Supplementary Material for CVPR 2021 paper #2839

Anonymous CVPR 2021 submission

Paper ID 2839

1. Relation Confidence Estimation Module

As introduced in the main paper, the RCE module is an important parts of our method. To demonstrate the effectiveness of the RCE module, we further introduce the learning details of the RCE module and performance comparison with the similar model proposed by previous works.

1.1. Learning

We use a supervised learning strategy to train the RCE module of BGNN, in which the predicate class labels (which predicate category and whether it is valid predicate or background) are used for supervision. Different from the cross-entropy loss $\mathcal{L}_p$ used for the final predicate predictions, we develop a multi-task loss $\mathcal{L}_{rce}$ for the RCE module. Specifically, we use two confidence predictions from the RCE: multi-categories confidence score $\mathbf{s}^m \in \mathbb{R}^{C_p}$ and binary confidence score s^b. We define two focal losses, $\mathcal{L}_m, \mathcal{L}_b$ on the confidence predictions of M predicate proposals $\{\mathbf{s}_1^m, ...\mathbf{s}_M^m\}, \{s_1^b, ...s_M^b\}$, respectively. Formally:

$$\mathcal{L}_m = -\alpha \frac{1}{M} \sum_{k}^{M} \sum_{i}^{|C_p|} \mathbf{y}_{k,i} (1 - \mathbf{s}_{k,i}^m)^{\gamma} \cdot \log(\mathbf{s}_{k,i}^m) \quad (1)$$

$$\mathcal{L}_b = -\alpha \frac{1}{M} \sum_{k}^{M} y\prime_k (1 - s_k^b)^{\gamma} \cdot \log(s_k^b) \quad (2)$$

where $\mathbf{y}_k$ is one-hot vector and $y\prime_k$ is binary label of positive predicate proposals. α, γ are the hyper-parameters.

1.2. Performance

To demonstrate the effectiveness of the RCE module on removing negative predicate proposals, we use the **AUC** to measure its performance, and compare it with two alternatives: *production of entities prediction score* and *relation proposal network* proposed by Graph-RCNN. The AUC of those three methods are **0.839, 0.629, 0.671**, respectively on the validation set of VG, which indicates RCE module is more effective than previous works.

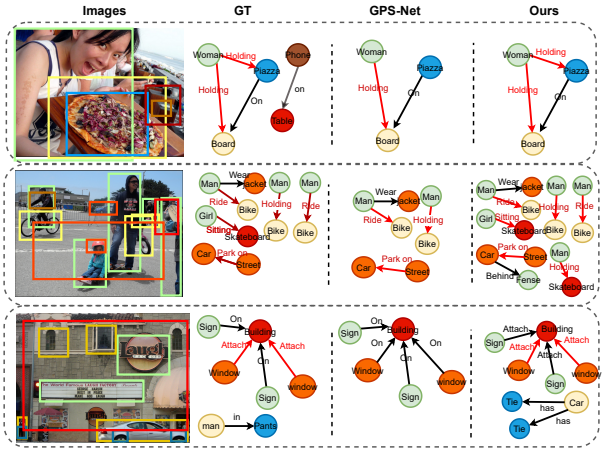

Figure 1: **Qualitative comparisons between our method and GPS-Net† in the SGGen setting.** The predicates in *body* and *tail* categories group are marked as red color. We also show the reasonable relationships detected by models which are not included in GT.

2. Model Comparison w/ Resampling

We note that we have reported the SOTA with recent RFS resampling in Tab.1. To further demonstrate the effectiveness of our BGNN, we add our BLS to other recent methods (reimplemented GPS-Net and MSDN) and perform comparisons on the SGGen task as below:

Models	SGGen				
	mR@100	R@100	Head	Body	Tail
GPS-Net $_{w/\ BLS}$	11.4	34.3	32.3	9.9	4.0
MSDN $_{w/\ BLS}$	11.8	34.4	32.4	10.5	5.1
BGNN $_{w/\ BLS}$	**12.6**	**35.8**	**34.0**	**12.9**	**6.0**

Table 1: **The performance comparison between SOTA with our BLS.**

The results show that our BGNN still outperforms other approaches under the same resampling strategy. In addition, we emphasize that the bi-level resampling is also our

main contribution, and the above results demonstrate its effectiveness for improving all three methods.

