Revisiting Scene Graph Generation from the Perspective of Detector-Conditioned Reachability

Runfeng Qu[1,5], Pia K Bideau[3], Ole Hall[1,5], Julie Ouerfelli-Ethier[2,5], Klaus Obermayer[1,4,5], and Olaf Hellwich[1,5]

[1] Technische Universität Berlin, Germany
[2] Humboldt Universität zu Berlin, Germany
[3] Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, France
[4] Bernstein Center for Computational Neuroscience, Germany
[5] Science of Intelligence Research Cluster of Excellence, Germany
runfeng.qu@campus.tu-berlin.de

Abstract. Scene graph generation (SGG) approaches can be broadly classified into detector-based and query-based methods according to their underlying reasoning mechanisms. However, the discrepancy in their predictive behaviors, induced by these distinct mechanisms, has not been systematically analyzed. In this work, we design a controlled experimental setup to examine prediction discrepancies from the perspective of detector-conditioned reachability. The results suggest clear complementary clues. Motivated by this observation, we introduce a Dual-SGG method that consolidates both reasoning mechanisms via a dual-query design, thereby leveraging the complementary predictive behaviors of both detector-based and query-based methods. Extensive experiments on the Visual Genome, Open Images v6, and GQA-200 datasets demonstrate the effectiveness of the proposed method. Code is available at: Dual-SGG.

Keywords: Scene Graph Generation · Visual Relationship Detection · Scene Analysis and Understanding

1 Introduction

Scene graphs represent images as graphs, with nodes as entities and edges as pairwise relations. Each relationship is typically expressed as a (subject, predicate, object) triplet. Because of their high-level image understanding, scene graphs have been widely used in image captioning [33, 43], visual question answering [36], and image generation [13, 24].

Existing scene graph generation (SGG) approaches can be broadly categorized into **detector-based** and **query-based** methods according to their underlying reasoning mechanisms. Detector-based models employ an internal object detector to localize entities and subsequently perform reasoning over pairs of detected entities to predict triplets. Conceptually, these models exhibit difficulties in predicting triplets whose subject or object instances are semantically or

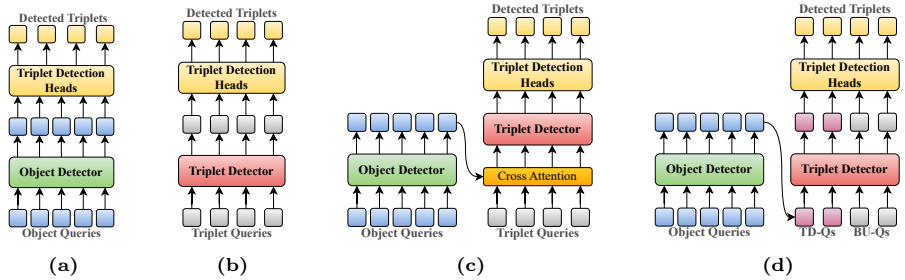


Fig. 1: SGG Models. (a) Detector-based model. (b) Query-based model. (c) Query-based model that additionally incorporates entity information derived from an object detector. (d) The proposed Dual-SGG method, which integrates detector-based and query-based reasoning mechanisms within a unified triplet decoder.

spatially uncovered by detections from the employed object detector. Here, we term this problem the **detector constraint**. To eliminate this detector constraint, query-based models introduce learnable triplet queries, inspired by the detection transformer (DETR) [1] paradigm, thereby enabling end-to-end triplet prediction without the need to enumerate entity pairs explicitly. In current studies, SGG models are predominantly evaluated using aggregate metrics such as Recall and mean-Recall, which quantify overall triplet prediction performance. However, these aggregate metrics do not elucidate the distinct predictive behaviors induced by the differing reasoning mechanisms. In particular, it remains unclear whether query-based models offer improvements on triplets that are challenging for detector-based models to detect due to the detector constraint.

To bridge this gap, we design a controlled experimental setup on the Visual Genome (VG) [17] dataset to compare prediction behaviors from the perspective of detector-conditioned reachability. To this end, we pre-train an external object detector that serves as a discriminator, determining whether a triplet is reachable for a detector-based SGG model under the original detector condition. Specifically, the ground-truth triplet set is split into two subsets. (1) **Det-T**, comprising triplets for which both the subject and object instances are successfully detected by the external object detector; and (2) **UDet-T**, comprising triplets for which at least one of the subject or object instances is not detected. Subsequently, the backbone, image encoder, and object decoder (when required) of the external object detector are employed and fine-tuned within three distinct SGG models. The first is a detector-based model (Fig. 1a), constructed following [11]. The second is a query-based model (Fig. 1b), implemented in accordance with [15]. The third is an additional query-based model (Fig. 1c) that leverages prior entity information extracted from the object detector, following [5]. To control for confounding factors other than the underlying reasoning mechanisms, all three models are trained under an identical experimental protocol. We evaluate the performance of each model on Det-T and UDet-T sets with micro-Recall, denoted as micro-DR (on Det-T) and micro-UDR (on UDet-T),

Table 1: Pre-ablation study on the SGG models presented in Fig. 1. The performance is evaluated on the VG validation dataset. All metrics are computed using the Top 100 predictions from each model. Here, micro-R, R, and mR correspond to micro-Recall, Recall, and mean-Recall, respectively, computed over the complete ground-truth set. The metrics micro-DR and micro-UDR represent micro-Recall computed on the corresponding specialized subsets.

Model	micro-R	micro-DR	micro-UDR	R	mR
a	32.5	53.6	0.3	33.4	10.9
b	30.7	47.3	5.4	31.4	9.6
c	31.8	48.8	5.8	32.4	10.3
d	34.2	53.6	4.8	35.0	11.2

in addition to standard aggregate metrics. The results are reported in Tab. 1. The detector-based model Tab. 1 (a) achieves strong performance on Det-T but nearly collapses on UDet-T, demonstrating that fine-tuning the object detector in the SGG task does not fundamentally boost the detector-based model to capture triplets involving unreachable instances for the employed object detector. In contrast, query-based models Tab. 1 (b, c) substantially improve micro-UDR. This indicates that query-based models can explore more triplets that, due to the detector constraint, are challenging for the detector-based model. However, the improvement in micro-UDR does not lead to better SGG performance. The absence of the explicit detector condition results in significant drops in micro-DR even when entity information is incorporated.

These empirical results suggest that detector- and query-based methods exhibit complementary predictive behaviors. Motivated by this complementarity, we seek a unified method that integrates both predictive behaviors. A straightforward solution would be to merge the detected triplets from detector- and query-based models by adopting the probabilistic ensembling [4] in the object detection task. However, compared to the object detection task, the triplet scores are collectively determined by three sub-probability distributions, which are non-trivial to calibrate. We therefore propose a **Dual-SGG** method, which models the mechanisms of detector- and query-based methods within a single triplet decoder, as illustrated in Fig. 1d. These two mechanisms are achieved by a dual-query design. The first group of queries is *top-down* triplet queries (TD-Qs). Specifically, they detect triplets by conditioning on given entity pairs proposed by an entity pair selector (EPS). The second group of queries is called *bottom-up* triplet queries (BU-Qs). All BU-Qs are established to globally explore triplets by originating from the image center and subsequently expanding outwards. With this approach, the final predictions encompass triplets based on both predictive behaviors. Since the predictions are obtained from a unique decoder, explicit post-hoc calibration among the predictions is not required. As demonstrated by Tab. 1 (d), our proposed method maintains equal micro-DR to the detector-based model while increasing micro-UDR, thus leading to overall superior performance.

