Hybrid Reciprocal Transformer with Triplet Feature Alignment for Scene Graph Generation
Appendix

1. Algorithm Details

1.1. Triplet-Guided Relation Learning with Bidirectional Refinement

The principal innovation of our study lies in the model architecture, particularly the triplet-guided relation learning with bidirectional refinement. This involves two interaction paradigms between hybrid triplet-level and component-level representations within a hybrid reciprocal transformer model. The first paradigm utilizes triplet-level representations as conditions to iteratively guide component-level learning within the same layer of the hybrid reciprocal transformer [4]. The second involves bidirectional refinement between triplet-level and component-level representations through the use of cross-attention mechanisms across successive layers of the hybrid reciprocal transformer [10].

The algorithmic pipeline is detailed in Alg. 1. This algorithm integrates two key interaction paradigms: triplet-guided relation learning and a bidirectional refinement mechanism. These paradigms are applied iteratively in each layer of the hybrid reciprocal transformer. Alg. 1 specifically outlines the process of triplet-guided relation learning in Lines 2-9. Here, the decoders $\mathcal{T}_x; x \in \{\text{t, s, o, p}\}$ are tailored to sequentially process triplet masks, subjects, objects, and predicates. This is achieved by splicing the processed results within the current layer of the hybrid reciprocal transformer, thus facilitating iterative component-level learning guided by triplets using processed position embedding $\bar{P}_x^j; x \in \{\text{s,o,p}\}$. Lines 10-13 discuss the implementation of the bidirectional refinement mechanism. This mechanism concurrently advances triplet-level learning and component-level learning, thereby mutually enhancing each aspect. The rationale for this bidirectional refinement is supported by the observation that the processed results for triplet-level representations $\widehat{Q}_{\text{t}}^j$ and component-level representations $\widehat{Q}_x^j; x \in \{\text{s,o,p}\}$ in each layer depict not only identical visual relation tuples but also exhibit inherent connectivity, underscored by the alignment loss.

In the hybrid reciprocal transformer, the process results $\widehat{Q}_x^j; x \in \{\text{t,s,o,p}\}$ for each layer are stored for intermediate supervision in triplet and component learning, respectively. Additionally, $\widehat{Q}_x^j; x \in \{\text{t,s,o,p}\}$ are utilized to initialize the next layer of the hybrid reciprocal transformer, thereby feeding into the subsequent layer from the processed results of the current layer as classic transformer-based detection methods [1]. The final output $\widehat{Q}_x^j; x \in \{\text{t,s,o,p}\}, j \in \{1, ..., M\}$ from multiple layers of the hybrid reciprocal transformer is processed by same prediction heads to supervise both intermediate and final outputs. This enhances model accuracy by providing feedback at various points within the network. During training, independent matching aligns each prediction with its corresponding ground truth, addressing the inherent permutation invariance of outputs characteristic of transformer architectures. Furthermore, independent matching in our hybrid triplet-level and component-level learning ensures that the output

Algorithm 1 Pipeline for Triplet-Guided Relation Learning with Bidirectional Refinement

