Salience-SGG: Enhancing Unbiased Scene Graph Generation with Iterative Salience Estimation

Supplementary Material

Overview of Supplementary Material

This supplementary material provides detailed information not covered in the main manuscript due to space constraints.

- The implementation details of our Salience-SGG on three datasets mentioned in the main paper.
- The implementation of Iterative Salience Decoder (ISD) with pre-trained IETrans [23] and TDE [18].
- The hyperparameters analysis of our Salience-SGG.
- The efficiency analysis of our ISD.
- The details of the top_down salience labels and the statistical differences with the bottom-up label.
- Spatial structure learning and predicate understanding analysis.
- A detailed comparison to SOTA Methods on VG.
- Qualitative comparisons.

A. Implementation Details

For the three datasets, the same settings are used, with the exception of the training epochs. First, a deformable DETR with ResNet-50 [4] as backbone is pre-trained on each SGG dataset for 25 epochs. The number of entity queries, N_e, is set to 200. All remaining hyperparameters (*e.g.* transformer layers, feature dimension, optimizer, matching weight, and loss weight) are consistent with the work of [25]. The matching results from the last decoder layer in the deformable DETR are utilized to create predicate labels $G^{'}$. Subsequently, the pre-trained object detector is frozen during the joint training of the predicate decoder and ISD, since no significant improvement is observed when fine-tuning it. The loss ratio between salience loss $\mathcal{L}_{salience}$ and predicate loss $\mathcal{L}_{pre}$ is $1:1$. For the VG dataset, the ISD and the predicate decoder are trained for 13 epochs, with a reduction in learning rate by factor 10 in the 11-th epoch. For OIv6, the total number of training epochs is 8, and the learning rate reduces in the 6-th epoch. Finally, for GQA-200, the model is trained for 12 epochs, with the learning rate dropping in the 10-th epoch. For all experiments the random seed is set to 42.

B. Implementation of ISD with Other Debiasing Strategies

In this section, we describe the implementation of combining our ISD with the existing Unbiased-SGG models, *i.e.* TDE [18] and IETrans [23]. Given the pre-trained Unbiased-SGG model, the entity boxes B, entity features Q, entity category distributions C, and predicted predicates G may be obtained. For all proposed entity pairs, the triplet salience labels $M^{'}$ may be constructed as described in the main paper. In this particular context, the entity features Q refer to the entity region of interest features from Faster R-CNN [17]. During the training of ISD, the pre-trained Unbiased-SGG model is kept frozen. It is important to note that in the ISD training, there is no sub-sampling applied on the salience labels.

L	R@50	R@100	mR@50	mR@100	F@50	F@100
1	27.5	32.0	15.9	19.4	20.1	24.1
2	28.2	32.7	16.0	19.6	20.4	24.5
3	28.6	33.1	17.3	21.3	21.6	25.9
4	28.8	33.4	18.0	21.6	22.1	26.2

Table A1. Performance with various number of ISD layers L.

$\mathcal{T}$	R@50	R@100	mR@50	mR@100	F@50	F@100
0.1	26.9	30.8	18.0	21.7	21.6	24.9
0.2	27.4	31.8	17.9	21.6	21.7	25.7
0.3	27.7	32.2	17.9	21.3	21.7	25.6
0.4	28.1	32.6	17.7	21.5	21.7	25.9
0.5	28.6	33.3	17.8	21.5	22.0	26.1
0.6	28.8	33.4	18.0	21.6	22.1	26.2
0.7	28.6	33.1	17.5	21.3	21.7	25.9
0.8	28.4	32.7	17.1	20.5	21.3	25.8
0.9	28.0	32.2	17.1	20.0	21.2	24.7

Table A2. Performance with the varying values of $\mathcal{T}$.

C. Hyperparameter Analysis

In Salience-SGG, there are two important hypeparameters $\mathcal{T}$ and L. Table A1 illustrates the performance of Salience-SGG with different L. The performance gradually increases with L becoming larger. Considering the training efficiency, we do not verify the performance when L is larger than 4. Table A2 shows the influence of $\mathcal{T}$. when $\mathcal{T}$ decreases from 0.9 to 0.6, performance gradually improves. For instance, R@100 increases from 32.2 to 33.4 and mR@100 from 20.0 to 21.6. When $\mathcal{T}$ keeps decreasing, there are no transparent changes on mR@K, whereas R@K significantly drops from 33.4 to 30.8. A lower $\mathcal{T}$ indicates a weak spatial structure supervision, which is more likely to result in salience insensitivity. This observation aligns with our motivation: Maintaining salience sensitivity enhances the robustness of Unbiased-SGG models to debiasing strategies.

