Groupwise Query Specialization and Quality-Aware Multi-Assignment for Transformer-based Visual Relationship Detection

Jongha Kim* Jihwan Park* Jinyoung Park*
Jinyoung Kim Sehyung Kim Hyunwoo J. Kim†
Department of Computer Science and Engineering, Korea University
{jonghakim, jseven7071, lpmn678, k012100, shkim129, hyunwoojkim}@korea.ac.kr

Abstract

Visual Relationship Detection (VRD) has seen significant advancements with Transformer-based architectures recently. However, we identify two key limitations in a conventional label assignment for training Transformer-based VRD models, which is a process of mapping a ground-truth (GT) to a prediction. Under the conventional assignment, an 'unspecialized' query is trained since a query is expected to detect every relation, which makes it difficult for a query to specialize in specific relations. Furthermore, a query is also insufficiently trained since a GT is assigned only to a single prediction, therefore near-correct or even correct predictions are suppressed by being assigned 'no relation (∅)' as a GT. To address these issues, we propose Groupwise Query ***Spe****cia****l****ization and* ***Q****uality-Aware Multi-Assignment (SpeaQ). Groupwise Query Specialization trains a 'specialized' query by dividing queries and relations into disjoint groups and directing a query in a specific query group solely toward relations in the corresponding relation group. Quality-Aware Multi-Assignment further facilitates the training by assigning a GT to multiple predictions that are significantly close to a GT in terms of a subject, an object, and the relation in between. Experimental results and analyses show that SpeaQ effectively trains 'specialized' queries, which better utilize the capacity of a model, resulting in consistent performance gains with 'zero' additional inference cost across multiple VRD models and benchmarks. Code is available at* https://github.com/mlvlab/SpeaQ.

1. Introduction

Visual Relationship Detection (VRD) is the task of detecting instances (*i.e.*, subject, object) and their relation (*i.e.*, predicate) given an image, including Scene Graph Generation (SGG) and Human-Object Interaction (HOI) Detection tasks. The task has a wide range of applications, including image retrieval [13], visual question answering [10, 29, 34] and image captioning [39]. Recently, Transformer-based architectures have been increasingly adopted for VRD tasks [3, 14, 16, 22], demonstrating remarkable performances.

To train Transformer-based VRD models, a label assignment is required, which is a process of mapping a ground-truth (GT) to a prediction. Following DETR [1], the Hungarian matching algorithm [19] has been a standard of label assignment for Transformer-based VRD models. However, we observe that queries trained under a standard label assignment are largely 'unspecialized', therefore leaving a large portion of a model's capacity underutilized. To this end, we first identify two major limitations of a standard label assignment that ends up training unspecialized queries.

Firstly, under a standard assignment, a query is trained to detect every relation rather than focusing on a specific relation. Such multiple roles imposed on a query make it difficult for a query to specialize in a specific role since it provides ambiguous training signals overall. The long-tailed property of relation distributions of VRD benchmarks even aggravates the problem since unbalanced training signals make it harder for a query to successfully balance between multiple relations. Secondly, due to a constraint in a standard assignment that a GT can only be assigned to a single prediction, near-correct or even correct predictions are assigned 'no relation (∅)' as a GT, which provides negative signals that suppress the predictions. For instance, about 45% of high-quality predictions[1] are assigned 'no relation (∅)' as a GT in the case of a model trained on the Visual Genome benchmark. In sum, an unspecialized query is trained due to multiple roles that defer the specialization of a query, and the deficiency in positive training signals under the standard assignment.

*Equal contribution.
†Corresponding author.

[1] A high-quality prediction is defined as a prediction that is correctly classified and overlaps with the GT on subject and object with IoU over 0.6.

To address these limitations, we propose a Groupwise Query **Spe**ci**a**lization and **Q**uality-Aware Multi-Assignment (**SpeaQ**). SpeaQ includes two components: Groupwise Query Specialization and Quality-Aware Multi-Assignment. With Groupwise query specialization, a specific target relation group is designated to a query and a query is trained to only detect relations that belong to a designated relation group. As a result, a query learns a specialized role instead of struggling to learn to detect every relation due to 'specific' training signals provided. Quality-aware multi-assignment further facilitates the training of specialized queries, providing 'abundant' training signals by assigning a GT to multiple high-quality predictions that are significantly close to GT.

Experimental results demonstrate that SpeaQ be applied to various architectures across Scene Graph Generation (SGG) and Human-Object Interaction (HOI) Detection tasks, resulting in a consistent performance gain. Due to the specialized queries, SpeaQ covers a wider range of relations where previous models fail (*e.g.*, rare relations), while also improving the performance on relations where previous models show decent performance. As a result, SpeaQ achieves the best performance in both of the two contradicting R@k and mR@k metrics on the VG benchmark [18], which are biased toward common and rare relations, respectively. Notably, such improvements are achieved without any additional post-processing, model parameters, inference cost, or modification in the inference pipeline compared to the baseline. In sum, our contributions are threefold:

- We introduce a Groupwise Query Specialization, which trains a 'specialized' query by dividing queries and relations into disjoint groups and directing a query solely toward relations in a corresponding relation group.
- We propose a Quality-Aware Multi-Assignment which assigns a GT to multiple predictions considering the triplet-level prediction quality, therefore adaptively providing richer training signals to promising predictions.
- Overall, Groupwise Query **Spe**ci**a**lization and **Q**uality-Aware Multi-Assignment (SpeaQ) effectively trains a specialized query, which better leverages the model capacity and therefore consistently improves performance across multiple VRD models and benchmarks with *zero* additional inference cost.

2. Related Works

2.1. Transformers for Visual Relationship Detection

Visual Relationship Detection (VRD), including Scene Graph Generation (SGG) [18] and Human-Object Interaction (HOI) Detection [2] is the task of detecting triplets existing in an image, where a triplet consists of instances (*i.e.*, subject, object) and a relation between those instances (*i.e.*, predicate). Recently, a line of research developing better Transformer-based [35] architectures for VRD tasks have been conducted [3, 6, 14, 16, 22, 25, 28, 31, 42] following the success of DETR [1]. In this paper, we propose a way to better train Transformer-based VRD models, which can be applied to multiple architectures to better leverage the capacity of those models.

