac{\pi}{n_{pos}} \sum_{h_i} \mathcal{L}_{bce}^-(h_i)), \tag{19}$$

where $\mathcal{L}_{bce}^+$ and $\mathcal{L}_{bce}^-$ denote the binary cross-entropy functions to calculate the risk of misclassifying the input into negative and positive samples, respectively. n_{pos} and n_u are the counts of positive and unlabeled samples, π represent the fraction of positive samples, and h_i and h_j are the predicted confidences of positive sample and unlabeled sample, respectively. The negative compositions judged by the USRL will narrow the range of C_{tpt}^u, which affects the margin vectors $\mathbb{I}_m^\alpha(s, c, o)$ and $\mathbb{I}_m(s, c, o)$ in Eq. 12 and Eq. 16, respectively.

4 EXPERIMENTS

4.1 Experiment Setting

4.1.1 Dataset. Our experiments for scene graph generation are conducted on the Visual Genome dataset [21]. We follow the most widely used VG150 split [20, 45, 52, 58], which contains the most frequent 150 object categories and 50 relation categories in Visual Genome.

4.1.2 Tasks. We adopt the following three conventional evaluation tasks. 1) Predicate Classification (**PredCls**) aims to predict the predicates of pairwise relationships with ground-truth object bounding boxes and their object categories. 2) Scene Graph Classification (**SGCls**) aims to predict the predicates and object categories with ground-truth object bounding boxes. 3) Scene Graph Detection (**SGDet**) aims to detect object bounding boxes and categories in the image and predict their pairwise relationships.

4.1.3 Evaluation Metric and Protocol. We evaluate SGG methods with image-wise recall evaluation metrics, including Recall@K (R@K) and Zero-Shot Recall@K (zR@K), which are also adopted in previous works [9, 20, 43, 46]. The R@K measures the fraction of ground-truth relationship triplets that appear in the top K most confident triplet predictions in an image, zR@K metric measures the fraction of zero-shot ground-truth relationship triplets that appear in the top K most confident triplet predictions in an image. We average these fractions across images to obtain R@K and zR@K separately. Note that we do not consider test images that do not contain zero-shot triplets for zR@K following [9, 14, 19, 20, 43, 44, 46]. Previous test protocols for zero-shot scene graph generation

are diverse [9, 14, 19, 20, 43, 44, 46], and some of them apply Frequency Bias [58] that greatly degrades zero-shot performance (4.8 versus 20.5 for Motifs [58] under zR@100 and Predcls). Our method cannot be directly compared with all previous approaches. Thus, we unify the test protocols and re-implement previous methods based on their source codes without Frequency Bias for fair comparisons.

We first review the previous test protocols before introducing our unified test protocol. There are several key differences between previous test protocols.

1) **Object Overlap.** The requirement to have overlap between the training objects is proposed by Zellers *et al.* [58]. This means that only relationships of objects that overlap with other objects are allowed to be used as training data. They think objects that do not overlap with other objects typically own low-quality Region of Interest (RoI). This requirement limits the relationships in the training set, resulting in less training data and more unseen samples in the corresponding test set.

2) **Validation Set.** The most widely used split in Visual Genome dataset [21] is VG150 [52]. VG150 is composed of a training set, a validation set, and a test set. The current mainstream [9, 14, 43, 44, 46] zero-shot Scene Graph Generation (SGG) test protocol does not use the validation set. However, some methods [19, 20] apply the validation set, and thus their zero-shot test set does not contain triplets in the validation set.

3) **Frequency Bias.** Frequency Bias [58] is a trick to improve the performance of SGG. However, as shown in the paper, it will significantly damage the performance of the SGG model on unseen triplets.

There mainly exist three evaluation protocols used in previous works.

1) The first zero-shot SGG evaluation protocol[1] is built on top of *neural-motifs*[2]. It does not require the object overlap, applies the validation set, and does not use the Frequency Bias. Methods, *e.g.* [19, 20], using this evaluation protocol apply VGG16 [42] as their backbone.

2) The second zero-shot SGG evaluation protocol[3] is introduced by Tang *et al.* [44]. It requires the object overlap, does not applies the validation set, and uses the Frequency Bias. Methods *e.g.* [44] using the second evaluation protocol apply ResNeXt-101-FPN [27] as their backbone.

3) The third zero-shot SGG evaluation protocol is built on top of *Scene-Graph-Benchmark.pytorch*[3]. It does not require the object overlap, does not apply the validation set, and uses the Frequency Bias. Methods *e.g.* [9, 43, 46] using the third evaluation protocol apply ResNeXt-101-FPN [27] as their backbone.

Our unified zero-shot SGG evaluation protocol is also built on top of *Scene-Graph-Benchmark.pytorch*. To exploit the whole training data, our evaluation protocol does not require object overlap, which is also the option used in most zero-shot SGG methods. It does not apply the validation set and removes the Frequency Bias to improve the performances on unseen triplets.

For comparison with VGG-16, we apply the same evaluation protocol as the comparison methods, and their results are obtained directly from the original papers. For the experiments with ResNeXt-101-FPN, we apply our evaluation protocol and re-implement the comparison methods including IMP [52], VTransE [59], Motifs [58], IMP++ [19], TDE [44], UVTransE [14], BGNN [24], EBM [43], GRAPHN [20], and SSR [46] without Frequency Bias according to their source codes for fair comparisons.

4.1.4 Implementation Details. For fair comparison on VG, we adopt the pre-trained VGG-16 [42] and ResNeXt-101-FPN [27] as the backbones. We follow previous methods and use GloVe [35] with 200 dimensions as the linguistic embedding. We also apply the same post-processing method

---

[1]https://github.com/bknyaz/sgg
[2]https://github.com/rowanz/neural-motifs
[3]https://github.com/KaihuaTang/Scene-Graph-Benchmark.pytorch

Table 1. Performance comparison results of state-of-the-art SGG models on three SGG tasks with graph constraint. "B" denotes the backbone of object detector (Faster R-CNN [39]) used in each SGG model. † denotes that the results are obtained with an unknown evaluation protocol, and thus, may not be directly comparable. "-" represents that the result is not mentioned in the original paper or the method is unavailable in that configuration. The **best** and **second best** results under each setting are marked in **red** and **blue**, respectively.