3. Quantitative Studies

We extend the quantitative studies as a supplement to the main paper. In this section, we show the detail of long-tail parts partition, and performance comparison on each long-tail part in Sec 3.1. For the fair comparison with the previous methods, we also show the per-class performance comparison on the PredCls subtask in Sec 3.2. In Sec 3.3, we show the comparison of model prediction by visualizing the scene graph generated by BGNN and previous SOTA GPS-Net.

3.1. Long-tail Categories Groups Partition

Visual Genome First, we report the data distribution and long-tail categories set partition detail of Visual Genome [3, 7] in figure 2. We divide the categories into three disjoint groups according to the instance number in training split: *head*(more than 10k), *body*(0.5k $\sim$ 10k), *tail*(less than 0.5k)

We further present the performance comparison of the baseline model(MSDN) between our upper bound assumption referred to Sec. 1 of main paper. The result indicates reducing noise in context modeling improve the baseline model with a large margin especially on tail categories, which only have several data points.

Open Images The long-tail categories group partition and per-class performance comparison on Open Images dataset are reported in Fig. 3. Similarly, we divide the categories of Open Images V6 into three groups according to the instance number in training split: *head*(more than 12k), *body*(0.2k $\sim$ 12k), *tail*(less than 0.2k). For performance comparison with the SOTA method, our method achieves significant improvement on tail categories and achieves the comparable overall performance with the GPS-Net [4] and Causal [5].

3.2. Per-class Performance Comparison with the Other Models

Following the previous works setting [1, 6, 4, 5], we show the comparison of Recall@100 on PredCls sub-task of each categories with the two SOTA methods [4, 5], as shown in fig 4.

Instead of only comparing the top-35 frequency categories, we present all 50 categories of Visual Genome. Our model achieves a significant performance gain on low-frequency categories, which demonstrates the effectiveness of our BGNN.

3.3. Visualization of Model Prediction

To better understand the BGNN, we visualize scene graph generation prediction from the Visual Genome dataset. As shown in Fig. 1, our model has a significant improvement for *body* and *tail* categories group compared with GPS-Net. With a more effective confidence-aware message propagation mechanism, our model has better context modeling capability of visual representations for low-frequency categories.

References

[1] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6163–6171, 2019. 2

[2] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5356–5364, 2019. 3

[3] 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: Con-

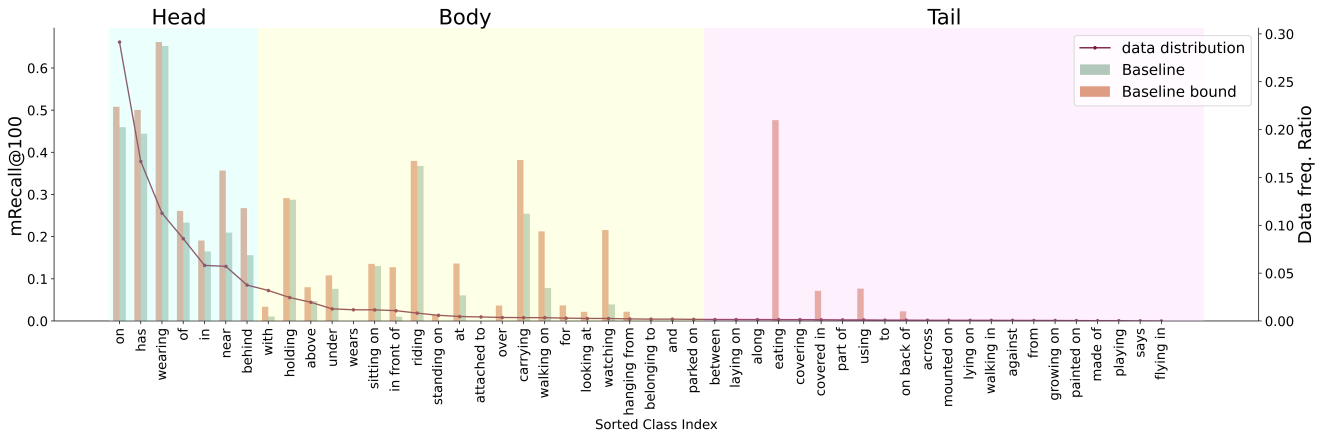

Figure 2: **The long-tail categories groups partition and the upper-bound comparison on Visual Genome dataset.**