2.2. Effective training of VRD models

To mitigate the long-tailed property of VRD benchmarks, multiple learning strategies have been proposed, including data resampling [5, 21], loss re-weighting [14, 38, 40], and building class-specific classifiers [7]. However, such an approach inevitably results in a loss in common classes since it seeks a trade-off between common and rare classes under the same model capability. Our work differs from these approaches in that ours enhances the model's capability itself by training a specialized query, therefore improving performance across classes regardless of frequency. On the other hand, recent works in object detection tried tailoring a label assignment process for detectors, including multiple works actively providing training signals to predictions with low localization costs [4, 11, 12, 27, 36] on a single object. Motivated by those works, we introduce an enhanced label assignment strategy for VRD tasks that comprehensively considers a triplet-level localization and classification quality, which is an initiative work exploring the better label assignment strategy for VRD tasks.

3. Method

In this section, we briefly introduce the structure of Transformer-based VRD models and the standard label assignment strategy (Sec. 3.1). We then propose a Groupwise Query Specialization that directs a query toward a specific predicate group (Sec. 3.2). We also present a Quality-Aware Multi-Assignment which assigns a GT to multiple high-quality predictions considering the triplet-level prediction quality, and the overall pipeline (Sec. 3.3).

3.1. Preliminary

Transformer-based visual relationship detection. A Visual Relationship Detection (VRD) dataset $\mathcal{D} = \{(\mathcal{I}_i, \mathcal{T}_i)\}_{i=1}^{|\mathcal{D}|}$ consists of pairs of an image $\mathcal{I}_i$ and a corresponding GT set $\mathcal{T}_i$. A GT set $\mathcal{T}_i = \{t_j = (s_j, p_j, o_j)\}_{j=1}^{|\mathcal{T}_i|}$ is a set of GT triplets t_j, where a triplet $t_j = (s_j, p_j, o_j)$ consists of a subject s_j, a predicate p_j and an object o_j. Note that the 'relation' between instances are often termed as 'predicate' in the context of VRD. The subject, predicate and object are represented by bounding boxes $b_j^s, b_j^p, b_j^o \in \mathbb{R}^4$ and class labels c_j^s, c_j^p, c_j^o. Note that $\mathcal{T}_i$ includes 'no relation ($\varnothing$)' label padded to N_t GT labels, so that $|\mathcal{T}_i| = N_q$ holds, where N_q is the number of decoder queries. Given

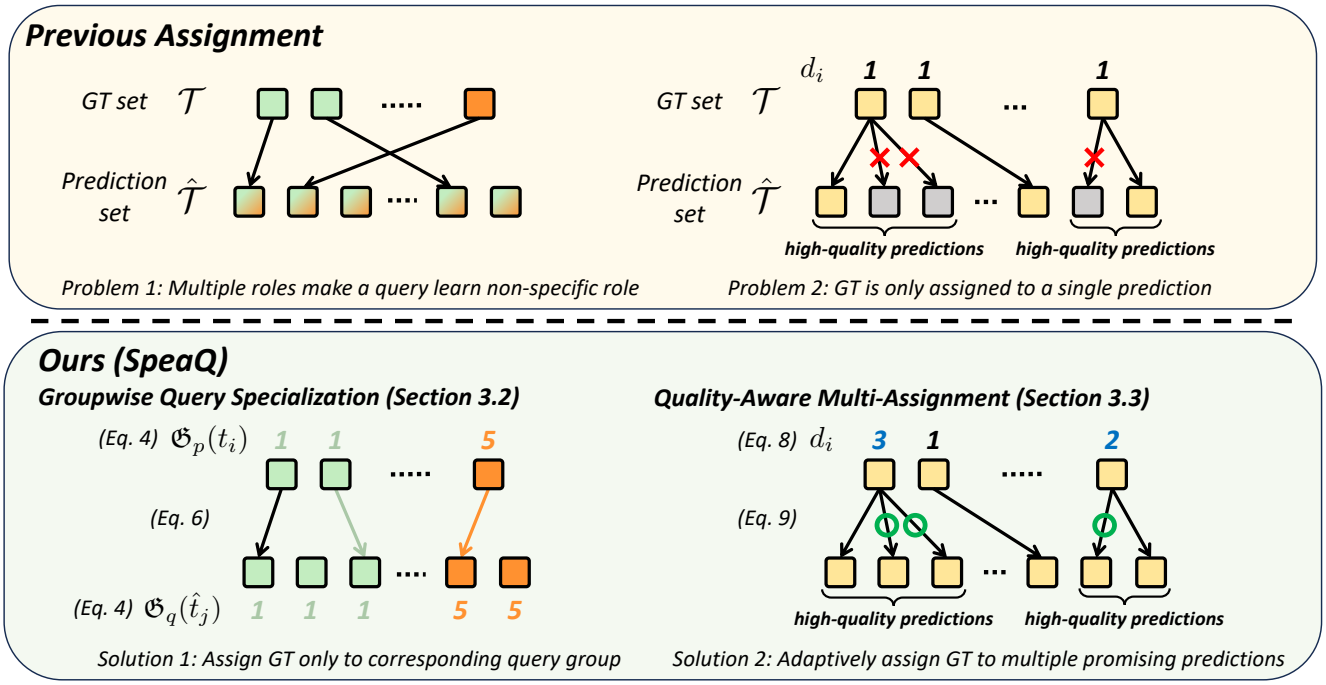


Figure 1. **Overview of the proposed SpeaQ.** SpeaQ consists of two key components: Groupwise Query Specialization and Quality-Aware Multi-Assignment. Groupwise Query Specialization (Sec. 3.2) divides predicates and queries into disjoint predicate groups and query groups and assigns a GT in a specific predicate group only to a query in the corresponding query group, therefore designating a specialized role to a query. Quality-Aware Multi-Assignment (Sec. 3.3) adaptively assigns a GT to a different number of predictions considering overall prediction quality on a subject, object, and predicate to provide richer training supervision to predictions that are close to a GT.

an image $\mathcal{I}_i$, typical Transformer-based VRD models output the set of predictions $\hat{\mathcal{T}}_i = \{\hat{t}_j = (\hat{s}_j, \hat{p}_j, \hat{o}_j)\}_{j=1}^{N_q}$. Typical Transformer-based VRD models consist of a CNN backbone and encoder-decoder Transformers. A conventional CNN backbone network (*e.g.*, ResNet [9]) first generates a visual feature $\mathcal{F}_i \in \mathbb{R}^{C\times H\times W}$ given an input image $\mathcal{I}_i$. Then, the visual feature $\mathcal{F}_i$ is fed into a Transformer encoder which outputs an encoded feature $\mathcal{Z}_i \in \mathbb{R}^{C\times HW}$. Transformer decoders take $\mathcal{Z}_i$ as a feature for cross-attention and transform $\mathcal{Q} = \{q_j\}_{j=1}^{N_q}$, the set of N_q learnable queries into output embeddings. Finally, output embeddings are translated into final predictions, where the set of whole predictions is denoted as $\hat{\mathcal{T}}_i$.