B	Models	PredCls			SGCls			SGDet		
		zR@20	zR@50	zR@100	zR@20	zR@50	zR@100	zR@20	zR@50	zR@100
VGG-16	IMP [52]	-	14.5	17.2	-	2.5	3.2	-	-	0.9
	Motifs [58]	-	6.5	9.5	-	1.1	1.7	-	-	0.3
	IMP++ [19]	-	18.4	21.5	-	3.4	4.2	-	-	0.8
	GRAPHN [20]	-	19.5	22.4	-	3.8	4.5	-	-	1.1
	T-CAR (ours)	23.1	29.6	32.8	6.4	7.6	8.7	2.4	3.4	4.2
X-101-FPN	IMP [52]	12.3	17.5	19.9	1.2	1.9	2.2	0.1	0.5	0.9
	VTransE [59]	7.0	11.7	15.0	1.1	1.9	2.3	0.5	0.9	1.6
	Motifs [58]	11.8	17.7	20.5	2.6	4.1	5.0	0.9	1.9	2.7
	Motifs+Freq [58]	0.1	2.8	4.8	0.4	0.7	1.1	0.0	0.0	0.2
	IMP++ [19]	12.9	19.2	22.4	2.6	4.2	5.0	0.1	0.4	0.9
	TDE [44]	7.7	12.5	16.4	1.6	2.6	3.5	1.1	2.0	2.6
	UVTransE [14]	10.7	16.5	18.9	2.2	3.3	3.9	0.6	1.2	2.1
	EBM [43]	11.3	16.8	20.0	3.4	5.3	6.2	1.0	2.0	3.0
	BGNN [24]	1.5	3.5	5.2	0.9	1.7	2.2	0.1	0.1	0.3
	SSR(Base) [46]	-	-	-	-	-	-	1.6	2.6	3.6
	SSR(Large) [46]	-	-	-	-	-	-	1.8	2.8	4.2
	NARE† [9]	9.1	13.5	-	4.3	6.2	-	2.2	3.3	-
	T-CAR (ours)	24.5	31.9	34.9	6.9	9.3	10.6	3.2	4.7	6.0

as previous methods [28, 29], *i.e.* relational-NMS, to filter the generated redundant triplets. Our network is optimized by Stochastic Gradient Descent (SGD) with an initial learning rate of 10^{-3}, and the batch size is set as 14. The number of total iterations is 16k, and the learning rate is decayed by the factor of 10 on the $10k^{th}$ iterations. The reduced prediction space accounts for 85% of the total triplet space. The parameters λ and π in Eq. 15 and Eq. 19 are set as 0.01 and 0.03, respectively. Our codes are implemented with PyTorch and 2 NVIDIA GeForce RTX 2080Ti GPUs.

4.2 Comparisons with State-of-the-Art Methods

We compare several state-of-the-art methods on the Visual Genome dataset to demonstrate the effectiveness of our approach. IMP [52], VTransE [59], and Motifs [58] focus on all triplet prediction, so their performance on seen triplets is much better than on unseen triplets. IMP++ [19], TDE [44], UVTransE [14], EBM [43], GRAPHN [20], SSR [46], and NARE [9] aim to generate unseen triplets. BGNN [24] is a state-of-the-art method focusing on unbiased predicate prediction.

We compare BGNN with our method to demonstrate the difference between unbiased scene graph generation and zero-shot scene graph generation, *i.e.* methods that concentrate solely on unbiased scene graph generation do not work well on zero-shot scene graph generation task. It is worth noting that GRAPHN [20] generates visual feature maps to augment the input scene graphs and improve the generalization of its scene graph model on unseen triplets. Here we only compare GRAPHN [20] with VGG-16 backbone since we argue that it is challenging to extend the generative model of GRAPHN from one scale to multiple scales with Feature Pyramid Network (FPN). Due to the limitation of SSR, its comparison is only possible under the SGDet task.

Fig. 4. Absolute zR@100 improvement in PredCls task by T-CAR compared to IMP++ [19] with ResNeXt-101-FPN backbone. The predicate categories are sorted according to their frequency.

Fig. 5. Absolute zR@100 improvement in SGCls task by T-CAR compared to IMP++ [19] with ResNeXt-101-FPN backbone. The predicate categories are sorted according to their frequency.

4.2.1 Comparison with Graph Constraint. Tab. 1 shows the comparison results of our model on the Visual Genome dataset with graph constraint. We have the following observations:

1) Our T-CAR model consistently and significantly outperforms all state-of-the-art methods on all three tasks with two backbones, which achieves a large margin of improvements by 12.5%, 4.2%, and 1.8% on zR@100 for PredCls, SGCls, and SGDet, respectively. Compared with the predicate

Table 2. Performance comparison results of our T-CAR method with state-of-the-art SGG models on three SGG tasks without graph constraint. "B" denotes the backbone of object detector (Faster R-CNN [39]) used in each SGG model. "-" represents that the result is not mentioned in the original paper or the method is unavailable in that configuration.