Require: $\mathcal{T}_x; x \in \{\text{t,s,o,p}\}$
$Q_x^j; x \in \{\text{t,s,o,p}\}, j \in \{1, ..., M\}$
$P_x^j; x \in \{\text{t,s,o,p}\}, j \in \{1, ..., M\}$
Ensure: $\widehat{Q}_x^j \; x \in \{\text{t,s,o,p}\} \; j \in \{1, ..., M\}$
1: **for** $j = 1$ to M **do**
2: $\bar{P}_{\text{t}}^j = P_{\text{t}}^j$
3: $\widehat{Q}_{\text{t}}^j = \mathcal{T}_{\text{t}}^j(\bar{Q}_{\text{t}}^{j-1} | \bar{P}_{\text{t}}^j, \mathcal{I})$
4: $\bar{P}_{\text{s}}^j = P_{\text{s}}^j + \text{FFN}(\text{MHA}(P_{\text{s}}^j, \widehat{Q}_{\text{t}}^j, \widehat{Q}_{\text{t}}^j))$
5: $\widehat{Q}_{\text{s}}^j = \mathcal{T}_{\text{s}}^j(\bar{Q}_{\text{s}}^{j-1} | \bar{P}_{\text{s}}^j, \mathcal{I})$
6: $\bar{P}_{\text{o}}^j = P_{\text{o}}^j + \text{FFN}(\text{MHA}(P_{\text{o}}^j, [\widehat{Q}_{\text{t}}^j, \widehat{Q}_{\text{s}}^j], [\widehat{Q}_{\text{t}}^j, \widehat{Q}_{\text{s}}^j]))$
7: $\widehat{Q}_{\text{o}}^j = \mathcal{T}_{\text{o}}^j(\bar{Q}_{\text{o}}^{j-1} | \bar{P}_{\text{o}}^j, \mathcal{I})$
8: $\bar{P}_{\text{p}}^j = P_{\text{p}}^j + \text{FFN}(\text{MHA}(P_{\text{p}}^j, [\widehat{Q}_{\text{t}}^j, \widehat{Q}_{\text{s}}^j, \widehat{Q}_{\text{o}}^j], [\widehat{Q}_{\text{t}}^j, \widehat{Q}_{\text{s}}^j, \widehat{Q}_{\text{o}}^j]))$
9: $\widehat{Q}_{\text{p}}^j = \mathcal{T}_{\text{p}}^j(\bar{Q}_{\text{p}}^{j-1} | \bar{P}_{\text{p}}^j, \mathcal{I})$
10: $\bar{Q}_{\text{t}}^j = \widehat{Q}_{\text{t}}^j + \text{FFN}(\text{MHA}(\widehat{Q}_t^j, [\widehat{Q}_x^j], [\widehat{Q}_x^j])); x \in \{\text{s,o,p}\}$
11: $\bar{Q}_{\text{s}}^j = \widehat{Q}_{\text{s}}^j + \text{FFN}(\text{MHA}(\widehat{Q}_{\text{s}}^j, \widehat{Q}_{\text{t}}^j, \widehat{Q}_{\text{t}}^j))$
12: $\bar{Q}_{\text{o}}^j = \widehat{Q}_{\text{o}}^j + \text{FFN}(\text{MHA}(\widehat{Q}_{\text{o}}^j, \widehat{Q}_{\text{t}}^j, \widehat{Q}_{\text{t}}^j))$
13: $\bar{Q}_{\text{p}}^j = \widehat{Q}_{\text{p}}^j + \text{FFN}(\text{MHA}(\widehat{Q}_{\text{p}}^j, \widehat{Q}_{\text{t}}^j, \widehat{Q}_{\text{t}}^j))$
14: **end for**
15: **return** $\widehat{Q}_x^j; x \in \{\text{t,s,o,p}\}, j \in \{1, ..., M\}$