Label assignment for Transformer-based VRD models. Label assignment maps a ground-truth to a prediction to train a Transformer-based VRD model. For the label assignment in Transformer-based architectures, the Hungarian matching algorithm [19] that finds a one-to-one assignment between ground-truths and predictions is widely adopted. Given a GT set $\mathcal{T}_i$ and a prediction set $\hat{\mathcal{T}}_i$, Hungarian matching algorithm finds $\sigma^*_{\text{hungarian}} \in \mathfrak{S}_{N_q}$, the permutation of predictions with the minimal matching cost below:

$$\sigma^*_{\text{hungarian}} = \underset{\sigma\in\mathfrak{S}_{N_q}}{\operatorname{argmin}} \sum_{i}^{N_q} \mathcal{H}_{\text{match}}\left(t_i, \hat{t}_{\sigma(i)}\right), \tag{1}$$

where $\mathcal{H}_{\text{match}}$ is a matching cost between a ground-truth t_i and a prediction $\hat{t}_{\sigma(i)}$ with an index $\sigma(i)$. In VRD tasks, the overall matching cost $\mathcal{H}_{\text{match}}$ is defined as:

$$\begin{aligned}&\mathcal{H}_{\text{match}}\left(t_i, \hat{t}_{\sigma(i)}\right)\\ &= \mathbb{1}_{\{t_i\neq\varnothing\}}\left[\mathcal{C}_s\left(s_i, \hat{s}_{\sigma(i)}\right) + \mathcal{C}_p\left(p_i, \hat{p}_{\sigma(i)}\right) + \mathcal{C}_o\left(o_i, \hat{o}_{\sigma(i)}\right)\right],\end{aligned} \tag{2}$$

where $\mathbb{1}$ is an indicator function and $\mathcal{C}_s, \mathcal{C}_p, \mathcal{C}_o$ denote subject, predicate and object matching cost, respectively. Each matching cost consists of a classification cost (*e.g.*, cross-entropy loss) and the sum of localization costs (*e.g.*, L1 and generalized IoU loss).

3.2. Groupwise Query Specialization

Frequency-based predicate and query grouping. To let a query specialize on specific target predicates, we first divide the set of whole predicate classes into N_g disjoint predicate groups $\{\mathcal{G}_i^p\}_{i=1}^{N_g}$ based on their frequencies. Predicates with similar frequencies are grouped to ensure a more balanced distribution of frequencies within each predicate group, which helps to avoid optimization difficulties caused by the class imbalance. Further details about the predicate grouping are provided in the Sec. A and Alg. 1 of the supplementary material. The set of query $\mathcal{Q}$ with N_q queries is also divided into N_g query groups $\{\mathcal{G}_i^q\}_{i=1}^{N_g}$. To divide the query set $\mathcal{Q}$ into N_g groups, we propose a *proportional query grouping*, where the number of queries in the k'th query group is set proportional to the sum of frequencies of predicates in the k'th predicate group in the training set,

Group	Number of GTs	Number of queries
Group 1	990k (48.4%)	146 (48.7%)
Group 2	398k (19.5%)	58 (19.3%)
Group 3	299k (14.6%)	43 (14.3%)
Group 4	173k (8.5%)	25 (8.3%)
Group 5	186k (9.1%)	28 (9.3%)

Table 1. **Statistics of query groups when N_g = 5.**

formulated as:

$$|\mathcal{G}_k^q| \propto \sum_i^{|\mathcal{D}|} \sum_j^{|\mathcal{T}_i|} \mathbb{1}[c_j^p \in \mathcal{G}_k^p]/|\mathcal{D}|. \tag{3}$$

The detailed algorithm for the proportional query grouping is provided in Alg. 1 of the supplementary material. An example of the sum of predicate frequencies of a predicate group $\mathcal{G}_k^p$ and $|\mathcal{G}_k^q|$, the number of queries in each query group is in Tab. 1. Proportional query grouping enables an output distribution to better resemble the GT distribution by design, where related results are presented in Fig. 2.

Groupwise query specialization. Given N_g query groups $\{\mathcal{G}_i^q\}_{i=1}^{N_g}$ and predicate groups $\{\mathcal{G}_i^p\}_{i=1}^{N_g}$ defined as above, groupwise query specialization forces a query to only detect predicates in the corresponding predicate group. In other words, a query in an i'th query group $\mathcal{G}_i^q$ is forced only to detect predicates in the i'th predicate group $\mathcal{G}_i^p$, instead of struggling to detect predicates in $\mathcal{G}_j^p$ where $i \neq j$. To do so, we first define two mapping functions $\mathfrak{G}_p$ and $\mathfrak{G}_q$. The first function $\mathfrak{G}_p$ returns the index of a predicate group in which a GT t_i with a predicate label c_i^p belongs to. Similarly, the second function $\mathfrak{G}_q$ returns a query group in which a prediction $\hat{t}_j$ from a query q_j belongs to as defined below:

$$\begin{aligned} \mathfrak{G}_p(t_i) &= k \text{ if } c_i^p \in \mathcal{G}_k^p, \\ \mathfrak{G}_q(\hat{t}_j) &= k \text{ if } q_j \in \mathcal{G}_k^q. \end{aligned} \tag{4}$$

If a GT t_i has a predicate label c_i^p that belongs to the k'th predicate group, $\mathfrak{G}_p(t_i) = k$ holds. Similarly, if a predicted triplet $\hat{t}_j$ is from a query q_j that belongs to the k'th query group, $\mathfrak{G}_q(\hat{t}_j) = k$ holds. Based on the mapping functions defined above, a grouping cost $\mathcal{H}_{\text{group}}$ is defined as:

$$\mathcal{H}_{\text{group}}\left(t_i, \hat{t}_j\right) = \begin{cases} 0, & \text{if } \mathfrak{G}_p\left(t_i\right) = \mathfrak{G}_q\left(\hat{t}_j\right) \text{ or } t_i = \varnothing, \\ \infty, & \text{otherwise.} \end{cases} \tag{5}$$

Then, the groupwise query specialization is done by adding the grouping cost $\mathcal{H}_{\text{group}}$ to the original matching cost $\mathcal{H}_{\text{match}}$ as below:

$$\sigma_{\text{spec}}^* = \underset{\sigma \in \mathfrak{S}_{N_q}}{\operatorname{argmin}} \sum_i^{N_q} \mathcal{H}_{\text{match}}\left(t_i, \hat{t}_{\sigma(i)}\right) + \mathcal{H}_{\text{group}}\left(t_i, \hat{t}_{\sigma(i)}\right). \tag{6}$$

Adding $\mathcal{H}_{\text{group}}$ to the original matching cost results in a GT exclusively being assigned to predictions from the query with a group index identical to that of the GT, since the matching cost is set to ∞ otherwise.