B	Models	PredCls			SGCls			SGDet		
		zR@20	zR@50	zR@100	zR@20	zR@50	zR@100	zR@20	zR@50	zR@100
X-101-FPN	IMP [52]	14.4	27.5	38.9	1.5	3.8	6.8	0.1	0.5	1.1
	VTransE [59]	8.2	17.9	28.6	1.6	3.9	7.0	0.9	1.7	3.2
	Motif [58]	14.3	27.4	39.7	3.1	6.9	11.2	1.1	2.7	4.5
	IMP++ [19]	14.6	27.3	39.0	3.1	7.5	11.4	0.1	0.4	0.9
	TDE [44]	8.9	17.4	26.7	1.3	4.2	8.3	0.0	0.1	0.4
	EBM [43]	13.8	26.2	38.3	4.1	8.5	13.9	1.3	2.7	4.4
	UVTransE [14]	12.7	25.4	37.3	2.7	5.9	9.8	0.9	2.3	3.9
	BGNN [24]	2.2	7.5	15.6	1.3	3.9	6.8	0.1	0.2	0.5
	SSR(Base) [46]	-	-	-	-	-	-	2.0	4.1	6.0
	SSR(Large) [46]	-	-	-	-	-	-	2.1	4.0	5.9
	NARE [9]	-	-	-	-	-	-	-	-	-
	T-CAR (ours)	**27.4**	**39.8**	**50.8**	**7.8**	**12.4**	**16.7**	**3.6**	**6.2**	**8.7**

Table 3. Performance comparison results of our T-CAR method with state-of-the-art SGG models on three SGG tasks with graph constraint. "B" denotes the backbone of object detector (Faster R-CNN [39]) used in each SGG model. † denotes that the results are obtained with an unknown evaluation protocol, and thus, may not be directly comparable. "-" represents that the result is not mentioned in the original paper or the method is unavailable in that configuration.

B	Models	PredCls		SGCls		SGGen	
		zR@50/100	R@50/100	zR@50/100	R@50/100	zR@50/100	R@50/100
X-101-FPN	IMP [52]	17.5 / 19.9	61.3 / 63.3	1.9 / 2.2	61.3 / 35.3	0.5 / 0.9	25.7 / 31.2
	VTransE [59]	11.7 / 15.0	58.2 / 62.6	1.9 / 2.3	33.4 / 35.6	0.9 / 1.6	27.2 / 31.5
	Motif [58]	17.7 / 20.5	**65.0 / 66.9**	4.1 / 5.0	39.1 / 39.9	1.9 / 2.7	**32.6 / 37.0**
	IMP++ [19]	19.2 / 22.4	62.1 / 64.3	4.3 / 5.1	40.0 / 40.8	0.4 / 0.9	21.1 / 27.4
	TDE [44]	12.5 / 16.4	45.7 / 51.1	2.6 / 3.5	28.0 / 30.5	2.0 / 2.6	16.7 / 20.3
	EBM [43]	16.8 / 20.0	64.5 / 66.5	5.3 / 6.2	**43.5 / 44.7**	2.0 / 3.0	30.3 / 34.6
	UVTransE [14]	16.5 / 18.9	64.7 / 66.4	3.3 / 3.9	37.9 / 38.8	1.2 / 2.1	31.9 / 36.1
	BGNN [24]	3.5 / 5.2	58.1 / 60.9	1.7 / 2.2	36.2 / 37.4	0.1 / 0.3	25.9 / 31.1
	SSR(Base) [46]	- / -	- / -	- / -	- / -	2.6 / 3.6	23.3 / 26.5
	SSR(Large) [46]	- / -	- / -	- / -	- / -	2.8 / 4.2	23.7 / 27.3
	NARE† [9]	13.5 / -	47.6 / 52.0	6.2 / -	32.8 / 35.8	3.3 / -	19.0 / 21.0
	T-CAR (ours)	**31.9 / 34.9**	60.0 / 63.0	**9.3 / 10.6**	40.4 / 42.0	**4.7 / 6.0**	28.5 / 32.9

granularity-based debiasing method [44] and unlabeled sample mining method [9], our triplet granularity-based T-CAR significantly improves the compositional generalization ability, which indicates that it is better to solve the zero-shot SGG problem at the triplet granularity. Our method surpasses the approach Structured Sparse R-CNN (SSR) [46] that directly predicts triplets and GAN-based model GRAPHN [20], demonstrating the effectiveness of excavating unseen triplets in the training set, which greatly reduces the seen triplet bias.

2) Compared with methods aiming to address the problem of imbalanced predicates, *i.e.* BGNN [24], we witness that its performance is much lower than the SGG baselines, namely IMP [52],

Table 4. Ablation studies on each component of T-CAR. We use the same object detection backbone as in [24].

Module			SGCls			PredCls		
CEN	TCL	USRL	zR@20	zR@50	zR@100	zR@20	zR@50	zR@100
✗	✗	✗	1.7	3.1	3.9	8.0	13.2	16.6
✓	✗	✗	3.4	5.3	6.4	14.7	21.0	24.3
✗	✓	✗	3.9	5.9	7.3	16.5	22.6	26.7
✓	✓	✗	6.3	8.6	9.8	24.4	31.0	34.1
✗	✓	✓	4.1	6.3	7.9	16.1	23.1	27.6
✓	✓	✓	**6.9**	**9.3**	**10.6**	**24.5**	**31.9**	**34.9**

Table 5. Ablation studies on the margin in TCL.

Module		SGCls			PredCls		
MU	MCE	zR@20	zR@50	zR@100	zR@20	zR@50	zR@100
✗	✗	6.1	8.8	10.0	23.0	30.3	34.0
✓	✗	6.3	8.8	10.0	22.7	30.5	34.2
✗	✓	6.5	9.0	10.1	24.5	31.6	34.8
✓	✓	**6.9**	**9.3**	**10.6**	**24.5**	**31.9**	**34.9**

Motifs [58], and UVTransE [14], indicating that the problem of zero-shot SGG is not equivalent to the issue of imbalanced predicates.

3) Deleting Frequent Bias on Motifs severely boosts its performance in zero-shot Recall. It demonstrates that the Frequent Bias module forces the models to bias toward seen triplets and decays their generalization capabilities.

4.2.2 Improvements on Each Predicate Category. As shown in Fig. 4, we further investigate the improvement of our method over the IMP++ [19] on each predicate category to show its performance on different predicates. We find that T-CAR significantly improves the performance in head, body, and tail predicate categories. It is fundamentally different from approaches solving imbalanced predicates that typically boost the performance of tail classes and weaken the performance of the head ones.