In summary, our contributions are as follows: We systematically and quantitatively compare the predictive behaviors of query- and detector-based SGG methods from the perspective of detector-conditioned reachability. Motivated by the results, we propose a method that integrates both detector-based and query-based reasoning mechanisms within a single triplet decoder to fuse their predictive behaviors. We conduct extensive experiments on three public datasets to demonstrate the effectiveness and efficiency of the proposed method.

2 Related Work

Scene Graph Generation. Early SGG works largely adopt the detector-based method that employs an off-the-shelf Faster R-CNN [35] to propose entities. Given a detected entity pair, [31] leverages entity categories, locations, visual representations, and language prior knowledge to predict the predicates between them. [45] proposes utilizing global context information to support SGG. More recently, [21, 51] focus on improving predicate and triplet representations for predicate understanding.

Query-based SGG methods [2, 40] update a set of triplet queries to achieve SGG. Iterative-SGG [15] decomposes the triplet queries into subject, object, and predicate queries and models their dependencies with three transformer [41] decoders. [5,22] propose introducing the entity information from an object detector into triplet query updating. SpeaQ [16] adopts one-to-many and groupwise matching at the triplet level to enhance supervision during training.

Improve DETR Training with Auxiliary Queries. Due to one-to-one Hungarian matching [18], only a few detections can be assigned positive labels during the training of DETR [1], which may impede training speed. Numerous efforts [3, 12, 47, 50] have been made to accelerate training by performing one-to-many matching strategies with auxiliary queries. However, labels assigned through Hungarian matching may still pose optimization challenges. To address this issue, DN-DETR [20] and DINO-DETR [49] adopt dual-query designs. They introduce additional queries formed by perturbing ground-truth labels and train them with a denoising objective for stable supervision. For this purpose, these auxiliary queries are used only during training.

Inspired by the query grouping, we adopt a dual-query for SGG. To the best of our knowledge, this study is the first to propose fusing distinct predictive behaviors for SGG. The dual-query is designed for a fundamentally different purpose than the object detection work: not to stabilize training, but to explicitly model complementary prediction within a single end-to-end framework, replacing the need for multiple specialized models and complex fusion strategies.

3 Method

Preliminary. SGG aims to predict a set of triplets $R = \{r_k\} = \{(s_k, o_k, p_k)\}_{k=1}^{K}$ where s_k, o_k, p_k denote the subject, object, and predicate, respectively. During

inference, SGG computes a score for every triplet candidate and returns the Top K highest scoring triplets as the final predictions, *i.e.*, $R = \text{Top}\,K_r(S(r))$. According to the reasoning mechanism, the scoring function $S(r)$ of the detector-based method can be expressed as $S(r) = S_d(r \mid e_s, e_o, I)$, where I represents the image features, e_s and e_o denote the subject–object pair derived from the detected entity set E, which is $e_s, e_o \in E$. In contrast, for the query-based SGG method, the score function can be formulated as $S(r) = S_q(r \mid I)$. As this study aims to preserve the predictive behaviors of both methods, we formulate the score function as follows:

$$S(r) = \max(S_d(r \mid e_s, e_o, I),\ S_q(r \mid I)) \tag{1}$$

However, if d and q in Eq. (1) are independent, directly applying a max operation to their scores may cause one branch to dominate due to score scale inconsistencies, which would undermine the intended complementarity between the two prediction sets. To avoid this issue, we integrate both reasoning mechanisms into a unified triplet decoder θ via a dual-query design, enabling joint optimization within a single framework. Let $\hat{R} = \{\hat{r}_x\} = \{(\hat{s}_x, \hat{o}_x, \hat{p}_x)\}$ denote the set of ground-truth triplets, while the objective of our method is:

$$\max_{\theta} \sum_{x=1}^{X} \left(\log S_\theta(\hat{r}_x \mid I) + \sum_{(s,o)} (m_x \log S_\theta(\hat{r}_x \mid e_s, e_o, I)) \right) \tag{2}$$

This objective maximizes the log-likelihood of each ground-truth triplet under two conditions: (1) global image-conditioned prediction, corresponding to query-based reasoning, and (2) detector-conditioned prediction. The indicator variable m_x activates the second term only when such a match exists, defined as:

$$m_x = \mathbf{1}[\text{match}(\hat{s}_x, e_s) \wedge \text{match}(\hat{o}_x, e_o)] \tag{3}$$

thereby ensuring that the detector-conditioned supervision is applied exclusively when the subject and object instances are reachable for the object detector.

Model Overview. Our Dual-SGG is illustrated in Fig. 2. An object detector first extracts visual features and detects entities (see Sec. 3.1). An Entity Pair Selector (EPS) selects subject-object pairs from these entities to initialize TD-Qs for detector-conditioned reasoning (see Sec. 3.2). Alongside TD-Qs, BU-Qs are initialized without any entity information for global image-conditioned prediction (see Sec. 3.3). Both query types, including content embeddings and triplet anchors, are jointly updated via a unified triplet decoder (see Sec. 3.4). Ultimately, the triplet detection heads are applied for triplet prediction.

3.1 Object Detector

We adopt Deformable-DETR [52] as the object detector. Given an image, the encoder first extracts visual features I. Subsequently, the decoder generates a

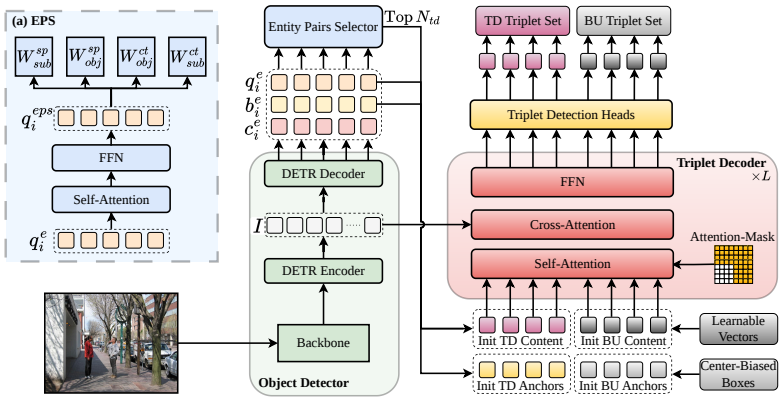


Fig. 2: Illustration of our Dual-SGG method. It includes an object detector, an entity pairs selector (EPS), and a triplet decoder. FFN denotes the feed-forward network. TD content and TD anchors form TD-Qs, while BU content and BU anchors form BU-Qs. (a) presents architectural details of EPS.

set of entity candidates, denoted as $E = \{e_i\} = \{(q_i^e, b_i^e, c_i^e)\}_{i=1}^{N_e}$, where $q_i^e \in \mathbb{R}^d$, $b_i^e \in \mathbb{R}^4$, and $c_i^e \in \mathbb{R}^{N_c}$ refer to the entity features, boxes, and category distributions, respectively.