3.3. Quality-Aware Multi-Assignment

The conventional assignment has a constraint that a GT can only be assigned to a single prediction. Due to the constraint, near-correct or even correct predictions are suppressed by being assigned a 'no relation ($\varnothing$)' as a GT, which hinders proper training. Therefore, we adopt the multi-assignment, which assigns a GT t_i to d_i number of high-quality predictions instead of only assigning it to a single prediction.

Triplet quality-aware determination of d_i. Since the number of high-quality predictions that correspond to a GT t_i may vary, we determine d_i adaptively considering the overall triplet-level prediction quality of a subject, object, and the predicate on a GT t_i, instead of setting d_i equal for every GT. In detail, vectors that represent subject, object and predicate prediction qualities between a GT t_i and every prediction $\{\hat{t}_j\}_{j=1}^{N_q}$ are firstly calculated as:

$$\begin{aligned} v_i^s &= \left[\text{IoU}\left(b_i^s, \hat{b}_j^s\right)\right]_{j=1}^{N_q} \in \mathbb{R}^{N_q}, \\ v_i^o &= \left[\text{IoU}\left(b_i^o, \hat{b}_j^o\right)\right]_{j=1}^{N_q} \in \mathbb{R}^{N_q}, \\ v_i^r &= \left[\hat{c}_j^p(c_i^p)\right]_{j=1}^{N_q} \in \mathbb{R}^{N_q}, \end{aligned} \tag{7}$$

where IoU is a function that outputs an IoU between two bounding boxes, and $\hat{c}_j^p(c_i^p)$ denotes the predicted probability of a GT predicate label c_i^p of a prediction $\hat{t}_j$. The j'th element in the resulting vectors represents the prediction quality of $\hat{t}_j$ on a GT t_i. Concretely, for a GT triplet t_i, IoU between the GT subject box b_i^s and predicted subject boxes $\hat{b}_j^s$ from every prediction $\hat{t}_j$ is calculated to form a subject quality vector $v_i^s \in \mathbb{R}^{N_q}$, where object quality vector $v_i^o \in \mathbb{R}^{N_q}$ is also analogously defined. Moreover, a predicate quality vector v_i^r is defined as a predicted score of a GT predicate label. Then, d_i is calculated given subject, object and predicate quality vectors as below:

$$\begin{aligned} v_i &= \mathcal{R}\left(v_i^s, v_i^o\right) + \lambda_{\text{rel}} v_i^r \in \mathbb{R}^{N_q}, \\ d_i &= \left\lfloor \max\left(\sum \text{top-k}\left(v_i\right), 1\right)\right\rfloor \in \mathbb{N}, \end{aligned} \tag{8}$$

where $\mathcal{R}$ is the element-wise function (*e.g.*, min, max), and top-k is a function that only retains k largest elements in the vector, and sets the value as zero otherwise. The triplet-level quality vector v_i is firstly obtained by combining the output of a relation function $\mathcal{R}$ and the predicate quality vector v_i^r, and then fed into the top-k function. Then, the floored result of the sum of elements in the resulting vector from the top-k

Method	R@50/100	mR@50/100	AvgR@50/100	F@50/100
X101-FPN backbone				
Motifs [41]	32.1 / 36.9	5.5 / 6.8	18.8 / 21.9	9.4 / 11.5
VCTree [32]	31.8 / 36.1	6.6 / 7.7	19.2 / 21.9	10.9 / 12.7
VCTree-TDE [33]	19.4 / 23.2	9.3 / 11.1	14.4 / 17.2	12.6 / 15.0
VCTree-EBM [30]	20.5 / 24.7	9.7 / 11.6	15.1 / 18.2	13.2 / 15.8
VCTree-BPLSA [8]	21.7 / 25.5	13.5 / 15.7	17.6 / 20.6	16.6 / 19.4
DT2-ACBS [5]	22.0 / 24.4	15.0 / 16.3	18.5 / 20.4	17.8 / 19.5
ResNet-101 backbone				
RelDN [22, 45]	30.3 / 34.8	4.4 / 5.4	17.4 / 20.1	7.7 / 9.3
BGNN [21, 22]	28.2 / 33.8	8.6 / 10.3	18.4 / 22.1	13.2 / 15.8
AS-Net [3]	18.7 / 21.1	6.1 / 7.2	12.4 / 14.2	9.2 / 10.7
SGTR [22]	25.1 / 26.6	12.0 / 14.6	18.6 / 20.6	16.2 / 18.9
HOTR* [16]	22.4 / 27.1	6.9 / 9.7	14.7 / 18.4	10.6 / 14.3
HOTR* + ***SpeaQ (Ours)***	24.7(+2.3) / 29.1(+2.0)	9.6(+2.7) / 12.7(+3.0)	17.2(+2.5) / 20.9(+2.5)	13.8(+3.2) / 17.7(+3.4)
ISG$^{*}_{\dagger}$ [14]	29.5 / 32.1	7.4 / 8.4	18.5 / 20.3	11.8 / 13.3
ISG$^{*}_{\dagger}$+ ***SpeaQ (Ours)***	**32.9**(+3.4) / **36.0**(+3.9)	11.8(+4.4) / 14.1(+5.7)	22.4(+3.9) / 25.1(+4.8)	17.4(+5.6) / 20.3(+7.0)
ISG*	27.2 / 30.1	15.0 / 16.6	21.1 / 23.4	19.3 / 21.4
ISG*+ ***SpeaQ (Ours)***	32.1(+4.9) / 35.5(+5.4)	**15.1**(+0.1) / **17.6**(+1.0)	**23.6**(+2.5) / **26.6**(+3.2)	**20.5**(+1.2) / **23.5**(+2.1)

Table 2. **Performance on Visual Genome.** The best results among models with ResNet-101 backbone are marked in **bold**. * denotes reproduced results. † denotes the performance without loss re-weighting proposed in [14].

function is set as d_i if the sum is larger than 1. Otherwise, d_i is set as 1 to ensure that every GT is assigned to a prediction at least once.