4.2.3 Comparison without Graph Constraint. The setting of *without graph constraint* is proposed by Zellers [58]. It allows the output scene graph to contain multiple edges between the subject and object, as opposed to *graph constraint*. Higher recall performance can usually be obtained without the graph constraint since the model is allowed to have multiple guesses for challenging relations. To extensively analyze the compositional generalization ability of our T-CAR method, we also report the comparison results without graph constraints. As shown in Tab. 2, with multiple guesses for challenging relations, our T-CAR method consistently outperforms state-of-the-art methods on all three tasks without graph constraint, demonstrating the superior compositional generalization ability of our method.

4.2.4 Comparison with Relationship Recall. The full results of Relationship Recall with graph constraint, including both conventional Recall@K and the adopted Zero-Shot Recall@K, are shown in Tab. 3.

IMP [52], VTransE [59], and Motifs [58] focus on all triplet predictions and achieve better results on Recall than other methods. We can observe that our method achieves state-of-the-art performance

Table 6. Ablation studies on the reduction rate.

	Reduction Rate							
	0%	50%	60%	70%	80%	85%	90%	100%
zR@20	6.3	6.4	6.6	6.5	6.7	**6.9**	6.6	3.3
zR@50	8.6	8.7	8.8	8.8	9.2	**9.3**	8.7	5.0
zR@100	9.8	10.0	10.1	10.0	10.5	**10.6**	10.3	6.1

Table 7. Ablation studies on the unseen space reduction method.

Method	AUC	Recall	Precision	F1-score
Random	49.3	51.9	0.5	1.0
BiLSTM	91.0	72.6	3.3	6.1
Ours	**93.0**	**73.8**	**4.2**	**7.9**

on unseen triplets with acceptable performance degradation on seen samples. Compared with the recently proposed state-of-the-art approaches, *i.e.* SSR [46] and NARE [9], our T-CAR model not only owns better compositional generalization ability but also significantly outperforms them on the seen samples.

4.3 Ablation Studies

4.3.1 Model Components. We conduct an ablation study to evaluate the importance of each component in our T-CAR, *i.e.* Contextual Encoding Network (CEN), Triplet Calibration Loss (TCL), and Unseen Space Reduce Loss (USRL). The results are shown in Tab. 4. Specifically, we remove all modules in T-CAR and use a baseline without explicit relation feature refinement. This baseline predicts predicates with visual features of pairwise entities, and its performance is lower than any other variants of T-CAR. Then we add these proposed components to the baseline method.

As shown in Tab. 4, all modules promote performance, and the best performance is achieved when all modules are involved. We observe that the TCL improves the CEN and achieves 7.9 and 32.0 on SGCls and PredCls tasks, demonstrating that reducing the seen triplet bias and relaxing constraints on potentially unseen samples can facilitate the zero-shot SGG. Compared with CEN+TCL, we witness an obvious performance gain with USRL. Note that the USRL is designed to reduce the search space of unseen triplets, so its relative boost on SGCls is greater than that of PredCls. Moreover, applying CEN alone is able to achieve similar performance to state-of-the-art methods [19, 43]. It verifies that more robust position encoding can alleviate the seen triplet bias, leading to more accurate predictions.

4.3.2 Margin of Calibration. We also evaluate the effectiveness of weighted seen triplet margins on TCL and report the results in Tab. 5. We first set the margins of seen triplets in TCL equal to 1 as the baseline. Then margin constraints on seen triplets and unseen triplets calibration are added, named MU and MCE, respectively. The results show that changing the margins on seen samples by their occurrence frequency in both cross-entropy and unseen sample mining losses can improve performance. Frequent triplets in scene graph do affect the generalization of unseen samples. Best performance is achieved when both MU and MCE are engaged.

4.3.3 Unseen Space Reduction Loss. In Tab. 7, we examine the effect of the interchangeability module in USRL on the reduction of unseen triplet space. Unseen triplets in the VG test set are treated as test samples. We apply the ranking metric AUC and commonly used evaluation metrics for binary classification, *i.e.* Recall, Precision, and F1, to evaluate model performance. The precision

Table 8. Ablation studies on the initialization of features of CEN and T-CAR. The w/o P and w L denote initialize features without relative positional encoding and with the linguistic feature, respectively.

Model	SGCls			PredCls		
	zR@20	zR@50	zR@100	zR@20	zR@50	zR@100
CEN w/o P	3.2	5.2	6.3	14.0	20.8	24.1
CEN w L	3.2	5.2	6.4	13.0	19.6	23.1
CEN	3.4	5.3	6.4	14.7	21.0	24.3
T-CAR w/o P	6.0	8.5	9.9	**24.8**	31.4	34.7
T-CAR w L	6.5	8.8	10.0	24.4	31.5	34.8
T-CAR	**6.9**	**9.3**	**10.6**	24.5	**31.9**	**34.9**

Table 9. Ablation studies on the hyper-parameter λ in TCL.

λ	SGCls			PredCls		
	zR@20	zR@50	zR@100	zR@20	zR@50	zR@100
0.001	6.3	8.5	9.8	23.9	31.0	34.3
0.01	**6.9**	**9.3**	**10.6**	**24.5**	**31.9**	**34.9**
0.1	5.3	7.4	9.0	23.4	29.3	33.3
1.0	2.2	4.2	5.8	10.4	16.5	22.3

is very low since the entire space is extremely large, and the unseen triplets in the VG test set are not equivalent to all positive unseen samples.

"Random" in Tab. 7 implies that we apply a randomly initialized MLP to assign the triplets of the unseen space as reasonable or unreasonable without the knowledge of seen realistic triplets. We perform random strategy five times and report their average results. In addition, we also apply a two-layer BiLSTM on triplets as the baseline method. Compared with the "random", it is observed that BiLSTM can learn from seen triplets and judge the reasonableness of unknown triplets. BiLSTM [13] is able to rank the reasonableness of triplets and achieves good performance. Compared with BiLSTM, our unseen space reduction improves the performance of all metrics, especially in AUC. It indicates that our method performs better in ranking the plausibility of triplets. We own this advantage to the alternative mining of triplets in subject, predicate, and object.