Object Detector Loss. We follow [52] to train the object detector. The Hungarian matching is applied to search for an optimal assignment between E and the ground-truth entity set by minimizing a global matching cost. Based on this assignment, the ground-truth entity set is permuted and padded with *no-object* labels for the unmatched detections. We define the permuted ground-truth set as $\hat{E} = \{\hat{e}_i\} = \{(\hat{b}_i^e, \hat{c}_i^e)\}_{i=1}^{N_e}$, where $\hat{b}_i^e \in \mathbb{R}^4$ and $\hat{c}_i^e \in \mathbb{R}^{N_c}$ indicate the target boxes and one-hot class labels of entities. The object detector loss includes a focal loss [26] for entity classification and L1 plus GIoU losses for box regression:

$$\mathcal{L}^e = \sum_{i=1}^{N_e} [\lambda^{cls}\mathcal{L}^{cls}(c_i^e, \hat{c}_i^e) + \lambda^{box}\mathcal{L}^{box}(b_i^e, \hat{b}_i^e) + \lambda^{giou}\mathcal{L}^{giou}(b_i^e, \hat{b}_i^e)] \tag{4}$$

Building upon the aligned entity set $\hat{E}$, we further construct a predicate matrix $\hat{P}^e \in \mathbb{R}^{N_e \times N_e \times N_p}$. For each detected entity pair (i, j), $\hat{P}_{ij}^e$ is a one-hot vector indicating the ground truth predicate between entity i and j. Note that the matrix $\hat{P}^e$ is used for training the triplet decoder and the EPS.

3.2 Entity Pairs Selector (EPS)

TD-Qs are initialized with the detected entity pairs to introduce the detector condition. However, enumerating all possible pairs results in N_e^2 queries, which leads to significant computational overhead. To reduce this cost, we employ an Entity

Pair Selector (EPS) that evaluates both the content and spatial compatibility of entity pairs and selects the Top N_{td} candidates for TD-Qs initialization. The content score favors entity pairs that correspond to ground-truth triplets, providing stable positive supervision for the subsequent training of triplet prediction. The spatial score emphasizes pairs that are spatially similar to content-positive candidates, providing informative hard negative samples.

Unlike prior work [14, 34, 42], which uses intricate modules to evaluate detected triplets and adjust their scores, EPS serves here as a selector rather than a post-hoc scoring mechanism. While not directly determine final predictions, EPS enables a computationally efficient design, as shown in Fig. 2 (a). EPS includes a self-attention layer and a FFN for updating the entity features $\{q_i^e\}_{i=1}^{N_e}$, yielding $\{q_i^{eps}\}_{i=1}^{N_e}$. The spatial logits $S^{sp} \in \mathbb{R}^{N_e \times N_e}$ and content logits $S^{ct} \in \mathbb{R}^{N_e \times N_e}$ are obtained by performing a dot-product as follows:

$$S^{sp} = \{q_i^{eps}\}W_{sub}^{sp} \cdot (\{q_i^{eps}\}W_{obj}^{sp})^T \qquad S^{ct} = \{q_i^{eps}\}W_{sub}^{ct} \cdot (\{q_i^{eps}\}W_{obj}^{ct})^T \tag{5}$$

where $W_{sub}^{sp}, W_{obj}^{sp}, W_{sub}^{ct}, W_{obj}^{ct}$ are linear projectors. Ultimately, the final scores for entity pairs $S^{eps} \in \mathbb{R}^{N_e \times N_e}$ are calculated as follows:

$$S^{eps} = \begin{cases} \text{sigmoid}(S^{sp}) \times \text{sigmoid}(S^{ct}), & \text{Training} \\ \text{sigmoid}(S^{ct}), & \text{Inference} \end{cases} \tag{6}$$

EPS Loss. The EPS loss consists of two focal losses: spatial loss $\mathcal{L}^{sp}$ and content loss $\mathcal{L}^{ct}$.

$$\mathcal{L}^{eps} = \mathcal{L}^{sp}(S^{sp}, \hat{S}^{sp}) + \mathcal{L}^{ct}(S^{ct}, \hat{S}^{ct}) \tag{7}$$

The content labels $\hat{S}^{ct} \in \mathbb{R}^{N_e \times N_e}$ are derived by converting the predicate matrix $\hat{P}^e$ introduced in Sec. 3.1 into a binary matrix:

$$\hat{S}_{ij}^{ct} = \begin{cases} 1, & \text{If } \sum_{k=1}^{N_p} \hat{P}_{ijk}^e > 0 \\ 0, & \text{Otherwise} \end{cases} \tag{8}$$

Rather than using the locations of ground-truth triplets, we use the locations of detected entity pairs with content-positive labels to assign spatial labels, as shown in Eq. (9). This ensures that all spatial-positive pairs exhibit spatial configurations similar to the content-positive samples:

$$\hat{S}_{ij}^{sp} = \begin{cases} 1, & \text{if IoU}(b_i^e, b_x^e) > 0.5 \wedge \text{IoU}(b_j^e, b_y^e) > 0.5 \wedge \hat{S}_{xy}^{ct} = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

where $b_i^e, b_j^e, b_x^e, b_y^e \in E$ are the entity boxes produced by the object detector.

3.3 Triplet Queries Initialization

Inspired by DAB-DETR [28], our TD-Qs and BU-Qs consist of content embeddings and triplet anchors. The triplet anchors are introduced to guide the attention of the triplet decoder in detecting triplets in two different patterns.

TD-Qs Initialization. To impose the detector condition, TD-Qs are constructed with the selected pairs of entities. Specifically, the TD-Qs content embeddings and triplet anchors are initialized by concatenating the corresponding entity features and bounding boxes, respectively. Given $S^{eps} \in \mathbb{R}^{N_e \times N_e}$, Top N_{td} entity pairs are selected. Let $\tilde{s}_{(i)}$ and $\tilde{o}_{(i)}$ denote the indices of the entities involved in the selected pairs by EPS. The content embedding $\{q_i^{td}\}_{i=1}^{N_{td}}$ and triplet anchors $\{a_i^{td}\}_{i=1}^{N_{td}}$ are initialized as follows:

$$\{q_i^{td}\} = \text{Concat}(\{q^e_{\tilde{s}_{(i)}}\}, \{q^e_{\tilde{o}_{(i)}}\}), \qquad \{a_i^{td}\} = \text{Concat}(\{b^e_{\tilde{s}_{(i)}}\}, \{b^e_{\tilde{o}_{(i)}}\}) \tag{10}$$

where $q_i^{td} \in \mathbb{R}^{2d}$ and $a_i^{td} \in \mathbb{R}^8$.

BU-Qs Initialization. For BU-Qs, the content embeddings $\{q_i^{bu}\}_{i=1}^{N_{bu}}$, where $q_i^{bu} \in \mathbb{R}^{2d}$, are randomly initialized with a set of learnable vectors. The anchors of BU-Qs, denoted as $\{a_i^{bu}\}_{i=1}^{N_{bu}}$, where $a_i^{bu} \in \mathbb{R}^8$, are initialized with a set of pairwise frozen boxes centered in the image. These center-biased anchors encourage the decoder to initially focus on the central region and gradually expand its attention outwards across decoder layers. Consequently, the BU-Qs capture triplets that are not reachable for the TD-Qs due to the detector constraint.