Model	FLOPs (G)	Peak Memory Allocated (MB)	Latency (ms)
Salience-SGG	242	1471	75.3
Salience-SGG w/o ISD	235	1385	58.8

Table A3. Cost and efficiency analysis of ISD

top_down$_{gt}$	top_down$_{entity}$	top_down$_{triplet}$	bottom_up
5.5	12.1	33.4	267

Table A4. Average number of each salience label on one image.

D. Efficiency Analysis

To evaluate the efficiency cost of our ISD, We measure the Floating Point Operations (FLOPs), Peak Allocated Memory and Inference Latency of the Salience-SGG with and without ISD module as illustrated in Table A3. The ISD module introduces 7G FLOPS, 96MB Peak Allocated Memory, and an increase in inference latency of 16.5ms.

E. Salience Label Analysis

In the main paper, we compare the performances of Salience-SGG supervised by multiple salience labels. In this supplementary material, we provide the details of the construction process of each top-down salience label. The top_down_{gt} is not special which is a direct mapping from the predicate label to a binary mask. The top_down_{entity} label is created by first performing one-to-many Hungarian Matching [10] at the entity level. Each annotated entity may correspond to multiple detected entities. Salience connections are built between every pair of detected entities based on the matching results and the ground-truth triplets. Finally, the salience connections are selected using the bottom-up salience label $M^{'}$ to produce the final top_down_{entity}. $top\_down_{triplet}$ label is constructed by performing one-to-many matching between the predicted predicates and the ground-truth predicates. Similarly to the top_down_{entity}, a matrix reflecting the connections between detected entities can be constructed based on the matching results. Ultimately, the connections in the matrix are further selected by the bottom-up salience label. Table A4 shows the average number of each kind of salience label on one image in the training data.

F. Spatial structure learning and predicate understanding

In our one-stage Salience-SGG setup, the predicate decoder and ISD are trained jointly. The ISD is equipped with a P-ECA, which may regulate the predicate prediction by the salience loss. To investigate whether training ISD would be beneficial for the predicate understanding of the predicate decoder, we compare the performances ranked solely on salience score with those on salience score in conjunction with object, subject, and predicate scores. The results of Salience-SGG, Salience-SGG w/o G-ESA and Salience-SGG w/o P-ECA are reported in Table A5. Salience-SGG w/o G-ESA and Salience-SGG w/o P-ECA models demonstrate comparable performance ranked on complete scores. However, it should be noted that there is a significant discrepancy between their performances ranked merely on the salience scores. This observation indicates that Salience-SGG without P-ECA benefits from enhanced ability to capture salient spatial structures, while Salience-SGG without G-ESA benefits from better predicate understanding. In conclusion, the final Salience-SGG improves the results by combining two benefits.

G-ESA	P-ECA	R@100 (M)	mR@100 (M)	R@100 (F)	mR@100 (F)
✓	×	29.8	17.6	31.2	20.6
×	✓	28.3	15.6	32.3	20.5
✓	✓	29.7	17.3	33.4	21.6

Table A5. Ablation study on spatial structure learning and predicate understanding. The upper alphabets behind metrics indicate the ranking scores. 'M' indicates salience scores only, while 'F' refers to the full scores (*i.e.* object scores * subject scores * predicate scores * salience scores).

G. Comparison to SOTA Methods on VG

In this section, we privde a more extensive comparsion to SOTA methods on VG dataset as illustrated in Table A6.

H. Qualitative Comparisons

In this section, we provide more qualitative comparisons between IETrans [23] and IETrans+ISD, TDE [18] and TDE+ISD, as illustrate in Figure F1.

References

[1] Minghan Chen, Guikun Chen, Wenguan Wang, and Yi Yang. Hydra-sgg: Hybrid relation assignment for one-stage scene graph generation. *arXiv preprint arXiv:2409.10262*, 2024. 3

[2] Yuren Cong, Michael Ying Yang, and Bodo Rosenhahn. Reltr: Relation transformer for scene graph generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):11169–11183, 2023. 3