In Tab. 6, we also explore the impact of reduction rate in USRL for SGCls task. Unseen triplet space gradually decreases as the reduction rate increases, and the model shifts its attention to excavating reasonable triplets. But when the reduction rate is larger than a certain degree, it will inevitably misclassify some of the unseen and reasonable triplets. Thus the performance of T-CAR behaves as an increase followed by a decrease, and the best results are achieved when the reduction rate is set as 85%.

4.3.4 Ablation Study on Hyper-Parameter in TCL. λ in Eq. 15 controls the attention of the model to unlabeled and unseen samples, which is important for unseen triplets excavation. We study the effect of this hyper-parameter on SGCls and PredCls tasks with the ResNeXt-101-FPN backbone. As shown in Tab. 9, our model gradually increases its attention to the unseen samples as the λ value increases. When the λ reaches a certain value, the attention to unseen samples affects the learning on seen samples, which leads to the weakened ability to represent diverse triplets and decreased performance on unseen samples. The best performance is achieved when λ equals 0.01.

4.3.5 Ablation Study on Initialization of Features. We verify the effectiveness of initialized features in CEN and T-CAR on compositional generalization ability with the ResNeXt-101-FPN backbone. As
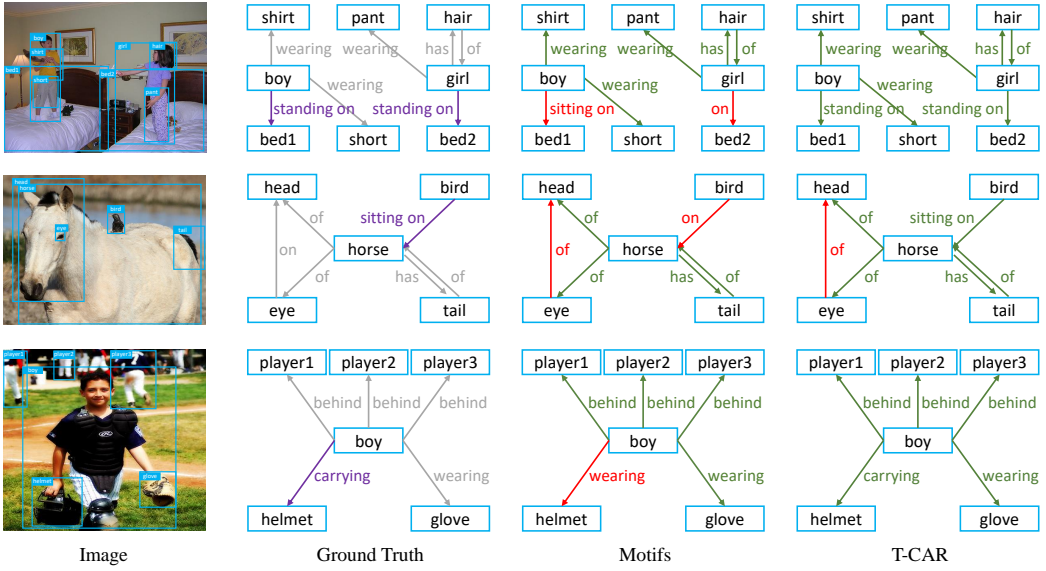
Fig. 6. Qualitative comparisons between our T-CAR and Motifs [58] in the PredCls setting. The purple color indicates the unseen triplets in test images. The green color denotes the correctly classified triplets, and the red suggests the misclassified triplets. Best viewed in color.

Table 10. Analysis on the knowledge transfer. We add the recently proposed distribution knowledge transfer loss (D-TRS) in TN-ZSTAD [60] in our T-CAR, which further improves the performance of our model.

Model	SGCls			PredCls		
	zR@20	zR@50	zR@100	zR@20	zR@50	zR@100
T-CAR	6.9	**9.3**	**10.6**	24.5	**31.9**	34.9
T-CAR + D-TRS [60]	**7.2**	9.2	**10.6**	**25.3**	31.6	**35.1**

shown in Tab. 8, both the T-CAR model and CEN have better compositional generalization ability without linguistic features and with relative positional encoding. CEN with linguistic features on the PredCls task encounters a severe drop in unseen performance, indicating that the subject and object priors introduced by linguistic features do affect the compositional generalization ability.

4.4 Analysis on the Knowledge Transfer

Our triplet calibration loss serves the same purpose as the distribution transfer loss (D-TRS) in TN-ZSTAD [60], *i.e.*, regularizing the model to transfer knowledge from seen to unseen. D-TRS takes the semantic similarity of labels to prevent the classifier from over-confidence toward seen classes and to match the predicted probability distribution of unseen classes. Such semantic similarity between triplet labels can guide the model to mine the unseen triplets that are most likely to be misclassified as background, which can also collaborate with our method.

We follow D-TRS and take the CLIP [37] text embedding to calculate the semantic similarity between triplet labels. For instance, given a triplet label <Subject, Predicate, Object>, we generate the corresponding description in the format of 'A photo of a/an [Subject] [Predicate] a/an [Object]'. Then, we generate the text embedding for each triplet label through the pre-trained CLIP text encoder. Finally, D-TRS is applied to obtain the similarity between labels and collaborate with our

Table 11. Model size and speed comparisons for SGDet. "N/A" denotes that the result is not available due to the limited GPU memory.

| Backbone | Models | Training | | | | | Inference |
		#Params (MB)	Iterations	Batch Size	Times (h)	Sec/image	FPS
X-101-FPN	IMP [52]	1,176	24,000	12	14.4	0.09	5.6
	VTransE [59]	1,170	28,000	12	18.3	0.10	5.6
	Motifs [58]	1,405	28,000	12	21.8	0.13	4.2
	IMP++ [19]	1,176	28,000	12	19.5	0.12	5.4
	TDE [44]	1,414	28,000	12	21.4	0.13	3.1
	UVTransE [14]	1,217	20,000	12	13.3	0.08	5.3
	EBM [43]	1,419	200,000	4	362.0	1.56	2.2
	BGNN [24]	1,304	100,000	6	105.1	0.62	2.5
	SSR(Base) [46]	1,045	320,000	2	127.1	0.71	11.8
	SSR(Large) [46]	1,049	N/A	N/A	N/A	N/A	3.4
	T-CAR (ours)	1,268	16,000	14	11.6	0.12	3.9

method. As shown in Tab. 10, the performance of our T-CAR method is further improved by D-TRS. Note that D-TRS boosts our method for zR@20 more than zR@100. It indicates that D-TRS gives much confidence to the unseen triplets mislabeled as backgrounds through the similarity between triplet labels in training.