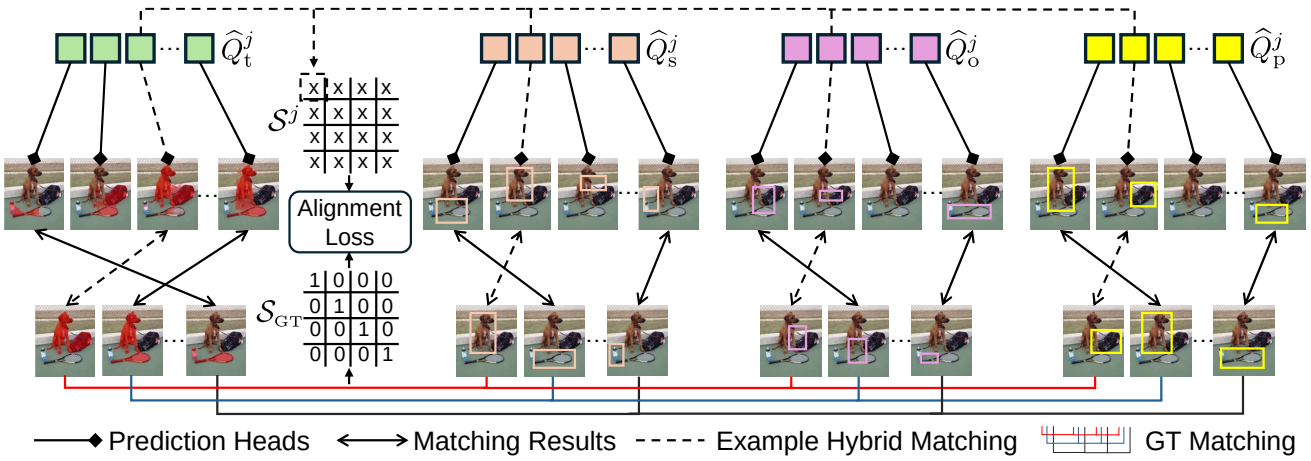

Figure 1. Matching process to calculate $\widetilde{Q}_x^j; x \in \{\text{t,s,o,p}\}$ and calculation process of triplet feature alignment loss. The dashed line illustrates a group of matched hybrid representations from triplet and component levels to calculate cosine similarity in one position of the predicted similarity matrix.

from each layer is individually aligned using an alignment loss. This maintains high alignment performance in calculating alignment loss, independent of other layers' influence. This multi-stage supervision and matching strategy, combined with our proposed triplet-guided relation learning and bidirectional refinement mechanism, aids in early prediction error correction and robust feature representation development. Such integration improves overall detection performance by effectively bridging the gap between hybrid triplet-level and component-level learning representations.

1.2. Triplet Feature Alignment Loss

The proposed alignment loss function is designed to harmonize the triplet-level representations with their corresponding component-level representations. This is achieved by fostering high similarity between matched pairs of triplet-level and component-level representations within the same visual relation tuple, while simultaneously discouraging similarity among unmatched hybrid representation pairs. The alignment loss function plays a critical role in enhancing the distinctiveness of the triplet features corresponding to their component-level representations. This is particularly useful for distinguishing objects that play multiple roles within an image, leveraging uniquely aligned semantic triplet features.

Specifically, the triplet feature alignment loss is computed using the predicted similarity matrix $\mathcal{S}^j$ of each layer of hybrid reciprocal transformer and ground truth similarity matrix $\mathcal{S}_{\text{GT}}$. The alignment loss effectively exploits the inherent one-to-one correspondence between the ground truth triplet mask and the component-level bounding boxes for subjects, objects, and predicates. This process not only facilitates feature distillation across hybrid representations but also enhances the exploration of interconnections among these representations.

To compute the predicted similarity matrix $\mathcal{S}^j$ for the j-th layer outputs of the hybrid reciprocal transformer, we first derive the valid matching output features, denoted as $\widetilde{Q}_x^j; x \in \{\text{t,s,o,p}\}$, from the raw output features $\widehat{Q}_x^j$. The features $\widetilde{Q}_x^j$ represent those query outputs in $\widehat{Q}_x^j$ that successfully matched with the ground truth triplet mask and the respective subject, object, and predicate bounding boxes. Subsequently, we construct the pseudo triplet-level representation $\mathbb{Q}_{\text{t}}^j$ as a weighted sum of the component-level representations $\widetilde{Q}_x^j; x \in \{\text{s,o,p}\}$. This representation $\mathbb{Q}\text{t}^j$ is utilized to compute the cosine similarity with $\widetilde{Q}^j\text{t}$, as outlined in Eq. 8 of our submitted paper.

Ground truth similarity, denoted as $\mathcal{S}_{\text{GT}}$, quantifies the inherent correspondences derived from the integration of triplet masks with their associated bounding boxes for subjects, objects, and predicates. As depicted in Figure 1, $\mathcal{S}_{\text{GT}}$ adheres to a one-hot encoding scheme across each row and column, where each row uniquely corresponds to the combined subject, object, and predicate information encapsulated by a triplet mask. Specifically, if a given row indicative of a triplet mask aligns accurately with its respective visual relation components—subject, object, and predicate—the ground truth assignment is encoded as one; otherwise, it is set to zero.