Quality-aware multi-assignment. With d_i calculated in a triplet quality-aware manner as elucidated above, an augmented GT set $\mathcal{T}_{i'}$ is constructed by duplicating t_i for d_i times and padding $\varnothing$ until $|\mathcal{T}_{i'}|$ reaches N_q. Then, the quality-aware multi-assignment is formally defined as:

$$\sigma^*_{\text{multi}} = \underset{\sigma \in \mathfrak{S}_{N_q}}{\operatorname{argmin}} \sum_{i'}^{N_q} \mathcal{H}_{\text{match}}\left(t_{i'}, \hat{t}_{\sigma(i')}\right). \tag{9}$$

Given the objective above, Hungarian algorithm finds the permutation σ^*_{multi} with the lowest matching cost between the augmented GT set $\mathcal{T}_{i'}$ and the prediction set $\hat{\mathcal{T}}_i$.

Final training objective. Proposed groupwise query specialization and quality-aware multi-assignment are combined to form the final assignment objective, dubbed as Groupwise Query **Spe**cia**l**ization and **Q**uality-Aware Multi-Assignment (SpeaQ) as follows:

$$\sigma^* = \underset{\sigma \in \mathfrak{S}_{N_q}}{\operatorname{argmin}} \sum_{i'}^{N_q} \mathcal{H}_{\text{match}}\left(t_{i'}, \hat{t}_{\sigma(i')}\right) + \mathcal{H}_{\text{group}}\left(t_{i'}, \hat{t}_{\sigma(i')}\right). \tag{10}$$

With the final matching cost defined above, the optimal permutation of predictions σ^* with the lowest matching cost is obtained. Then, the final training loss is defined as the sum of the subject, predicate and object loss, $\mathcal{L}_{\text{total}}\left(t_{i'}, \hat{t}_{\sigma^*(i')}\right) = \mathcal{L}_{\text{s}}\left(t_{i'}, \hat{t}_{\sigma^*(i')}\right) + \mathcal{L}_{\text{p}}\left(t_{i'}, \hat{t}_{\sigma^*(i')}\right) + \mathcal{L}_{\text{o}}\left(t_{i'}, \hat{t}_{\sigma^*(i')}\right)$. The subject loss $\mathcal{L}_{\text{s}}\left(t_{i'}, \hat{t}_{\sigma^*(i')}\right)$ is defined as:

$$\begin{aligned} &\mathcal{L}_{\text{s}}\left(t_{i'}, \hat{t}_{\sigma^*(i')}\right) \\ &= \mathcal{L}_{cls}(c^s_{i'}, \hat{c}^s_{\sigma^*(i')}) + \mathbb{1}_{\{t_{i'} \neq \varnothing\}} \mathcal{L}_{box}(b^s_{i'}, \hat{b}^s_{\sigma^*(i')}), \end{aligned} \tag{11}$$

where $\mathcal{L}_{cls}(c^s_{i'}, \hat{c}^s_{\sigma^*(i')})$ is a classification loss (*i.e.*, cross-entropy) between the subject label and the predicted subject logit and $\mathcal{L}_{box}(b^s_{i'}, \hat{b}^s_{\sigma^*(i')})$ is a sum of regression losses (*i.e.*, L1 loss and GIoU loss) between the ground-truth bounding box and the predicted bounding box. $\mathcal{L}_{\text{p}}\left(t_{i'}, \hat{t}_{\sigma^*(i')}\right)$ and $\mathcal{L}_{\text{o}}\left(t_{i'}, \hat{t}_{\sigma^*(i')}\right)$ are analogously defined.

4. Experiments

In this section, we compare the performance of our method SpeaQ with state-of-the-art methods for Scene Graph Generation and Human-Object Interaction Detection tasks. Further implementation details are in the Sec. B of the supplementary material.

4.1. Datasets

Visual Genome. Visual Genome dataset consists of 108k images with 75k objects and 37k predicates. Following previous works [37, 41], we use the subset of Visual Genome (*i.e.*, VG150), which is composed of the most frequent 150 objects and 50 predicate categories. We report the performance on two widely adopted metrics Recall@K (R@K)

Method	Full	Rare	Non-Rare
UnionDet [15]	14.25	10.23	15.46
PastaNet [24]	22.65	21.17	23.09
IDN [23]	23.36	22.47	23.63
HOITrans [46]	23.46	16.91	25.41
HOTR [16]	25.10	17.34	27.42
AS-Net [3]	28.87	24.25	30.25
QPIC [31]	29.07	21.85	31.23
MSTR [17]	31.17	25.31	32.92
CDN [42]	31.44	27.39	32.64
UPT [44]	31.66	25.94	33.36
GEN* [25]	33.12	27.12	34.91
GEN* + ***SpeaQ (Ours)***	**34.00**(+0.88)	**30.20**(+3.08)	**35.13**(+0.22)

Table 3. **Performance on HICO-DET.** Best results are marked in **bold**. * denotes reproduced result.

G	Q	R@100	mR@100	AvgR@100
		32.1	8.4	20.3
	✓	33.1	9.2	21.2
✓		35.3	13.5	24.4
✓	✓	**36.0**	**14.1**	**25.1**

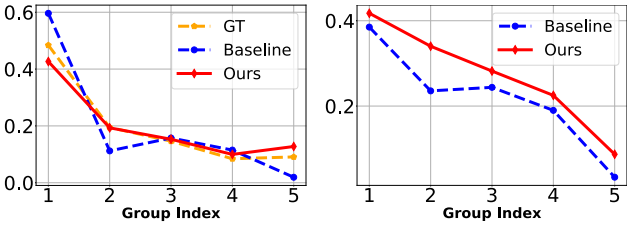
Table 4. **Ablation study on main components.** G : Groupwise Query Specialization (Sec. 3.2), Q : Quality-Aware Multi-Assignment (Sec. 3.3)

and Mean Recall@K (mR@K). Since R@K and mR@K are known to be biased toward the most frequent and the least frequent classes, we also report the arithmetic mean (AvgR@K) [20] and the harmonic mean (F@K) [14, 43] of two metrics to measure the overall balanced performance across predicate frequency.

HICO-DET. For the Human-Object Interaction Detection task, we report the performance on the HICO-DET [2] benchmark, which contains 47k images (37.5k for training and 9.5k for test) with more than 150k annotations of human-object pairs. It has 600 triplet classes, a subset of possible combinations between 80 instance classes and 117 verb classes. We report mAP on three different sets: *full* including all 600 classes, *rare* including 138 classes having less than 10 training instances and *non-rare* including 462 classes with more than 10 training samples.