[3] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19427–19436, 2022. 3

[4] 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. 1

Method	Backbone	# params (M)	R@20	R@50	R@100	mR@20	mR@50	mR@100	F@20	F@50	F@100
two-stage models											
Motifs* [22] [cvpr2018]	ResNeXt101-FPN	369.9	25.5	32.8	37.2	5.0	6.8	7.9	8.4	11.3	13.0
TDE† [18] [cvpr2020]	ResNeXt101-FPN	369.9	11.9	16.6	20.2	6.6	8.9	11.0	8.5	11.7	14.3
BGNN [13] [cvpr2021]	ResNeXt101-FPN	341.9	23.3	31.0	35.8	7.5	10.7	12.6	11.3	15.9	18.6
GCL† [3] [cvpr2022]	ResNeXt101-FPN	-	-	-	18.4	22.0	12.9	16.8	19.3	-	17.6
SHA [3] [cvpr2022]	ResNeXt101-FPN	-	-	-	14.9	18.2	**14.2**	17.9	20.9	-	16.3
NICE† [12] [iccv2022]	ResNeXt101-FPN	-	-	-	27.0	30.8	-	11.9	14.1	-	16.5
IETrans† [23] [eccv2022]	ResNeXt101-FPN	369.9	17.5	23.5	27.3	11.0	15.7	18.2	**13.5**	18.8	21.8
EICR† [15] [iccv2023]	ResNeXt101-FPN	-	-	-	27.9	32.2	-	15.5	18.2	-	19.9
SQUAT [6] [cvpr2023]	ResNeXt101-FPN	-	-	-	24.5	28.9	-	14.1	16.5	-	17.9
PE-NET [24] [cvpr2023]	ResNeXt101-FPN	-	-	-	26.5	30.9	-	16.7	20.9	-	20.5
ST-SGG† [9] [iclr2024]	ResNeXt101-FPN	-	-	-	26.7	30.7	-	11.6	14.2	-	16.2
DRM [11] [cvpr2024]	ResNeXt101-FPN	-	-	-	19.0	22.9	-	**20.4**	**24.1**	-	**20.8**
RA-SGG [21] [aaai2025]	ResNeXt101-FPN	-	-	-	26.0	30.3	-	14.4	17.1	-	18.5
SRD† [16] [wacv2025]	ResNeXt101-FPN	-	-	-	-	-	13.5	17.9	20.6	-	-
one-stage models											
SGTR [14] [cvpr2022]	ResNet101	117.1	-	20.6	25.0	-	15.8	20.1	-	17.9	22.3
ISG [7] [neurips2022]	ResNet101	93.5	21.8	27.1	29.7	11.2	15.6	17.1	14.8	19.8	21.7
SSR-CNN [19] [cvpr2022]	ResNeXt101-FPN	274.3	18.4	23.3	26.5	**13.5**	17.9	21.4	15.6	20.2	23.7
RelTR* [2] [tpami2023]	ResNet50	63.7	21.2	27.5	-	6.8	10.8	-	10.3	15.5	-
EGTR [5] [cvpr2024]	ResNet50	42.5	22.4	28.2	31.7	8.8	14.0	18.3	12.6	18.7	23.2
Mg-RMPN [20] [eccv2024]	ResNet50	-	22.5	29.1	33.5	10.3	14.4	17.3	14.1	19.3	22.8
SpeaQ [8] [cvpr2024]	ResNet101	93.4	25.1	32.1	35.5	10.1	15.1	17.6	14.4	20.5	23.5
Hydra-SGG [1] [iclr2025]	ResNet50	67.6	21.9	28.6	33.4	10.3	15.9	19.4	14.0	20.5	24.7
Salience-SGG (Ours)	ResNet50	77.7	21.9	28.8	33.4	12.8	**18.0**	**21.6**	**16.2**	**22.1**	**26.2**

Table A6. Comparison with SOTA methods evaluated on the VG test dataset. The methods are divided into two groups. The best and second-best results in each group are indicated with **bold** and underlined text, respectively. '*' denotes the performance without any debiasing strategy. '†' indicates the methods are combined with MOTIFS [22].