Figure 1 illustrates the alignment between the ground truth and predicted results. Specifically, for a given ground truth visual relation tuple consisting of a triplet mask and the associated subject, object, and predicate—indicated by dashed lines—we employ both individual triplet mask matching and component-level matching [6] for the sub-

ject, object, and predicate to monitor their respective output features. In the ground truth similarity matrix, the corresponding entries for these paired results are assigned a value of one. Conversely, in the predicted similarity matrix, the same entries are populated with the cosine similarity values computed from the tracked triplet mask representation and pseudo triplet representation from the sum of subject, object, and predicate representations. For the query representation of the triplet mask highlighted by the dashed line in Figure 1, the ground truth similarity values for any other matched subject, object, and predicate from differing paired visual relations are set to zero. This differentiation is crucial to accurately distinguish between the features of each triplet and help different semantics of multi-role objects.

2. Implementation Details

For the experiment in both Visual Genome (VG) dataset [7] and Action Genome (AG) dataset [3], we use exactly the same model architecture. We utilize a ResNet-101 backbone [2] to crop four scales of image features. Then the last scale of the image feature is processed with a six-layer transformer encoder with hidden size 512. For each part in $\mathcal{T}_x; x \in \{\text{t,s,o,p}\}$ of the hybrid reciprocal transformer, we deploy a six-layer transformer decoder architecture. The query number for component-level learning is 600 for $\mathcal{T}_x^j; x \in \{\text{s, o, p}\}$, and 300 for $\mathcal{T}_{\text{tri}}^j$. The triplet matching strategy employs a naive Hungarian algorithm, optimizing for mask loss, dice loss, and classification loss with weights of 5.0, 5.0, 1.0. For component-level matching, the approach adopts a grouping matching strategy as proposed in [6], ensuring consistency of subject, object, and predicate in same matched indices of their decoders. Both triplet mask matching and component subject, object, and predicate matching apply the one-to-many matching mechanism, the actual ground truth similarity matrix follows the extended one-hot formatting, where one value may occur as the same time as the multiple matching number in one-to-many matching to maintain consistency. The model undergoes pretraining on the VG and AG datasets, focusing on object detection and semantic segmentation, respectively. The trained parameters for mask segmentation and box detection are copied into their corresponding position in the hybrid reciprocal transformer to initialize model weights. In the complete model configuration, we employ the AdamW optimizer [9] for training purposes, initiating with a learning rate of 6e-4. This rate is gradually reduced following a cosine annealing schedule [8]. The training process extends over 150,000 steps, consistently applied across both the VG and AG datasets. All experiments are conducted on four A40 GPUs, utilizing a batch size of 16.

3. Ablation Study and Analysis Details

3.1. Hybrid Representation Guidance Order

We reverse the triplet-guide mechanism to observe the interaction among the triplet-level and component-level representations. The reversed mechanism can be shown as:

$$
\begin{aligned}
&\widehat{Q}_x^j = \mathcal{T}_x^j(\bar{Q}_x^{j-1} | \bar{P}_x^j, \mathcal{I}); x \in \{\text{s,o,p,t}\},\\
&\bar{P}_{\text{s}}^j = P_{\text{s}}^j,\\
&\bar{P}_{\text{o}}^j = P_{\text{o}}^j + \mathsf{FFN}(\mathsf{MHA}(P_{\text{o}}^j, \widehat{Q}_{\text{s}}^j, \widehat{Q}_{\text{s}}^j)),\\
&\bar{P}_{\text{p}}^j = P_{\text{o}}^j + \mathsf{FFN}(\mathsf{MHA}(P_{\text{p}}^j, [\widehat{Q}_{\text{s}}^j, \widehat{Q}_{\text{o}}^j], [\widehat{Q}_{\text{s}}^j, \widehat{Q}_{\text{o}}^j])),\\
&\bar{P}_{\text{t}}^j = P_{\text{t}}^j + \mathsf{FFN}(\mathsf{MHA}(P_{\text{t}}^j, [\widehat{Q}_{\text{s}}^j, \widehat{Q}_{\text{o}}^j, \widehat{Q}_{\text{p}}^j], [\widehat{Q}_{\text{s}}^j, \widehat{Q}_{\text{o}}^j, \widehat{Q}_{\text{p}}^j])).
\end{aligned}
$$

This mechanism uses component-level learning representation to guide triplet mask learning. The evaluation metric from R@50/100, and mR@50/100 are 33.2, 36.9, 15.3, and 18.9 respectively. This decreasing performance compared to our vanilla model is attributed to the fact that triplet mask learning representation contains the global feature for a visual relation triplet feature, the global feature can boost the following performance of component-level subject, object, and predicate detection. The reversing process will not leverage the global feature to enhance following component-level learning, and lose some precision in metric.