4.2. Experimental Results

Results on Visual Genome. In Tab. 2, we report the performance of SpeaQ when applied to two competitive Transformer-based models, ISG [14] and HOTR [16]. Applying SpeaQ on ISG results in a gain of 5.4 and 1.0 on R@100 and mR@100, achieving the state-of-the-art result. The result is remarkable in that no previous work has achieved the best performances on both of the two contradicting metrics R@100 and mR@100, which is shown by the best results on AvgR@100 and F@100. Similarly, when applied to HOTR, improvements of 2.0 and 3.0 on R@100 and mR@100 are reported. Consistent gains in both models show the generalizability of the proposed SpeaQ. Note that the boost in performance is gained with *zero* additional model parameters or inference cost.



(a) **Prediction frequency** per group. The closer to the GT, the better.

(b) **mR@100** per group. The higher, the better.

Figure 2. **Prediction frequency and mR@100** per group. Group 1 consists of the most frequent predicates, while group 5 consists of the least frequent predicates.

Results on HICO-DET. We also validate the effectiveness and generalizability of SpeaQ by applying SpeaQ on top of a competitive baseline in the Human-Object Interaction (HOI) Detection task, GEN [25]. The result is reported in Tab. 3. Applying SpeaQ to GEN results in a gain of 0.88, 3.08, and 0.22 in the full, rare, and non-rare sets, respectively. Again, the result shows that applying SpeaQ results in a consistent gain in performance across various tasks and models.

Ablation study on main components. In Tab. 4, the ablation results of the proposed components are reported. Quality-aware multi-assignment consistently boosts both performances on R@100 and mR@100 by 1.0 and 0.8 when applied to the baseline. Also, Groupwise query specialization results in a gain of 3.2 and 5.1 on R@100 and mR@100 when applied to the baseline. With both components combined, the best performance with 36.0 of R@100 and 14.1 of mR@100 is attained.

5. Analysis

In this section, we present various experimental results along with analyses to validate the effectiveness of the SpeaQ. 'Baseline' in all experiments denotes ISG [14]. Note that all experiments are done without the loss reweighting proposed in [14] to focus on the effect of components since it largely biases a model toward rare predicates.

Analyses on output frequency and mR@100 per group. In Fig. 2, the output frequency and mR@100 per predicate group are plotted. Note that we define mR@100 of a group as an average of recall of predicates in a group, similar to the definition of conventional mR@100. By applying SpeaQ, an overprediction of frequent classes and an underprediction of rare classes are relieved as shown in Fig. 2a, resulting in an output distribution (red) closer to the GT distribution (orange) in every group compared to the baseline, which was initially biased toward frequent predicates (blue). The effect is shown in Fig. 2b, where consistent performance gains in every group are reported. It is notable that the per-

Metrics	N_g 1	2	3	4	5	6	k 1	2	3	4	5	6	7
mR@100	8.8	12.2	12.6	14.1	14.2	**14.3**	8.8	12.4	12.9	13.5	13.8	**14.1**	13.8
R@100	33.0	35.6	35.6	**36.0**	35.7	35.7	33.0	35.9	35.8	36.0	**36.1**	36.0	35.9
AvgR@100	20.9	23.9	24.1	**25.1**	25.0	25.0	20.9	24.2	24.4	24.8	25.0	**25.1**	24.9

Table 5. **Performance under different numbers of groups N_g, and k for top-k function in Eq. (8).**

Method	N_q	R@100	mR@100	AvgR@100
Baseline	300	32.1	8.4	20.3
Baseline	600	32.3(+0.2)	7.8(-0.6)	20.1(-0.2)
Ours	300	35.8	13.0	24.4
Ours	600	**36.0**(+0.2)	**14.1**(+1.1)	**25.1**(+0.7)

Table 6. **Performance with a larger number of queries N_q.**

Method	R@100	mR@100	AvgR@100
Baseline	32.1	8.4	20.3
Uniform	33.8	13.2	23.5
Proportional	**36.0**	**14.1**	**25.1**

Table 7. **Ablation study on proportional query grouping.**

formance improves in the most frequent group while the prediction frequency declines, which shows that specialized queries are better at performing a task even with a smaller number of predictions compared to unspecialized queries. Overall, results show that the specialization of queries improves the performance of own target task of a query, and the collection of specialized queries better resembles the GT distribution.

Applying SpeaQ to the model with a larger N_q. In Tab. 6, performances of the baseline and the model trained with SpeaQ under a different number of queries N_q are reported. Naively enlarging N_q under a conventional training scheme results in a drop of 0.6 on mR@100, which is three times larger than the gain of 0.2 on R@100. In contrast, both R@100 and mR@100 are improved by 0.2 and 1.1 as enlarging N_q when trained with SpeaQ. The result validates that SpeaQ is better at fully leveraging a model's capacity by training specialized queries compared to the baseline which fails to successfully handle the model capacity, therefore benefits by scaling up the number of queries.

Analysis on N_g. In Tab. 5, experimental results under different numbers of groups N_g are reported. Compared to the baseline (N_g = 1), the performance gain is reported regardless of N_g. AvgR@100 gradually improves as N_g enlarges, and plateaus at N_g = 4 then slightly decreases afterward. Based on the result, we suppose that partitioning queries into an overly large number of groups may be suboptimal since the lack in the amount of GTs in a group may result in insufficient training signals.

Analysis on k. In Tab. 5, experimental results under different k for a top-k function used to calculate d_i in Eq. (8) are reported. The result shows that providing further positive signals consistently boosts the performance compared to the baseline ($k = 1$), robust to the choice of k. The best performance is achieved when $k = 6$ and the performance slightly decreases afterward, since overly large k may provide positive signals even to non-promising predictions.

Criterion	F	R@100	mR@100	AvgR@100
Baseline	✗	32.1	8.4	20.3
Random	✗	30.7	11.4	21.1
Semantic	✗	31.8	10.0	20.9
BGNN [21]	✓	32.7	12.7	22.7
SHA [7]	✓	34.1	13.4	23.8
Ours	✓	**36.0**	**14.1**	**25.1**

Table 8. **Analysis on predicate grouping criterion.** 'F' denotes that the predicates are grouped on a frequency-basis.

$\mathcal{R}$	R@100	mR@100	AvgR@100
Baseline	32.1	8.4	20.3
min	35.6	12.7	24.2
mean	**36.0**	13.3	24.7
max	**36.0**	**14.1**	**25.1**

Table 9. **Analysis on relation function $\mathcal{R}$.**

Ablation study on proportional query grouping. In Tab. 7, the performances of the proportional query grouping (Eq. (3)) compared to the uniform query grouping are reported, where the uniform query grouping denotes an equal number of queries assigned to every query group. Results show that while a uniform grouping improves the performance compared to the baseline by training specialized queries, the best performance is achieved under the proportional query grouping. Based on the result, we suggest that providing a balanced amount of supervision to every query on average helps better train queries.