3.4 Triplet Decoder

The triplet decoder consists of L transformer [41] blocks. To reduce computation, we adopt a single-branch decoder that reasons over complete triplets. This design can be extended to a multi-branch version, where each branch models a triplet component, as in [5, 8, 15]. The decoder takes the initialized TD-Qs and BU-Qs as input and updates them jointly through stacked transformer blocks. Triplet detection heads are then applied to the updated queries, producing two triplet sets denoted as $R^{td} = \{r_i^{td}\} = \{(s_i^{td}, o_i^{td}, p_i^{td})\}_{i=1}^{N_{td}}$ and $R^{bu} = \{r_i^{bu}\} = \{(s_i^{bu}, o_i^{bu}, p_i^{bu})\}_{i=1}^{N_{bu}}$, respectively. To prevent the rich prior in TD-Qs from leaking into BU-Qs and causing a significant dependency of R^{bu} on R^{td}, we use self-attention masks [41] to restrict the information flow from TD-Qs to BU-Qs in each transformer's self-attention layer.

TD-Loss. To ensure R^{td} predicted by TD-Qs follows the provided detector condition, we construct the labels for triplets detected with TD-Qs, denoted as $\hat{R}^{td} = \{\hat{r}_i^{td}\} = \{(\hat{s}_i^{td}, \hat{o}_i^{td}, \hat{p}_i^{td})\}_{i=1}^{N_{td}}$, based on the labels of their initial entities obtained from object detector training. Specifically, given the ground-truth labels $\hat{E} = \{\hat{e}_i\} = \{(\hat{b}_i^e, \hat{c}_i^e)\}_{i=1}^{N_e}$ of the detected entities introduced in Sec. 3.1, the labels of subjects and objects are extracted as $\{\hat{s}_i^{td}\} = \{(\hat{b}^e_{\tilde{s}_{(i)}}, \hat{c}^e_{\tilde{s}_{(i)}})\}_{i=1}^{N_{td}}$ and $\{\hat{o}_i^{td}\} = \{(\hat{b}^e_{\tilde{o}_{(i)}}, \hat{c}^e_{\tilde{o}_{(i)}})\}_{i=1}^{N_{td}}$, respectively, where $\tilde{s}_{(i)}$ and $\tilde{o}_{(i)}$ are the indices of entities involved by the selected entity pairs, as introduced in Sec. 3.3. The predicate labels are extracted from $\hat{P}^e$ defined as $\{\hat{p}_i^{td}\} = \{\hat{P}^e_{\tilde{s}_{(i)}\tilde{o}_{(i)}}\}_{i=1}^{N_{td}}$. The entire objective of TD-Qs is as follows:

$$\mathcal{L}^{TD} = \sum_{i}^{N_{td}} [\mathcal{L}^{TD}_{sub}(s_i^{td}, \hat{s}_i^{td}) + \mathcal{L}^{TD}_{obj}(o_i^{td}, \hat{o}_i^{td}) + \mathcal{L}^{TD}_{p}(p_i^{td}, \hat{p}_i^{td})] \tag{11}$$

where L_p^{TD} is a focal loss, and $\mathcal{L}_{sub}^{TD}$ and $\mathcal{L}_{obj}^{TD}$ are identical and share the same loss weights with the object detector loss. For instance:

$$\mathcal{L}_{sub}^{TD}(s_i^{td}, \hat{s}_i^{td}) = \lambda^{cls}\mathcal{L}^{cls}(s_{i,c}^{td}, \hat{s}_{i,c}^{td}) + \lambda^{box}\mathcal{L}^{box}(s_{i,b}^{td}, \hat{s}_{i,b}^{td}) + \lambda^{giou}\mathcal{L}^{giou}(s_{i,b}^{td}, \hat{s}_{i,b}^{td}) \tag{12}$$

$s_{i,c}^{td}, \hat{s}_{i,c}^{td}$ refer to the categories and $s_{i,b}^{td}, \hat{s}_{i,b}^{td}$ indicate the boxes of corresponding subjects. Note that the TD-Qs may not be trained on all ground-truth triplets due to the selection process of EPS.

BU-Loss. BU-Qs are supervised following the standard query-based SGG method, where labels are assigned by applying Hungarian matching between detected triplets and ground-truth triplets in a one-to-one manner, as in [15]. However, in our approach, the predicate cost is excluded from the matching cost calculation, as its computational overhead outweighs its empirical benefit. Ultimately, given the labels for the detected triplets, the calculation of BU-loss, denoted as $\mathcal{L}^{BU}$, is analogous to Eq. (11) and Eq. (12).

3.5 Training and Inference

Training. In summary, the entire loss function of our method is:

$$\mathcal{L} = \mathcal{L}^{e} + \mathcal{L}^{eps} + \mathcal{L}^{TD} + \mathcal{L}^{BU} \tag{13}$$

where $\mathcal{L}^{TD}$ and $\mathcal{L}^{BU}$ realize $\sum_{(s,o)}(m_x \log S_\theta(\hat{r}_x \mid e_s, e_o, I))$ and $\log S_\theta(\hat{r}_x \mid I)$ in Eq. (2), respectively. $\mathcal{L}^{e}$ and $\mathcal{L}^{eps}$ further ensure high-quality entity-pair proposals for TD-Qs initialization. Whilst the present work introduces additional loss terms in comparison to previous studies, all newly added losses are assigned a unit weight to isolate the influences from hyperparameter tuning.

Inference. In the inference stage, the predictions R^{td} and R^{bu} are concatenated to form the final prediction set $R = \{r_i\}_{i=1}^{N_{td}+N_{bu}}$. In addition, Non-Maximum Suppression (NMS) is applied to serve as a practical implementation of the max operation in Eq. (1) for removing duplicate triplets, as in [5, 15, 16, 21, 42]. Ultimately, Top K triplets are selected based on the triplet scores, which are calculated by multiplying the subject, object, and predicate scores together.

4 Experiments

4.1 Experimental Setup

This section contains the descriptions of datasets, evaluation metrics, and implementation details.

Datasets. The experiments are conducted on the following datasets:

Visual Genome (VG) [17] includes 57,723 training, 5,000 validation, and 26,446 test images. It defines 150 entity and 50 predicate categories.

Open Images V6 (OIv6) [19] consists of 126,368 training, 1,813 validation, and 5,322 test images. It features 601 entity and 30 predicate categories.

GQA-200 [10] contains 27,623 training, 5,000 validation, and 8,208 test images, comprising 200 entity and 100 predicate categories.

Evaluation Metrics. We evaluate our method on the scene graph detection (SGDet) task, following [11, 21, 40, 42, 45]. On VG and GQA-200, Recall@K (R@K), mean Recall@K (mR@K), and their harmonic mean F@K are reported under the graph constraint proposed by [45]. For OIv6, we report mR@50, micro-Recall@50 (micro-R@50), weighted mean AP of triplets (wmAPrel), weighted mean AP of phrases (wmAPphr), and a final score defined as score $= 0.2 \times$ micro-R@50 $+ 0.4 \times$ wmAPrel $+ 0.4 \times$ wmAPphr.