4.5 Qualitative Evaluation

We visualize the scene graphs generated by our T-CAR and compare them with Motifs [58] to show the importance of the zero-shot SGG. Results are shown in Fig. 6. As seen from the first row, Motifs tends to predict the frequently seen relationships. It predicts the predicate between "girl" and "bed2" as "on" instead of the more accurate "standing on", and misclassifies the relationships of "boy" and "bed1" as the frequent predicate "sitting on". T-CAR alleviates the seen triplets bias from frequent compositions and generates the correct category. As seen in the second row, T-CAR is equally effective in predicting the anthropomorphic actions made by animals. Finally, in the third row of Fig. 6, it can be observed that it is easy to exclude the predicate "wearing" and consider "carrying" based on the relative spatial information between "boy" and "helmet" by T-CAR. To sum up, our T-CAR makes better predictions than Motifs. We owe this performance gain of T-CAR to the explicit modeling of relative spatial features that alleviates the seen triplet bias in contextual encoding network.

4.6 Model Size and Speed

Experiments are also conducted to analyze the model size and speed. Though scene graphs are powerful, it is time-consuming to perform SGG on the large-scale dataset. We include several previous works and run their codes under the same settings to analyze the model efficiency. Our experiments are conducted on two NVIDIA GeForce GTX 2080 Ti GPUs and an Intel Xeon E5-2650 v4 CPU. We set the training batch size to the maximum under the GPU memory limit, and the inference batch size is fixed to 2. Training SSR (Large) encounters the out-of-memory error due to its query numbers (300 for SSR (base) and 800 for SSR (Large)). Therefore, we only report its inference speed and model size. The results are shown in Table 11. It is worth noting that our method requires only 16,000 iterations to converge, which is faster than all the other compared models. Our method also has fast inference speed and moderate parameter size. Overall, T-CAR performs well in terms of efficiency and is feasible for applications on large-scale datasets.

5 CONCLUSION

This paper introduces a Triplet Calibration and Reduction framework for zero-shot scene graph generation. It consists of a contextual encoding network, a triplet calibration loss, and an unseen space reduction loss. The contextual encoding network is based upon an entity encoder and a relation encoder. It explicitly models the relative spatial features between subjects and objects to alleviate seen triplet bias. The triplet calibration loss regularizes the representation of diverse triplets and mines the unseen triplets that are incorrectly annotated as background. Unseen Space Reduction Loss is built based on the interchangeability between seen triplets to reduce unreasonable triplets in unseen space. We also propose a new test protocol to facilitate a fair comparison of zero-shot SGG methods. Besides, both qualitative and quantitative evaluations are conducted to verify the effectiveness of the proposed method, and the results show that our method significantly outperforms the state-of-the-art zero-shot SGG methods on zero-shot triplets. In the future, we will explore leveraging external knowledge of large-scale pre-trained vision-language models, *e.g.* CLIP [37], to filter unreasonable triplets in the unseen space.

ACKNOWLEDGEMENT

REFERENCES

[1] Jessa Bekker and Jesse Davis. 2020. Learning from positive and unlabeled data: A survey. *Machine Learning* 109, 4 (2020), 719–760.

[2] Hedi Ben-Younes, Remi Cadene, Nicolas Thome, and Matthieu Cord. 2019. Block: Bilinear Superdiagonal Fusion for Visual Question Answering and Visual Relationship Detection. In *Proc. AAAI*. 8102–8109.

[3] Xiaojun Chang, Pengzhen Ren, Pengfei Xu, Zhihui Li, Xiaojiang Chen, and Alex Hauptmann. 2023. A Comprehensive Survey of Scene Graphs: Generation and Application. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 1 (2023), 1–26.

[4] Chao Chen, Yibing Zhan, Baosheng Yu, Liu Liu, Yong Luo, and Bo Du. 2022. Resistance Training Using Prior Bias: Toward Unbiased Scene Graph Generation. In *Proc. AAAI*. 212–220.

[5] Shizhe Chen, Qin Jin, Peng Wang, and Qi Wu. 2020. Say as You Wish: Fine-grained Control of Image Caption Generation with Abstract Scene Graphs. In *Proc. CVPR*. IEEE, 9962–9971.

[6] Yixin Chen, Siyuan Huang, Tao Yuan, Siyuan Qi, Yixin Zhu, and Song-Chun Zhu. 2019. Holistic++ scene understanding: Single-view 3d holistic scene parsing and human pose estimation with human-object interaction and physical commonsense. In *Proc. CVPR*. IEEE, 8648–8657.

[7] Helisa Dhamo, Azade Farshad, Iro Laina, Nassir Navab, Gregory D Hager, Federico Tombari, and Christian Rupprecht. 2020. Semantic Image Manipulation Using Scene Graphs. In *Proc. CVPR*. IEEE, 5213–5222.

[8] Xingning Dong, Tian Gan, Xuemeng Song, Jianlong Wu, Yuan Cheng, and Liqiang Nie. 2022. Stacked Hybrid-Attention and Group Collaborative Learning for Unbiased Scene Graph Generation. In *Proc. CVPR*. IEEE, 19427–19436.

[9] Arushi Goel, Basura Fernando, Frank Keller, and Hakan Bilen. 2022. Not All Relations are Equal: Mining Informative Labels for Scene Graph Generation. In *Proc. CVPR*. IEEE, 15596–15606.

[10] Jiuxiang Gu, Shafiq Joty, Jianfei Cai, Handong Zhao, Xu Yang, and Gang Wang. 2019. Unpaired Image Captioning via Scene Graph Alignments. In *Proc. ICCV*. IEEE, 10323–10332.