Analysis on predicate grouping criterion. In Tab. 8, results under different predicate grouping criteria are reported. 'Random' denotes predicates are randomly grouped into five groups with equal size, and 'semantic' denotes predicates are divided into three groups ('geometric', 'possessive', and 'semantic') by their lexical semantics [41]. Also, we report performance under adopting predicate groups from previous works [7, 21] split on a frequency-basis. Further details about predicate groups are in Sec. C in the supplementary material. Results show that adopting the frequency-basis group as $\mathcal{G}_i^p$ consistently outperforms random or semantic criterion regardless of the choice of the grouping strategy, since it relieves training difficulties

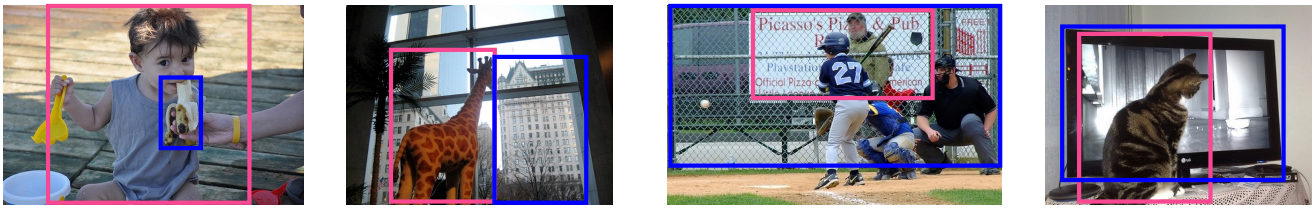


Figure 3. **Qualitative results on Visual Genome dataset.** Predictions of the baseline and the model trained with SpeaQ are visualized along with corresponding ground-truths. Correct and wrong prediction results are marked green and red, respectively.

Type	avg(d_i)	R@100	mR@100	AvgR@100
IoU	23.4	3.2	5.0	4.1
Single	1	32.1	8.4	20.3
Agnostic	3	33.0	9.0	21.0
Ours	3.2	**33.1**	**9.2**	**21.2**

Table 10. **Performances under various label assignment strategies.** avg(d_i) : Average number of predictions a GT is assigned to.

caused by imbalanced training signals, while the proposed grouping criterion results in the best performance.

Analysis on choice of $\mathcal{R}$. In Tab. 9, performances under adopting min, mean, and max as $\mathcal{R}$ in Eq. (8) are reported. The best result is reported in both R@100 and mR@100 when adopting max as $\mathcal{R}$, followed by mean and min functions. Based on the result, we suggest that providing a chance for cases where only a single instance is correctly detected helps the model better learn samples that the model is confused about (*i.e.*, max) than conservatively rewarding 'perfectly' detected cases (*i.e.*, min).

Quantitative results of quality-aware multi-assignment. Experimental results in Tab. 10 support the effectiveness of the quality-aware multi-assignment. We compare performance when adopting conventional assignment, quality-agnostic multi-assignment, and quality-aware multi-assignment as a label assignment strategy, where each is denoted as single, agnostic, and ours. A quality-agnostic multi-assignment denotes that d_i in Eq. (8) is set equal to every GT. We also report the performance under a simple IoU-based assignment commonly adopted in CNN-based detectors, where a GT is assigned to predictions with IoU over 0.5 for both subject and object. The result shows that an IoU-based assignment completely fails in training Transformer-based models. In contrast, quality-agnostic multi-assignment improves the performance compared to a single assignment, while quality-aware multi-assignment further improves the performance showing the best result on R@100 of 33.1 and mR@100 of 9.2. The result shows the effectiveness of multi-assignment, and it could be further improved with quality-aware determination of d_i. For better understanding, we further provide an intuitive running example that demonstrates the importance of the proposed assignment in Sec. D of the supplementary material.

Qualitative results. Fig. 3 presents qualitative examples comparing prediction results from the baseline model and SpeaQ. Regarding samples (a) and (b) of the figure, the model trained with SpeaQ successfully detects challenging samples that require a detailed understanding of both the predicate's semantics and the image. This is in contrast to the baseline model, which struggles in these samples. Furthermore, as shown in samples (c) and (d), SpeaQ helps the model detect less common predicates. Concretely, the model trained with SpeaQ correctly classifies 'behind' and 'in front of', which are 17 and 18 times less frequent compared to 'on' and 'in', predicted by the baseline. These improvements are attributed to queries trained with SpeaQ, which are specialized in target predicates, therefore, are better at detecting rare and challenging samples.

6. Conclusion

In this paper, we propose a Groupwise Query **Spec**i**a**lization and **Q**uality-Aware Multi-Assignment (SpeaQ). The first component trains a 'specialized' query by dividing queries and relations into groups and directing a query in a specific query group solely toward relations in the corresponding relation group. The second component provides abundant training signals considering the triplet-level quality of multiple predictions. Our experiments show that SpeaQ results in performance gains across multiple VRD models and benchmarks with zero additional inference cost.

Acknowledgements. This work was partly supported by ICT Creative Consilience Program through the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT)(IITP-2024-2020-0-01819), the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT)(NRF-2023R1A2C2005373), the Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government (24ZB1200, Fundamental Technology Research for Human-Centric Autonomous Intelligent Systems), and by Neubla.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1, 2

[2] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *WACV*, 2018. 2, 6

[3] Mingfei Chen, Yue Liao, Si Liu, Zhiyuan Chen, Fei Wang, and Chen Qian. Reformulating hoi detection as adaptive set prediction. In *CVPR*, 2021. 1, 2, 5, 6

[4] Qiang Chen, Xiaokang Chen, Jian Wang, Shan Zhang, Kun Yao, Haocheng Feng, Junyu Han, Errui Ding, Gang Zeng, and Jingdong Wang. Group detr: Fast detr training with group-wise one-to-many assignment. In *CVPR*, 2023. 2

[5] Alakh Desai, Tz-Ying Wu, Subarna Tripathi, and Nuno Vasconcelos. Learning of visual relations: The devil is in the tails. In *ICCV*, 2021. 2, 5

[6] Qi Dong, Zhuowen Tu, Haofu Liao, Yuting Zhang, Vijay Mahadevan, and Stefano Soatto. Visual relationship detection using part-and-sum transformers with composite queries. In *ICCV*, 2021. 2

[7] Xingning Dong, Tian Gan, Xuemeng Song, Jianlong Wu, Yuan Cheng, and Liqiang Nie. Stacked hybrid-attention and group collaborative learning for unbiased scene graph generation. In *CVPR*, 2022. 2, 7, 11