Implementation Details. For all three datasets, we adopt the same training setup and hyperparameters. The training consists of two parts. First, we pre-train the object detector (Deformable-DETR [52]) with a ResNet50 [9] backbone and 200 entity queries (N_e). The loss weights λ^{cls}, λ^{box}, and λ^{giou} are 2, 5, and 2. Next, the EPS and triplet decoder are jointly trained while fine-tuning the pre-trained object detector. We set $N_{td} = 300$, $N_{bu} = 800$, and $L = 4$. We use AdamW [30] as an optimizer with an initial learning rate of 10^{-4} and a weight decay of 10^{-4}. The batch size is 16 in total. This second part is trained for 25 epochs, with the learning rate reduced by a factor of 0.1 at epoch 20. All experiments run on a single NVIDIA RTX A6000 GPU.

4.2 Main Results

We evaluate our method by comparing it with previous SGG models, including Motifs [45], SSR-CNN [40], PE-Net [51], DRM w/o DKT [21], RelTR [5], EGTR [11], and ISG [15]. To evaluate Dual-SGG on the long-tailed distribution problem, we apply logit adjustment (LA) [32] as a debiasing strategy. The result is compared with other Unbiased-SGG models, such as BGNN [23], SHA [7], IETrans [46], SSR-CNN+LA [40], DRM [21], EGTR+LA [11], and Salience-SGG [34]. **Note**: We further compare with VETO [38], SpeaQ [16], and HQSG [8] without graph constraints in the **suppl. material**.

VG Dataset. Tab. 2 presents the comparison results on the VG test set. Among SGG models with a ResNet [9] backbone, Dual-SGG achieves the best results, outperforming the second-best EGTR in all metrics. Despite being heavier, our model achieves a similar inference speed to EGTR (14.0 FPS vs 14.7 FPS), as EGTR employs an MLP performing on N_e^2 detected entity pairs. RelTR exhibits close performance in mR@K and F@K to our method but has a significant lag in R@K of 6.0/7.8. Our method achieves comparable performance to DRM w/o DKT, which employs X101-FPN as the backbone. Specifically, the relative differences between Dual-SGG and DRM w/o DKT are -0.5/-0.4 in R@K, 1.3/1.1 in mR@K, and 1.6/1.2 in F@K. However, our Dual-SGG is much lighter and faster than DRM w/o DKT, as the X101-FPN backbone alone requires more parameters than our entire model. Regarding Unbiased-SGG, the debiasing strategy in Dual-SGG+LA leads to a relatively smaller degradation in R@K compared to SSR-CNN+LA and EGTR+LA. This is attributed to the improved exploration of subject-object instances by our Dual-SGG, which is less sensitive to predicate understanding. Consequently, our Dual-SGG+LA at

Table 2: Graph-Constraint results on the VG test set. Methods are grouped by backbone into *X101-FPN* [25] (top) and *ResNet* [9] (bottom). In each group, the best and second-best results are highlighted in **bold** and underline. ✓ denotes methods with debiasing strategies (*i.e.* Unbiased-SGG). LA indicates logit adjustment [32]

Unbiased	Method	#params(M)	FPS	R@50	R@100	mR@50	mR@100	F@50	F@100
	Motifs [45][CVPR'18]	369.9	1.9	32.1	36.9	5.5	6.8	9.4	11.5
	VCTree [39][CVPR'19]	361.5	0.8	31.8	36.1	6.6	7.7	10.9	12.7
✓	BGNN [23][CVPR'21]	341.9	1.7	31.0	35.8	10.7	12.6	15.9	18.6
✓	DT2-ACBS [6][ICCV'21]	-	-	15.0	16.3	**22.0**	**24.4**	17.8	19.5
✓	IETrans [46][ECCV'22]	369.9	1.9	23.5	27.3	15.7	18.2	18.8	21.8
✓	SHA [7][CVPR'22]	-	-	14.9	18.2	17.9	20.9	16.3	19.5
	SSR-CNN [40][CVPR'22]	274.3	3.1	33.5	38.4	8.6	10.3	13.7	16.2
✓	SSR-CNN+LA* [40][CVPR'22]	274.3	3.1	23.7	27.3	18.6	22.5	**20.8**	**24.7**
	PE-Net [51][CVPR'23]	-	-	32.4	36.9	8.9	11.0	14.0	16.9
	DRM w/o DKT [21][CVPR'24]	-	-	**34.0**	**38.9**	9.0	11.2	14.2	17.4
✓	DRM [21][CVPR'24]	-	-	19.0	22.9	20.4	24.1	20.8	23.5
✓	RA-SGG [44][AAAI'25]	-	-	26.0	30.3	14.4	17.1	18.5	21.9
✓	SGTR [22][CVPR'22]	117.1	6.2	25.1	26.6	12.0	14.6	16.2	18.9
	RelTR [5][TPAMI'22]	63.7	13.4	27.5	30.7	10.8	12.3	15.5	17.6
	Relationformer [37][ECCV'22]	92.9	8.5	28.4	31.3	9.3	10.7	14.0	15.9
	ISG [15][NIPS'22]	93.5	6.0	29.7	32.1	8.0	8.8	12.6	13.8
✓	Mg-RMPN [42][ECCV'24]	-	-	29.1	33.5	14.4	17.3	19.3	22.8
	EGTR [11][CVPR'24]	42.5	14.7	30.2	34.3	7.9	10.1	12.5	15.6
✓	EGTR+LA [11][CVPR'24]	42.5	14.7	24.2	26.7	17.1	21.4	20.0	23.8
✓	Hydra-SGG [2][ICLR'25]	67.6	5.3	28.4	33.1	16.0	19.7	20.5	24.7
✓	Salience-SGG [34][WACV'26]	77.7	13.3	28.8	33.4	18.0	21.6	22.1	26.2
	Dual-SGG (Ours)	84.7	14.0	**33.5**	**38.5**	10.3	12.3	15.8	18.6
✓	Dual-SGG+LA $\tau = 0.2$ (Ours)	84.7	14.0	30.2	35.2	18.7	22.2	23.1	27.3
✓	Dual-SGG+LA $\tau = 0.3$ (Ours)	84.7	14.0	25.3	30.0	**22.2**	**25.2**	**23.6**	**27.4**

$\tau = 0.3$ achieves state-of-the-art performance, attaining F@K of 23.6/27.4 and mR@K of 22.2/25.2 simultaneously.

GQA-200 Dataset. The results on the GQA-200 test dataset are shown in Tab. 3. Consistent with the observations for VG, Dual-SGG attains the best performance among SGG models, including VTransE, Motifs, and VCTree. Furthermore, compared with other Unbiased-SGG models, our Dual-SGG+LA at $\tau = 0.3$ establishes new state-of-the-art results, achieving mR@K of 19.0/21.1 and F@K of 20.4/23.2 simultaneously.

OIv6 Dataset. Tab. 4 demonstrates the effectiveness of our method on the OIv6 dataset. Dual-SGG outperforms previous methods under all metrics.

4.3 Ablation Studies

In the ablation studies, all results are produced on the VG test dataset. The analysis of hyperparameters and efficiency is provided in the **suppl. material**.