[11] Yutian Guo, Jingjing Chen, Hao Zhang, and Yu-Gang Jiang. 2020. Visual Relations Augmented Cross-modal Retrieval. In *Proc. ICMR*. ACM, 9–15.

[12] W. Hamilton, R. Ying, and J. Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *Advances in Neural Information Processing Systems*. MIT Press, 1025–1035.

[13] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[14] Zih-Siou Hung, Arun Mallya, Svetlana Lazebnik, and . 2021. Contextual Translation Embedding for Visual Relationship Detection and Scene Graph Generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43, 11 (2021), 3820–3832.

[15] Dat Huynh and Ehsan Elhamifar. 2020. Compositional Zero-Shot Learning via Fine-Grained Dense Feature Composition. In *Advances in Neural Information Processing Systems*, Vol. 33. MIT Press, 19849–19860.

[16] Phillip Isola, Joseph J Lim, and Edward H Adelson. 2015. Discovering States and Transformations in Image Collections. In *Proc. CVPR*. IEEE, 1383–1391.

[17] Shyamgopal Karthik, Massimiliano Mancini, and Zeynep Akata. 2022. KG-SP: Knowledge Guided Simple Primitives for Open World Compositional Zero-Shot Learning. In *Proc. CVPR*. IEEE, 9336–9345.

[18] Ryuichi Kiryo, Gang Niu, Marthinus C Du Plessis, and Masashi Sugiyama. 2017. Positive-Unlabeled Learning with Non-Negative Risk Estimator. In *Advances in Neural Information Processing Systems*, Vol. 30. MIT Press.

[19] Boris Knyazev, Harm de Vries, Cătălina Cangea, Graham W Taylor, Aaron Courville, and Eugene Belilovsky. 2020. Graph Density-aware Losses for Novel Compositions in Scene Graph Generation. In *Proc. BMVC*. BMVA, 1–14.

[20] Boris Knyazev, Harm de Vries, Cătălina Cangea, Graham W Taylor, Aaron Courville, and Eugene Belilovsky. 2021. Generative Compositional Augmentations for Scene Graph Prediction. In *Proc. ICCV*. IEEE, 15827–15837.

[21] 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.

[22] Lin Li, Long Chen, Yifeng Huang, Zhimeng Zhang, Songyang Zhang, and Jun Xiao. 2022. The Devil is in the Labels: Noisy Label Correction for Robust Scene Graph Generation. In *Proc. CVPR*. IEEE, 18869–18878.

[23] Qun Li, Fu Xiao, Bir Bhanu, Biyun Sheng, and Richang Hong. 2022. Inner Knowledge-Based Img2Doc Scheme for Visual Question Answering. *ACM Trans. Multimedia Comput. Commun. Appl.* 18, 3, Article 76 (mar 2022), 21 pages.

[24] Rongjie Li, Songyang Zhang, Bo Wan, and Xuming He. 2021. Bipartite Graph Network with Adaptive Message Passing for Unbiased Scene Graph Generation. In *Proc. CVPR*. IEEE, 11109–11119.

[25] Xiangyu Li, Xu Yang, Kun Wei, Cheng Deng, and Muli Yang. 2022. Siamese Contrastive Embedding Network for Compositional Zero-Shot Learning. In *Proc. CVPR*. IEEE, 9326–9335.

[26] Kongming Liang, Yuhong Guo, Hong Chang, and Xilin Chen. 2018. Visual Relationship Detection with Deep Structural Ranking. In *Proc. AAAI*.

[27] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature Pyramid Networks for Object Detection. In *Proc. CVPR*. IEEE, 2117–2125.

[28] Xin Lin, Changxing Ding, Yibing Zhan, Zijian Li, and Dacheng Tao. 2022. HL-Net: Heterophily Learning Network for Scene Graph Generation. In *Proc. CVPR*. IEEE, 19476–19485.

[29] Xin Lin, Changxing Ding, Jing Zhang, Yibing Zhan, and Dacheng Tao. 2022. RU-Net: Regularized Unrolling Network for Scene Graph Generation. In *Proc. CVPR*. IEEE, 19457–19466.

[30] Yibing Liu, Yangyang Guo, Jianhua Yin, Xuemeng Song, Weifeng Liu, Liqiang Nie, and Min Zhang. 2022. Answer Questions with Right Image Regions: A Visual Attention Regularization Approach. *ACM Trans. Multimedia Comput. Commun. Appl.* 18, 4, Article 93 (mar 2022), 18 pages.

[31] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. 2016. Visual Relationship Detection with Language Priors. In *Proc. ECCV*. Springer, 852–869.

[32] Yichao Lu, Himanshu Rai, Jason Chang, Boris Knyazev, Guangwei Yu, Shashank Shekhar, Graham W Taylor, and Maksims Volkovs. 2021. Context-aware Scene Graph Generation with Seq2Seq Transformers. In *Proc. ICCV*. IEEE, 15931–15941.

[33] Xinyu Lyu, Lianli Gao, Yuyu Guo, Zhou Zhao, Hao Huang, Heng Tao Shen, and Jingkuan Song. 2022. Fine-Grained Predicates Learning for Scene Graph Generation. In *Proc. CVPR*. IEEE, 19467–19475.

[34] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. 2021. Open World Compositional Zero-Shot Learning. In *Proc. CVPR*. IEEE, 5222–5230.

[35] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proc. EMNLP*. ACL, 1532–1543.

[36] Farhad Pourpanah, Moloud Abdar, Yuxuan Luo, Xinlei Zhou, Ran Wang, Chee Peng Lim, Xi-Zhao Wang, and QM Jonathan Wu. 2022. A review of generalized zero-shot learning methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022).

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proc. ICML*, Vol. 139. PMLR, 8748–8763.

[38] Yunbo Rao, Ziqiang Yang, Shaoning Zeng, Qifei Wang, and Jiansu Pu. 2022. Dual Projective Zero-Shot Learning Using Text Descriptions. *ACM Trans. Multimedia Comput. Commun. Appl.* (jul 2022).