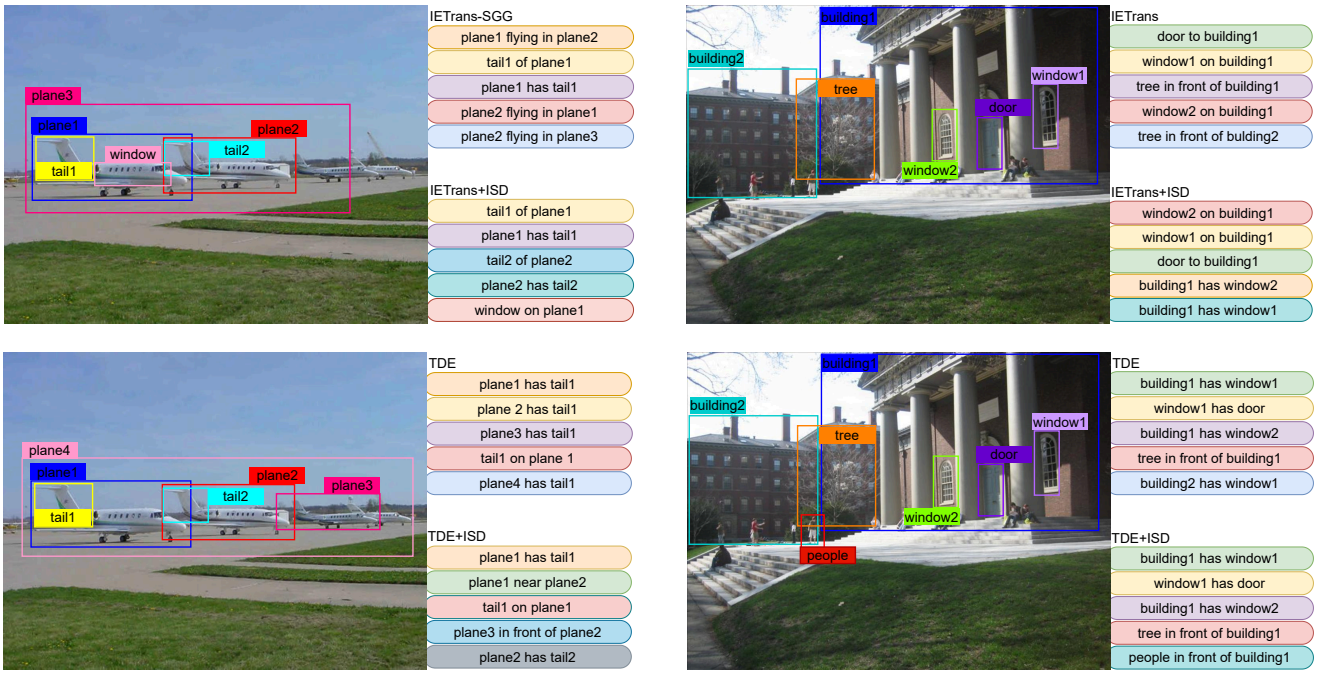

Figure F1. Comparison of the top-5 detected triplets. For each image, the colors indicate triplet identifications; identical colors before and after re-ranking refer to the same triplets.

[5] Jinbae Im, JeongYeon Nam, Nokyung Park, Hyungmin Lee, and Seunghyun Park. Egtr: Extracting graph from transformer for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*

Recognition, pages 24229–24238, 2024. 3

[6] Deunsol Jung, Sanghyun Kim, Won Hwa Kim, and Minsu Cho. 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*, pages 18664–18674, 2023. 3

[7] Siddhesh Khandelwal and Leonid Sigal. Iterative scene graph generation. *Advances in Neural Information Processing Systems*, 35:24295–24308, 2022. 3

[8] 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. 3

[9] Kibum Kim, Kanghoon Yoon, Yeonjun In, Jinyoung Moon, Donghyun Kim, and Chanyoung Park. Adaptive self-training framework for fine-grained scene graph generation. In *ICLR*, 2024. 3

[10] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 2

[11] Jiankai Li, Yunhong Wang, Xiefan Guo, Ruijie Yang, and Weixin Li. Leveraging predicate and triplet learning for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28369–28379, 2024. 3

[12] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18869–18878, 2022. 3

[13] Rongjie Li, Songyang Zhang, Bo Wan, and Xuming He. 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*, pages 11109–11119, 2021. 3

[14] Rongjie Li, Songyang Zhang, and Xuming He. Sgtr: End-to-end scene graph generation with transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19486–19496, 2022. 3

[15] Yukuan Min, Aming Wu, and Cheng Deng. Environment-invariant curriculum relation learning for fine-grained scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13296–13307, 2023. 3

[16] Thanh-Son Nguyen, Hong Yang, and Basura Fernando. Effective scene graph generation by statistical relation distillation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 8420–8430. IEEE, 2025. 3

[17] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 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. 1

[18] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3716–3725, 2020. 1, 2, 3

[19] Yao Teng and Limin Wang. Structured sparse r-cnn for direct scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19437–19446, 2022. 3

[20] Lei Wang, Zejian Yuan, and Badong Chen. Multi-granularity sparse relationship matrix prediction network for end-to-end scene graph generation. In *European Conference on Computer Vision*, pages 105–121. Springer, 2024. 3

[21] Kanghoon Yoon, Kibum Kim, Jaehyeong Jeon, Yeonjun In, Donghyun Kim, and Chanyoung Park. Ra-sgg: Retrieval-augmented scene graph generation framework via multi-prototype learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9562–9570, 2025. 3

[22] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5831–5840, 2018. 3

[23] 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 *European conference on computer vision*, pages 409–424. Springer, 2022. 1, 2, 3

[24] Chaofan Zheng, Xinyu Lyu, Lianli Gao, Bo Dai, and Jingkuan Song. Prototype-based embedding network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22783–22792, 2023. 3

[25] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 1