TD-Qs and BU-Qs Analysis. To analyze the effect of TD-Qs and BU-Qs, we compare the results of the Dual-SGG w/o TD-Qs, Dual-SGG w/o BU-Qs, and Dual-SGG. Note that when only a single group of triplet queries is employed,

Table 3: Graph-Constraint results on the GQA-200 test set.

Unbiased	Method	R@50	R@100	mR@50	mR@100	F@50	F@100
	VTransE [48][CVPR'17]	27.2	30.7	5.8	6.6	9.6	10.9
	Motifs [45][CVPR'18]	**28.9**	**33.1**	6.4	7.7	10.5	12.5
	VCTree [39][CVPR'19]	28.3	31.9	6.5	7.4	10.6	12.0
✓	SHA [7][CVPR'22]	14.8	17.9	17.8	20.1	16.2	18.9
✓	DRM [21][CVPR'24]	18.6	21.7	**18.9**	**21.0**	**18.7**	**21.3**
✓	Mg-RMPN [42][ECCV'24]	23.2	25.7	12.8	14.5	16.5	18.5
✓	Hydra-SGG [2][ICLR'25]	22.8	26.5	12.7	15.9	16.3	19.9
✓	Salience-SGG [34][WACV'26]	23.6	26.6	16.2	18.4	19.2	21.7
	Dual-SGG (Ours)	**29.4**	**33.3**	9.1	10.4	13.9	15.8
✓	Dual-SGG+LA $\tau = 0.2$ (Ours)	26.2	29.9	17.1	19.3	**20.7**	**23.4**
✓	Dual-SGG+LA $\tau = 0.3$ (Ours)	22.0	25.7	**19.0**	**21.1**	20.4	23.2

Table 4: Results on the OIv6 test set.

Method	mR@50	micro-R@50	wmAPrel	wmAPphr	score
BGNN [23][CVPR'21]	40.5	75.0	33.5	34.2	42.1
RU-Net [27][CVPR'22]	-	**76.9**	35.4	34.9	43.5
SSR-CNN [40][CVPR'22]	42.8	76.7	**41.5**	**43.6**	**49.4**
PE-Net [51][CVPR'23]	-	76.5	36.6	37.4	44.9
SQUAT [14][ICCV'23]	-	75.8	34.9	35.9	43.5
DRM [21][CVPR'24]	-	75.9	40.5	41.4	47.9
SGTR [22][CVPR'22]	42.6	59.9	37.0	38.7	42.3
RelTR [5][TPAMI'22]	-	71.7	37.2	37.5	43.0
Mg-RMPN [42][ECCV'24]	45.5	77.8	35.5	36.4	43.6
Hydra-SGG [2][ICLR'25]	-	76.1	42.8	44.3	50.1
Salience-SGG [34][WACV'26]	48.0	78.1	45.6	44.9	51.8
Dual-SGG (Ours)	48.0	**79.6**	**46.0**	**46.5**	**52.9**

we extend the number of triplet queries to $N_{td} + N_{bu}$ for fair comparison. In addition to R@K and mR@K, we also detail the micro-R@K, micro-UDR@K, and micro-DR@K, as introduced in Sec. 1, to provide a breakdown analysis from the perspective of detector-conditioned reachability. To this end, we report the performance of the detector-based model introduced in Fig. 1a as the baseline. The sizes of the created Det-T and UDet-T on the VG test set are 98,103 and 54,123. The comparison is shown in Tab. 5 above. Dual-SGG w/o BU-Qs slightly improves SGG performance compared to the baseline by achieving a higher micro-UDR@K. Despite the TD-Qs being designed to approximate the detector-based reasoning mechanism, updating TD-Qs in the context of triplets provides the possibility to explore triplets involving instances that are semantically uncovered by the object detector. However, the micro-UDR@K is still lagging behind Dual-SGG w/o TD-Qs, as TD-Qs struggle to capture the triplets spatially outside the detector-conditioned reachability, as shown in Fig. 3. Consistent with Tab. 1, the great micro-UDR@K of Dual-SGG w/o TD-Qs comes at the cost of micro-DR@K, which leads to lower overall performance. Dual-SGG achieves the best overall performance and outperforms the baseline and Dual-SGG w/o BU-Qs by increasing micro-UDR@K while maintaining micro-DR@K. This comparison demonstrates the necessity of the co-existence of TD-Qs and BU-Qs under the same total triplet query budget.

Table 5: Ablation study on BU-Qs and TD-Qs. Baseline refers to the model shown in Fig. 1a. $*$ indicates that the model employ Swin-B as backbone and N_e is set as 900.

Models	R@50/100	mR@50/100	micro-R@50/100	micro-DR@50/100	micro-UDR@50/100
Baseline	31.6/36.0	9.1/11.2	30.2/34.8	46.7/53.8	0.2/0.3
Dual-SGG w/o BU-Qs	31.9/36.3	8.6/10.3	30.4/35.2	46.3/53.4	1.5/2.2
Dual-SGG w/o TD-Qs	31.3/34.7	10.0/11.6	30.0/33.7	44.1/49.0	4.5/5.9
Dual-SGG	33.5/38.5	10.3/12.3	31.8/36.9	47.3/54.3	3.7/5.2
Dual-SGG* w/o BU-Qs	37.1/42.7	10.1/13.0	35.2/40.6	46.5/53.6	1.5/1.7
Dual-SGG* w/o TD-Qs	35.3/38.5	11.3/13.0	33.6/37.2	43.4/47.6	4.4/6.1
Dual-SGG*	38.8/44.6	11.7/14.8	36.3/42.5	47.2/54.9	3.8/5.7

Table 6: Ablation study on complementarity. The top part indicates the performances of specialized models. The bottom part shows the results from various fusion strategies.

Models	R@50/100	mR@50/100	micro-R@50/100	micro-DR@50/100	micro-UDR@50/100
Dual-SGG w/o BU-Qs	31.7/36.1	8.4/9.9	30.4/34.9	46.3/53.1	1.5/1.8
Dual-SGG w/o TD-Qs	30.8/33.8	9.4/11.0	29.5/32.8	43.8/48.2	3.7/4.8
Max	31.7/36.3	9.0/11.2	28.9/34.1	43.1/50.3	3.2/4.7
Average	17.6/29.7	5.3/9.3	17.1/28.9	25.3/42.7	2.2/4.1
Normalization	32.2/37.3	9.6/11.8	30.7/36.0	45.8/53.2	3.3/4.9
Dual-SGG	33.5/38.5	10.3/12.3	31.8/36.9	47.3/54.3	3.7/5.2

Strong Detector Condition Case Analysis. The performance improvement of Dual-SGG over Dual-SGG w/o BU-Qs primarily arises from its enhanced results on UDet-T, which may be sensitive to the size of UDet-T. If a stronger object detector is employed, reducing UDet-T, the improvement of Dual-SGG may be negligible. To investigate the effect of Dual-SGG under stronger detector conditions, we conduct an additional experiment. Specifically, we train new Dual-SGG, Dual-SGG w/o BU-Qs, and Dual-SGG w/o TD-Qs based on a new object detector constructed by replacing the backbone of the default detector (Deformable DETR) with Swin-B [29], and the entity query (N_e) is set to 900. The results are shown in Tab. 5 below. Although UDet-T shrinks from 35% to 28% under a stronger detector condition, our Dual-SGG still remains effective.