[39] 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*, Vol. 28. MIT Press.

[40] Sahand Sharifzadeh, Sina Moayed Baharlou, Martin Schmitt, Hinrich Schütze, and Volker Tresp. 2022. Improving scene graph classification by exploiting knowledge from texts. In *Proc. AAAI*, Vol. 36. 2189–2197.

[41] Jiaxin Shi, Hanwang Zhang, and Juanzi Li. 2019. Explainable and Explicit Visual Reasoning over Scene Graphs. In *Proc. CVPR*. IEEE, 8376–8384.

[42] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-scale Image Recognition. In *Proc. ICLR*.

[43] Mohammed Suhail, Abhay Mittal, Behjat Siddiquie, Chris Broaddus, Jayan Eledath, Gerard Medioni, and Leonid Sigal. 2021. Energy-based Learning for Scene Graph Generation. In *Proc. CVPR*. IEEE, 13936–13945.

[44] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. 2020. Unbiased Scene Graph Generation From Biased Training. In *Proc. CVPR*. IEEE, 3716–3725.

[45] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. 2019. Learning to Compose Dynamic Tree Structures for Visual Contexts. In *Proc. CVPR*. IEEE, 6619–6628.

[46] Yao Teng and Limin Wang. 2022. Structured Sparse R-CNN for Direct Scene Graph Generation. In *Proc. CVPR*. IEEE, 19437–19446.

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Advances in Neural Information Processing Systems*, Vol. 30. MIT Press.

[48] Wenbin Wang, Ruiping Wang, Shiguang Shan, and Xilin Chen. 2020. Sketching Image Gist: Human-Mimetic Hierarchical Scene Graph Generation. In *Proc. ECCV*. Springer, 222–239.

[49] Wenguan Wang, Tianfei Zhou, Siyuan Qi, Jianbing Shen, and Song-Chun Zhu. 2021. Hierarchical human semantic parsing with comprehensive part-relation modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 7 (2021), 3508–3522.

[50] Xiaohan Wang, Linchao Zhu, Yu Wu, and Yi Yang. 2021. Symbiotic Attention for Egocentric Action Recognition with Object-centric Alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021), 1–13.

[51] Hanjie Wu, Yongtuo Liu, Hongmin Cai, and Shengfeng He. 2022. Learning Transferable Perturbations for Image Captioning. *ACM Trans. Multimedia Comput. Commun. Appl.* 18, 2, Article 57 (feb 2022), 18 pages.

[52] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. 2017. Scene Graph Generation by Iterative Message Passing. In *Proc. CVPR*. IEEE, 5410–5419.

[53] Xing Xu, Jialin Tian, Kaiyi Lin, Huimin Lu, Jie Shao, and Heng Tao Shen. 2021. Zero-Shot Cross-Modal Retrieval by Assembling AutoEncoder and Generative Adversarial Network. *ACM Trans. Multimedia Comput. Commun. Appl.* 17, 1s, Article 3 (mar 2021), 17 pages.

[54] Rintaro Yanagi, Ren Togo, Takahiro Ogawa, and Miki Haseyama. 2022. Interactive Re-Ranking via Object Entropy-Guided Question Answering for Cross-Modal Image Retrieval. *ACM Trans. Multimedia Comput. Commun. Appl.* 18, 3, Article 68 (mar 2022), 17 pages.

[55] Hongchuan Yu, Mengqing Huang, and Jian J. Zhang. 2022. Domain Adaptation Problem in Sketch Based Image Retrieval. *ACM Trans. Multimedia Comput. Commun. Appl.* (oct 2022).

[56] Jin Yuan, Lei Zhang, Songrui Guo, Yi Xiao, and Zhiyong Li. 2020. Image Captioning with a Joint Attention Mechanism by Visual Concept Samples. *ACM Trans. Multimedia Comput. Commun. Appl.* 16, 3, Article 83 (jul 2020), 22 pages.

[57] Alireza Zareian, Zhecan Wang, Haoxuan You, and Shih-Fu Chang. 2020. Learning Visual Commonsense for Robust Scene Graph Generation. In *Proc. ECCV*. Springer, 642–657.

[58] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural Motifs: Scene Graph Parsing with Global Context. In *Proc. CVPR*. IEEE, 5831–5840.

[59] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. 2017. Visual Translation Embedding Network for Visual Relation Detection. In *Proc. CVPR*. IEEE, 5532–5540.

[60] Lingling Zhang, Xiaojun Chang, Jun Liu, Minnan Luo, Zhihui Li, Lina Yao, and Alex Hauptmann. 2023. TN-ZSTAD: Transferable Network for Zero-Shot Temporal Activity Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 3 (2023), 3848–3861.

[61] Yong Zhang, Yingwei Pan, Ting Yao, Rui Huang, Tao Mei, and Chang-Wen Chen. 2022. Boosting Scene Graph Generation with Visual Relation Saliency. *ACM Trans. Multimedia Comput. Commun. Appl.* (mar 2022).

[62] Yunrui Zhao, Qianqian Xu, Yangbangyan Jiang, Peisong Wen, and Qingming Huang. 2022. Dist-PU: Positive-Unlabeled Learning From a Label Distribution Perspective. In *Proc. CVPR*. IEEE, 14461–14470.

[63] Yiwu Zhong, Jing Shi, Jianwei Yang, Chenliang Xu, and Yin Li. 2021. Learning to Generate Scene Graph from Natural Language Supervision. In *Proc. ICCV*. IEEE, 1823–1834.

[64] Yiwu Zhong, Liwei Wang, Jianshu Chen, Dong Yu, and Yin Li. 2020. Comprehensive Image Captioning via Scene Graph Decomposition. In *Proc. ECCV*. Springer, 211–229.

[65] Tianfei Zhou, Siyuan Qi, Wenguan Wang, Jianbing Shen, and Song-Chun Zhu. 2022. Cascaded Parsing of Human-Object Interaction Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 6 (2022), 2827–2840.