3.2. Grouping Hybrid Representations in Uniform Matcher

In [6], the authors use a groupwise same matched indices among subject, object, and predicate for the consistency in calculation of scores of visual relation tuples from multiply of subject, object, and predicate scores. Inspired by [6], we explore to group triplet mask with component-level subject, object, and predicate in a hybrid uniform matcher to replace the alignment loss. The hybrid uniform matching of hybrid triplet and component representations refers to all of them follows the same matched indices between output and targets.

Specifically, the hybrid uniform matcher follows the design of [6] but add extra optimization target for triplet mask, which contains mask loss, dice loss, and classification loss with weights of 5.0, 5.0, 1.0. We edit the query number for both triplet mask and component subject, object, and predicate as 300 in the experiment. Table 3 in original paper presents a comparative analysis of the outcomes, employing varied matching ratios between component and triplet levels in the uniform hybrid matcher. The study's results suggest that integrating hybrid representations through a hybrid uniform matching mechanism results in performance that is suboptimal when compared to that of our vanilla model. More precisely, diminishing the granularity of component-level matching adversely affects the effectiveness of the

Figure 2. Visualization results of the triplet mask from our method, and its corresponding visual relation in VG datasets.

system. The experimental results demonstrate that there is a significant discrepancy between the triplet-level and component-level representations due to their distinct patterns. Directly matching them with hard constraints via a uniform matcher can adversely impact the overall efficacy of the learning process.

3.3. Matching of Hybrid Representations in Inference Stage

Traditional one-stage scene graph methods [4–6] use the multiply results from bitwise subject, object, and predicate classification scores to crop the top part to organize final visual relation detection results. Due to our extra triplet mask representation for each visual relation compared to traditional methods, we explore the potential to multiply triplet mask scores with previous component-level multiply results together to calculate the final scores for each detected visual relation.

In our submission paper, we compare three different model variants as outlined in Table 5. The first variant, referred to as our vanilla model, solely multiplies component-level scores. The second variant multiplies the scores of triplet masks directly with component scores. The third variant employs Hungarian matching to align the order of triplet mask scores with that of component-level predictions before multiplication. Our experimental results indi-