Complementarity Analysis. In this section, we provide a detailed comparison to analyze 1) whether Dual-SGG can truly capture the complementary predictive behaviors of specialized detector-based and query-based models, and 2) whether the joint updating of TD-Qs and BU-Qs is necessary. Specifically, we independently train a Dual-SGG w/o TD-Qs and a Dual-SGG w/o BU-Qs as specialized models, setting the number of triplet queries as N_{bu} and N_{td}. The performances of these models are reported in Tab. 6 above. Subsequently, Top 100 predictions from each specialized model are taken to form a 200-candidate pool. We apply three post-hoc methods, including max, average, and normalization, as alternative fusion strategies that directly calibrate the 200 triplet scores and

Table 7: Ablation study on individual model components. MSA indicates the self-attention mask and cA^{bu} refers to the center-biased boxes in BU-Qs initialization.

S^{sp}	S^{ct}	MSA	cA^{bu}	R@50	R@100	mR@50	mR@100
✓		✓	✓	33.0	37.5	10.1	12.2
	✓	✓	✓	31.5	36.5	9.5	11.6
✓	✓		✓	32.8	37.5	9.8	11.7
✓	✓	✓		32.9	37.8	9.7	12.0
✓	✓	✓	✓	33.5	38.5	10.3	12.3

re-rank them, yielding 100 predictions for the evaluation. As shown in Tab. 6, our Dual-SGG outperforms both specialized models on all evaluation metrics, demonstrating that our Dual-SGG is attributable to the complementary predictions of the two specialized models. In addition, the comparison with other fusion strategies reflects the necessity of the joint updating of TD-Qs and BU-Qs.

Individual Components. Tab. 7 demonstrates the effect of other individual components in Dual-SGG, including the spatial and content scores in EPS *i.e.* S^{sp} and S^{ct}, the self-attention mask in the triplet decoder, and the center-biased initialization of triplet anchors $\{a_i^{bu}\}_{i=1}^{N_{bu}}$. The exclusion of S^{sp} or S^{ct} results in degradation, as they provide high-quality positive and negative samples, respectively. When the self-attention mask is removed, the decreased performance demonstrates that the dependency of R^{bu} on R^{td} leads to a negative effect on complementarity. The results in the last two rows illustrate that the center-biased initialization for $\{a_i^{bu}\}_{i=1}^{N_{bu}}$ is important for achieving better performance. Note that when center-biased initialization is removed, the anchors of BU-Qs are initialized with uniformly distributed boxes over the image, which is widely adopted in DETR-style models.

4.4 Qualitative Results

Attention Visualization. We identify the Top 100 triplets from each of R^{td} and R^{bu}. The corresponding triplet queries are tracked across the decoder layers. For each layer, the cross-attention maps of the TD-Qs and BU-Qs are aggregated separately to produce heat maps. The heat maps indicate the focused regions of TD-Qs and BU-Qs for exploring triplets. As shown in Fig. 3, distinct exploration patterns are exhibited. Specifically, BU-Qs initially focus on the center of the image due to the center-biased initialization and gradually spread outwards. In contrast, TD-Qs consistently focus on the regions of the initial entity pairs. This figure illustrates that the BU-Qs can capture the triplets involving instances that are spatially unreachable for the TD-Qs due to the detector constraint.

Complementarity Visualization. Fig. 4a and Fig. 4b demonstrate the complementarity property of our Dual-SGG. The triplets missed by either Dual-SGG w/o TD-Qs or Dual-SGG w/o BU-Qs can be successfully predicted by Dual-SGG. Fig. 4c illustrates that sometimes the complementarity of Dual-SGG cannot recover every detected ground-truth triplet from Dual-SGG w/o TD-Qs and Dual-SGG w/o BU-Qs. We aim to address this limitation in future work.

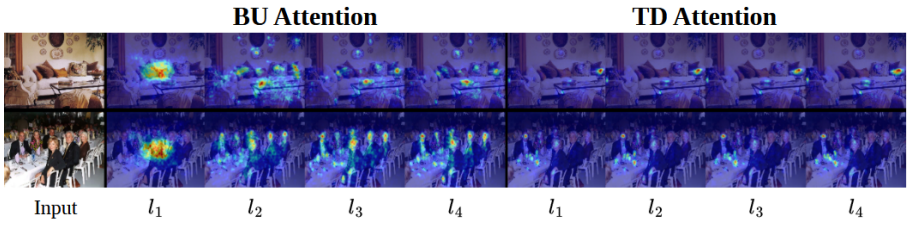
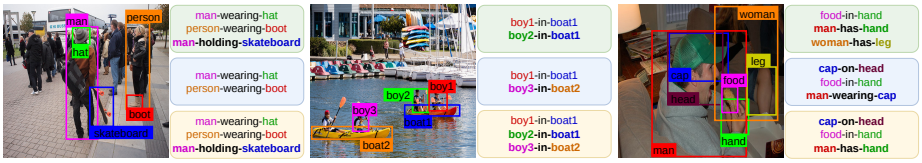


Fig. 3: Attention visualization. Columns 2–5 and 6-end depict the attention distributions of BU-Qs and TD-Qs across the decoder layers (l_i), respectively.



(a) **(b)** **(c)**

Fig. 4: Example visualization. For each image, the detected ground-truth triplets by Dual-SGG w/o TD-Qs (top), Dual-SGG w/o BU-Qs (middle), and Dual-SGG (bottom), are provided. Triplets highlighted in **bold** denote missed triplets by at least one of the other two models.

5 Conclusion

This study investigates the distinct predictive behaviors of detector-based and query-based SGG methods from the perspective of detector-conditioned reachability. The analysis reveals that detector-based methods effectively capture triplets that lie within their detector-conditioned reachability space, whereas query-based methods are capable of identifying triplets beyond this space. Based on these insights, a Dual-SGG method is introduced, achieving superior performance by amalgamating the strengths of the two reasoning mechanisms within a single framework. The end-to-end framework consolidates the distinct predictive behaviors without the need for multiple specialized models and complex fusing strategies.

Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2002/1 "Science of Intelligence" – project number 390523135. This work has been partially supported by MIAI Cluster - project number ANR-23-IACL-0006.