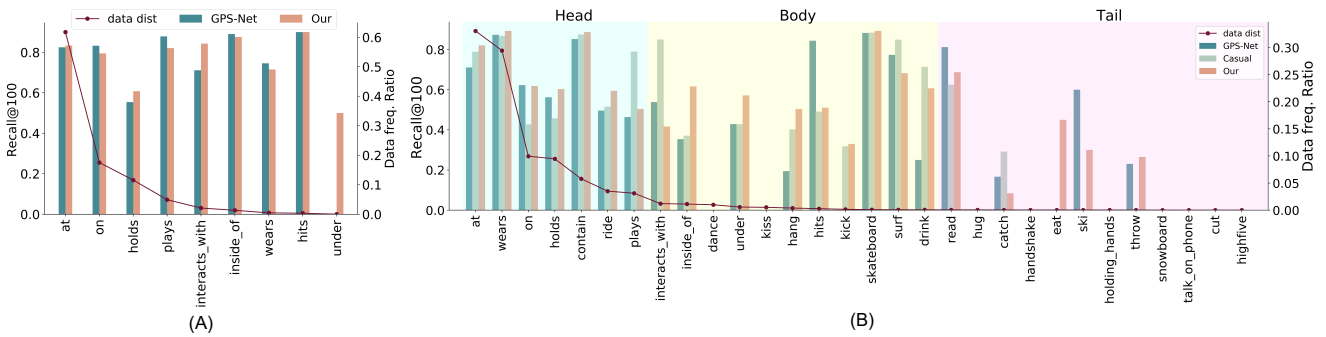

Figure 3: **The long-tail categories groups partition and per-class performance comparison of Open Images dataset**. Part (A) is the Open Images V4, part (A) is the Open Images V6 dataset. We compare with the two SOTA methods: Causal [5], and GPS-Net [4].

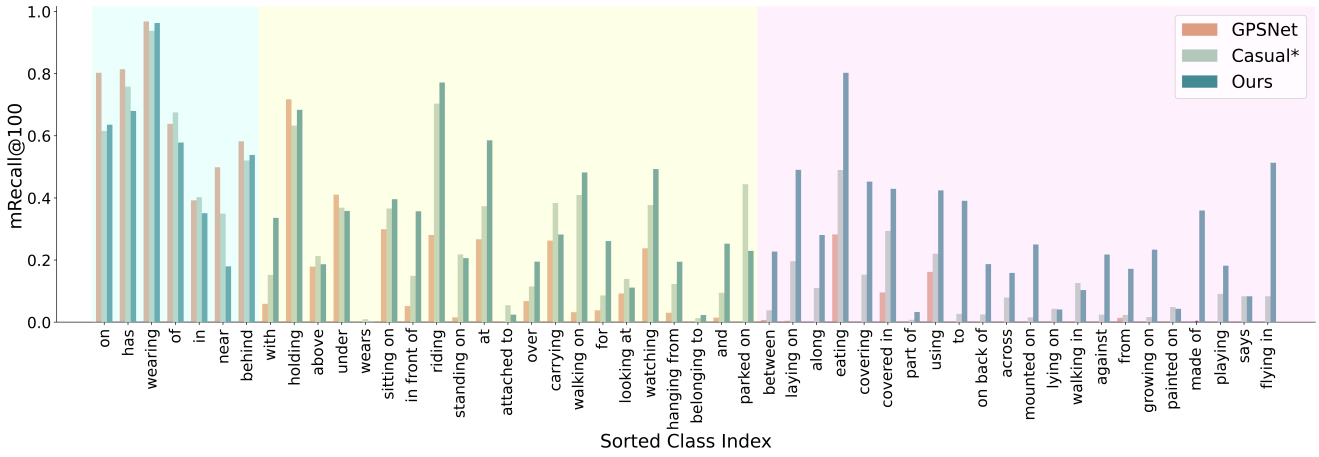

Figure 4: **The Recall@100 on Predicate Classification(PredCls) of all categories.** We compare with the SOTA methods: Causal [5], and GPS-Net [4]. $*$ denotes the re-sampling [2] is applied for this model.

necting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017. 2

[4] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3746–3753, 2020. 2, 3

[5] 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. 2, 3

[6] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6619–6628, 2019. 2

[7] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 5410–5419, 2017. 2