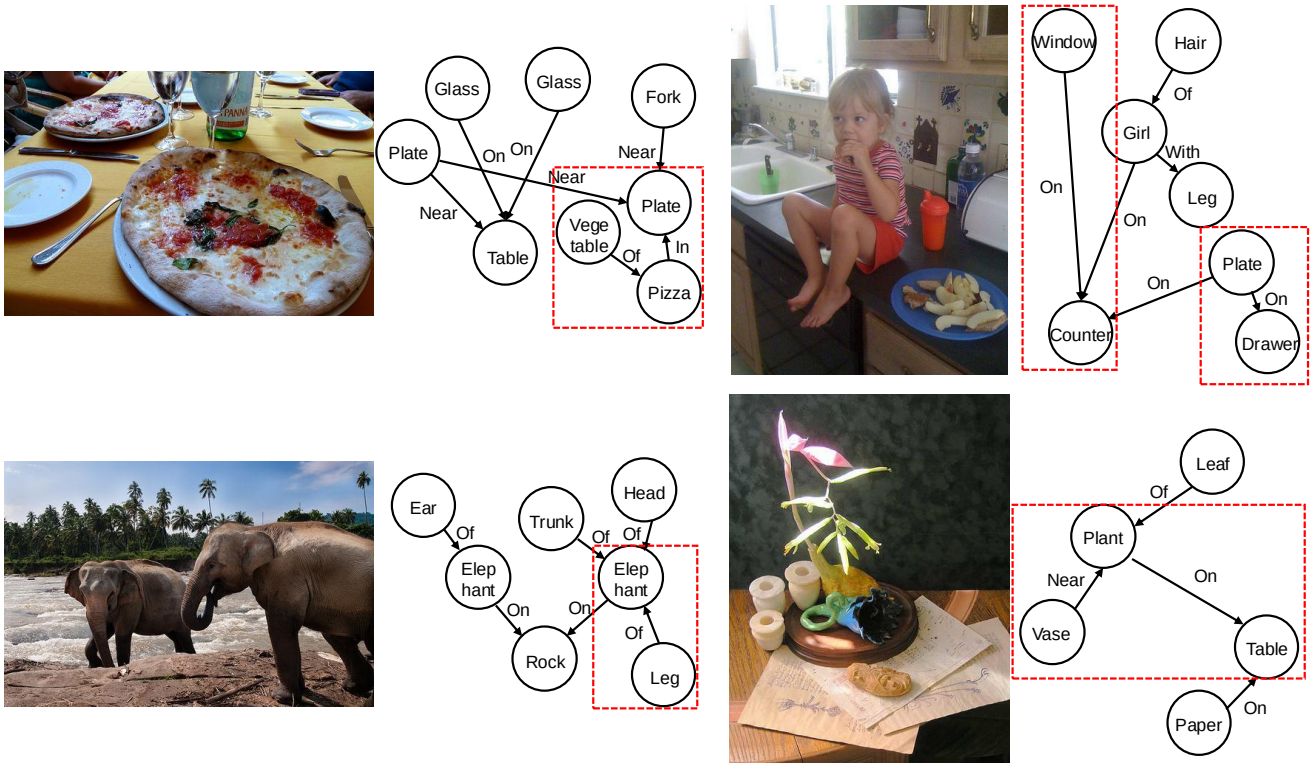

Figure 3. Qualitative results of our methods on VG datasets. Left part is the original image, right part is the constructed scene graph by our method. Dashed rectangle marks our correct detected visual relation that is not annotated in ground truth.

cate that direct multiplication negatively impacts the overall performance of scene graph generation. This degradation is attributed to the triplet masks following a different order compared to the component-level results, thereby introducing noise during the multiplication process and disrupting the ordering of visual relation tuples. Although the third variant demonstrates a slight improvement over the vanilla model—attributable to Hungarian matching, which realigns the triplet mask scores to match the distribution of component-level results—we opt not to integrate this variant into the vanilla model due to its increased inference time, resulting in lower frames per second (FPS). Specifically, across all the VG test dataset, while our vanilla model achieves 11.4 FPS, the third variant reaches only 2.4 FPS due to the matching process in the inference stage.

4. Qualitative Results

4.1. Triplet Mask with Visual Relation

We present the output results of our triplet mask alongside the corresponding visual relation tuples in Figure 2. The visualization demonstrates that our method can generate various triplet masks and precisely assigns the triplet mask to specific visual relation tuples across various scenarios. This precision allows for the accurate delineation of the subject, object, and predicate components within our hybrid reciprocal transformer, where the triplet mask features help to leverage global information at the triplet level. Furthermore, based on the accurate triplet masks output, our novel alignment loss not only aligns the triplet and component features but also enables the utilization of aligned triplet features to differentiate the semantic meanings of multi-role objects, thereby enhancing the performance of scene graph generation.

4.2. Scene Graph

We present detailed visualization results of the scene graph generation by our method in Figure 3. The red dashed rectangle highlights the visual relations correctly identified by our method but not annotated in the ground truth. Our method is capable of identifying challenging visual relationships, such as $\{\textit{vegetable-of-pizza}\}$, which exemplifies the effectiveness of our approach in extreme scenarios. These qualitative results demonstrate that our method can accurately construct scene graphs across multiple images, thereby significantly enhancing the understanding of the entire visual relationship graph.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[3] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10236–10247, 2020. 3

[4] Siddhesh Khandelwal and Leonid Sigal. Iterative scene graph generation. *Advances in Neural Information Processing Systems*, 35:24295–24308, 2022. 1, 4

[5] Bumsoo Kim, Junhyun Lee, Jaewoo Kang, Eun-Sol Kim, and Hyunwoo J Kim. Hotr: End-to-end human-object interaction detection with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 74–83, 2021.

[6] Jongha Kim, Jihwan Park, Jinyoung Park, Jinyoung Kim, Sehyung Kim, and Hyunwoo J Kim. 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*, pages 28160–28169, 2024. 2, 3, 4

[7] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017. 3

[8] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts, 2017. 3

[9] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. 3

[10] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 1