References

1. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
2. Chen, M., Chen, G., Wang, W., Yang, Y.: Hydra-sgg: Hybrid relation assignment for one-stage scene graph generation. arXiv preprint arXiv:2409.10262 (2024)
3. Chen, Q., Chen, X., Wang, J., Zhang, S., Yao, K., Feng, H., Han, J., Ding, E., Zeng, G., Wang, J.: Group detr: Fast detr training with group-wise one-to-many assignment. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6633–6642 (2023)
4. Chen, Y.T., Shi, J., Ye, Z., Mertz, C., Ramanan, D., Kong, S.: Multimodal object detection via probabilistic ensembling. In: European Conference on Computer Vision. pp. 139–158. Springer (2022)
5. Cong, Y., Yang, M.Y., Rosenhahn, B.: Reltr: Relation transformer for scene graph generation. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(9), 11169–11183 (2023)
6. Desai, A., Wu, T.Y., Tripathi, S., Vasconcelos, N.: Learning of visual relations: The devil is in the tails. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15404–15413 (2021)
7. Dong, X., Gan, T., Song, X., Wu, J., Cheng, Y., Nie, L.: Stacked hybrid-attention and group collaborative learning for unbiased scene graph generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19427–19436 (2022)
8. Fu, J., Zhang, T., Chen, K., Dou, Q.: Hybrid reciprocal transformer with triplet feature alignment for scene graph generation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 8953–8963 (2025)
9. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: European conference on computer vision. pp. 630–645. Springer (2016)
10. Hudson, D.A., Manning, C.D.: Gqa: A new dataset for real-world visual reasoning and compositional question answering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6700–6709 (2019)
11. Im, J., Nam, J., Park, N., Lee, H., Park, S.: Egtr: Extracting graph from transformer for scene graph generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 24229–24238 (2024)
12. Jia, D., Yuan, Y., He, H., Wu, X., Yu, H., Lin, W., Sun, L., Zhang, C., Hu, H.: Detrs with hybrid matching. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19702–19712 (2023)
13. Johnson, J., Gupta, A., Fei-Fei, L.: Image generation from scene graphs. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1219–1228 (2018)
14. Jung, D., Kim, S., Kim, W.H., Cho, M.: Devil's on the edges: Selective quad attention for scene graph generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18664–18674 (2023)
15. Khandelwal, S., Sigal, L.: Iterative scene graph generation. Advances in Neural Information Processing Systems **35**, 24295–24308 (2022)
16. Kim, J., Park, J., Park, J., Kim, J., Kim, S., Kim, H.J.: Groupwise query specialization and quality-aware multi-assignment for transformer-based visual relationship detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 28160–28169 (2024)

17. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., et al.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. International journal of computer vision **123**(1), 32–73 (2017)
18. Kuhn, H.W.: The hungarian method for the assignment problem. Naval research logistics quarterly **2**(1-2), 83–97 (1955)
19. Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., et al.: The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. International journal of computer vision **128**, 1956–1981 (2020)
20. Li, F., Zhang, H., Liu, S., Guo, J., Ni, L.M., Zhang, L.: Dn-detr: Accelerate detr training by introducing query denoising. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13619–13627 (2022)
21. Li, J., Wang, Y., Guo, X., Yang, R., Li, W.: Leveraging predicate and triplet learning for scene graph generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 28369–28379 (2024)
22. Li, R., Zhang, S., He, X.: Sgtr: End-to-end scene graph generation with transformer. In: proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19486–19496 (2022)
23. Li, R., Zhang, S., Wan, B., He, X.: Bipartite graph network with adaptive message passing for unbiased scene graph generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11109–11119 (2021)
24. Li, Y., Ma, T., Bai, Y., Duan, N., Wei, S., Wang, X.: Pastegan: A semi-parametric method to generate image from scene graph. Advances in Neural Information Processing Systems **32** (2019)
25. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
26. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
27. Lin, X., Ding, C., Zhang, J., Zhan, Y., Tao, D.: Ru-net: Regularized unrolling network for scene graph generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19457–19466 (2022)
28. Liu, S., Li, F., Zhang, H., Yang, X., Qi, X., Su, H., Zhu, J., Zhang, L.: Dab-detr: Dynamic anchor boxes are better queries for detr. arXiv preprint arXiv:2201.12329 (2022)
29. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
30. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
31. Lu, C., Krishna, R., Bernstein, M., Fei-Fei, L.: Visual relationship detection with language priors. In: European conference on computer vision. pp. 852–869. Springer (2016)
32. Menon, A.K., Jayasumana, S., Rawat, A.S., Jain, H., Veit, A., Kumar, S.: Long-tail learning via logit adjustment. arXiv preprint arXiv:2007.07314 (2020)
33. Nguyen, K., Tripathi, S., Du, B., Guha, T., Nguyen, T.Q.: In defense of scene graphs for image captioning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1407–1416 (October 2021)

34. Qu, R., Hall, O., Bideau, P.K., Ouerfelli-Ethier, J., Rolfs, M., Obermayer, K., Hellwich, O.: Salience-sgg: Enhancing unbiased scene graph generation with iterative salience estimation. arXiv preprint arXiv:2601.08728 (2026)
35. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE transactions on pattern analysis and machine intelligence **39**(6), 1137–1149 (2016)
36. Shi, J., Zhang, H., Li, J.: Explainable and explicit visual reasoning over scene graphs. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8376–8384 (2019)
37. Shit, S., Koner, R., Wittmann, B., Paetzold, J., Ezhov, I., Li, H., Pan, J., Sharifzadeh, S., Kaissis, G., Tresp, V., et al.: Relationformer: A unified framework for image-to-graph generation. In: European conference on computer vision. pp. 422–439. Springer (2022)
38. Sudhakaran, G., Dhami, D.S., Kersting, K., Roth, S.: Vision relation transformer for unbiased scene graph generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 21882–21893 (2023)
39. Tang, K., Zhang, H., Wu, B., Luo, W., Liu, W.: Learning to compose dynamic tree structures for visual contexts. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6619–6628 (2019)
40. Teng, Y., Wang, L.: Structured sparse r-cnn for direct scene graph generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19437–19446 (2022)
41. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
42. Wang, L., Yuan, Z., Chen, B.: Multi-granularity sparse relationship matrix prediction network for end-to-end scene graph generation. In: European Conference on Computer Vision. pp. 105–121. Springer (2024)
43. Xu, N., Liu, A.A., Liu, J., Nie, W., Su, Y.: Scene graph captioner: Image captioning based on structural visual representation. Journal of Visual Communication and Image Representation **58**, 477–485 (2019)
44. Yoon, K., Kim, K., Jeon, J., In, Y., Kim, D., Park, C.: Ra-sgg: retrieval-augmented scene graph generation framework via multi-prototype learning. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 9562–9570 (2025)
45. Zellers, R., Yatskar, M., Thomson, S., Choi, Y.: Neural motifs: Scene graph parsing with global context. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5831–5840 (2018)
46. Zhang, A., Yao, Y., Chen, Q., Ji, W., Liu, Z., Sun, M., Chua, T.S.: Fine-grained scene graph generation with data transfer. In: European conference on computer vision. pp. 409–424. Springer (2022)
47. Zhang, C.B., Zhong, Y., Han, K.: Mr. detr: Instructive multi-route training for detection transformers. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 9933–9943 (2025)
48. Zhang, H., Kyaw, Z., Chang, S.F., Chua, T.S.: Visual translation embedding network for visual relation detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5532–5540 (2017)
49. Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.Y.: Dino: Detr with improved denoising anchor boxes for end-to-end object detection. arXiv preprint arXiv:2203.03605 (2022)

50. Zhao, C., Sun, Y., Wang, W., Chen, Q., Ding, E., Yang, Y., Wang, J.: Ms-detr: Efficient detr training with mixed supervision. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 17027–17036 (2024)
51. Zheng, C., Lyu, X., Gao, L., Dai, B., Song, J.: Prototype-based embedding network for scene graph generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22783–22792 (2023)
52. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159 (2020)