[8] Yuyu Guo, Lianli Gao, Xuanhan Wang, Yuxuan Hu, Xing Xu, Xu Lu, Heng Tao Shen, and Jingkuan Song. From general to specific: Informative scene graph generation via balance adjustment. In *ICCV*, 2021. 5

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3, 11

[10] Marcel Hildebrandt, Hang Li, Rajat Koner, Volker Tresp, and Stephan Günnemann. Scene graph reasoning for visual question answering. *arXiv preprint arXiv:2007.01072*, 2020. 1

[11] Qinghang Hong, Fengming Liu, Dong Li, Ji Liu, Lu Tian, and Yi Shan. Dynamic sparse r-cnn. In *CVPR*, 2022. 2

[12] Ding Jia, Yuhui Yuan, Haodi He, Xiaopei Wu, Haojun Yu, Weihong Lin, Lei Sun, Chao Zhang, and Han Hu. Detrs with hybrid matching. In *CVPR*, 2023. 2

[13] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *CVPR*, 2015. 1

[14] Siddhesh Khandelwal and Leonid Sigal. Iterative scene graph generation. In *NeurIPS*, 2022. 1, 2, 5, 6, 11, 12

[15] Bumsoo Kim, Taeho Choi, Jaewoo Kang, and Hyunwoo J. Kim. Uniondet: Union-level detector towards real-time human-object interaction detection. In *ECCV*, 2020. 6

[16] Bumsoo Kim, Junhyun Lee, Jaewoo Kang, Eun-Sol Kim, and Hyunwoo J Kim. Hotr: End-to-end human-object interaction detection with transformers. In *CVPR*, 2021. 1, 2, 5, 6, 11, 12

[17] Bumsoo Kim, Jonghwan Mun, Kyoung-Woon On, Minchul Shin, Junhyun Lee, and Eun-Sol Kim. Mstr: Multi-scale transformer for end-to-end human-object interaction detection. In *CVPR*, 2022. 6

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

[19] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 1955. 1, 3

[20] Lin Li, Long Chen, Yifeng Huang, Zhimeng Zhang, Songyang Zhang, and Jun Xiao. The devil is in the labels: Noisy label correction for robust scene graph generation. In *CVPR*, 2022. 6

[21] Rongjie Li, Songyang Zhang, Bo Wan, and Xuming He. Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *CVPR*, 2021. 2, 5, 7, 11

[22] Rongjie Li, Songyang Zhang, and Xuming He. Sgtr: End-to-end scene graph generation with transformer. In *CVPR*, 2022. 1, 2, 5

[23] Yong-Lu Li, Xinpeng Liu, Xiaoqian Wu, Yizhuo Li, and Cewu Lu†. Hoi analysis: Integrating and decomposing human-object interaction. In *NeurIPS*, 2020. 6

[24] Yong-Lu Li, Liang Xu, Xinpeng Liu, Xijie Huang, Yue Xu, Shiyi Wang, Hao-Shu Fang, Ze Ma, Mingyang Chen, and Cewu Lu. Pastanet: Toward human activity knowledge engine. In *CVPR*, 2020. 6

[25] Yue Liao, Aixi Zhang, Miao Lu, Yongliang Wang, Xiaobo Li, and Si Liu. Gen-vlkt: Simplify association and enhance interaction understanding for hoi detection. In *CVPR*, 2022. 2, 6, 11, 12

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 12

[27] Jeffrey Ouyang-Zhang, Jang Hyun Cho, Xingyi Zhou, and Philipp Krähenbühl. Nms strikes back. *arXiv:2212.06137*, 2022. 2

[28] Jihwan Park, SeungJun Lee, Hwan Heo, Hyeong Kyu Choi, and Hyunwoo J Kim. Consistency learning via decoding path augmentation for transformers in human object interaction detection. In *CVPR*, 2022. 2

[29] Jiaxin Shi, Hanwang Zhang, and Juanzi Li. Explainable and explicit visual reasoning over scene graphs. In *CVPR*, 2019. 1

[30] Mohammed Suhail, Abhay Mittal, Behjat Siddiquie, Chris Broaddus, Jayan Eledath, Gerard Medioni, and Leonid Sigal. Energy-based learning for scene graph generation. In *CVPR*, 2021. 5

[31] Masato Tamura et al. Qpic: Query-based pairwise human-object interaction detection with image-wide contextual information. In *CVPR*, 2021. 2, 6

[32] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *CVPR*, 2019. 5

[33] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *CVPR*, 2020. 5

[34] Damien Teney, Lingqiao Liu, and Anton van Den Hengel. Graph-structured representations for visual question answering. In *CVPR*, 2017. 1

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2

[36] Jianfeng Wang, Lin Song, Zeming Li, Hongbin Sun, Jian Sun, and Nanning Zheng. End-to-end object detection with fully convolutional network. In *CVPR*, 2021. 2

[37] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *CVPR*, 2017. 5

[38] Shaotian Yan, Chen Shen, Zhongming Jin, Jianqiang Huang, Rongxin Jiang, Yaowu Chen, and Xian-Sheng Hua. Pcpl: Predicate-correlation perception learning for unbiased scene graph generation. In *ACM MM*, 2020. 2

[39] Xuewen Yang, Yingru Liu, and Xin Wang. Reformer: The relational transformer for image captioning. In *ACM MM*, 2022. 1

[40] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Bridging knowledge graphs to generate scene graphs. In *ECCV*, 2020. 2

[41] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, 2018. 5, 7

[42] Aixi Zhang, Yue Liao, Si Liu, Miao Lu, Yongliang Wang, Chen Gao, and Xiaobo Li. Mining the benefits of two-stage and one-stage hoi detection. In *NeurIPS*, 2021. 2, 6

[43] Ao Zhang, Yuan Yao, Qianyu Chen, Wei Ji, Zhiyuan Liu, Maosong Sun, and Tat-Seng Chua. Fine-grained scene graph generation with data transfer. In *ECCV*, 2022. 6

[44] Frederic Z Zhang, Dylan Campbell, and Stephen Gould. Efficient two-stage detection of human-object interactions with a novel unary-pairwise transformer. In *CVPR*, 2022. 6

[45] Ji Zhang, Kevin J Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. Graphical contrastive losses for scene graph parsing. In *CVPR*, 2019. 5

[46] Cheng Zou, Bohan Wang, Yue Hu, Junqi Liu, Qian Wu, Yu Zhao, Boxun Li, Chenguang Zhang, Chi Zhang, Yichen Wei, et al. End-to-end human object interaction detection with hoi transformer. In *CVPR*, 2021. 6