DBiased-P: Dual-Biased Predicate Predictor for Unbiased Scene Graph Generation

Xianjing Han, Xuemeng Song, *Senior Member, IEEE*, Xingning Dong, Yinwei Wei, *Member, IEEE*, Meng Liu, *Member, IEEE*, Liqiang Nie, *Senior Member, IEEE*

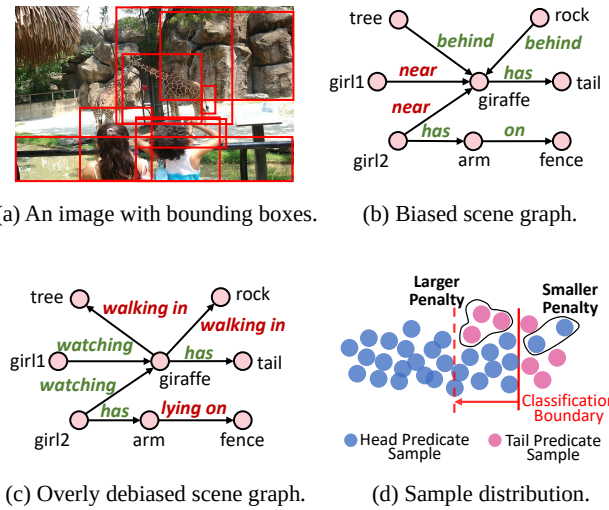
***Abstract*—Scene Graph Generation (SGG) is to abstract the objects and their semantic relationships within a given image. Current SGG performance is mainly limited by the biased predicate prediction caused by the long-tailed data distribution. Though many unbiased SGG methods have emerged to enhance the prediction of the tail predicates, their improvements on the tail predicates are often accompanied by the deterioration on the head ones, leading the prediction overly debiased. Toward this end, in this work, we propose a Dual-Biased Predicate Predictor (DBiased-P) to boost the unbiased SGG, which comprises a re-weighted primary classifier and an unweighted auxiliary classifier. The former classifier is tail-biased and used for the final predicate prediction, while the latter one is head-biased and designed to boost the head predicate prediction of the primary classifier by a head-oriented soft regularization. Experiments conducted on Visual Genome and Open Image datasets indicate the superiority of our DBiased-P in unbiased SGG, which significantly improves the recall@50 of the state-of-the-art unbiased SGG method DT2-ACBS from 23.3% to 55.5% as well as the mean recall@50 from 35.9% to 37.7%.**

***Index Terms*—Scene Graph Generation, Vision and Language, Re-Weighting Classification, KL-Divergence.**

I. Introduction

Scene Graph Generation (SGG) aims to detect objects and predict their pairwise relationships (*i.e.*, predicates) in an image. Essentially, the scene graph can be abstracted as a set of $\langle object_1, predicate, object_2 \rangle$ triplets, reflecting the structured representation of the image. Due to its various applications, such as image captioning [1]–[3] and visual question answering [4], [5], SGG has attracted tremendous research efforts. Early studies mainly focus on using the language priors [6]–[8] or modeling the object contextual information with advanced neural networks [9]–[12] to boost the object and predicate prediction. Despite achieving promising performance, they suffer from the biased predicate prediction caused by the long-tailed data distribution, which is a staple of the real world. In specific, in most SGG datasets (*e.g.*, Visual Genome [6]), only a few head predicate classes have abundant training samples, while plenty of tail predicate classes possess limited training data. Therefore, the model tends to predict the head predicate classes (*e.g.*, *on*, *has*, and *near*) rather than the tail ones (*e.g.*, *lying on*, *using*, and *watching*). As shown in Fig. 1 (b), the scene graph generated from the biased SGG model Motifs [10] is trivial and contains very limited informative predicates.

To tackle this downside, some studies [13]–[17] tend to enhance the unbiased SGG with various debiasing methods, such as re-sampling [16], [17] and re-weighting [14], [15]. To be specific, the re-sampling strategy [16] aims to enhance the tail predicate prediction by either over-sampling the tail predicate samples or under-sampling the head ones. Nevertheless, the



(a) An image with bounding boxes. (b) Biased scene graph. (c) Overly debiased scene graph. (d) Sample distribution.

Fig. 1. (a) An image with bounding boxes. (b) Biased scene graph generated by the biased model Motifs [10]. (c) Overly debiased scene graph generated by Motifs equipped with the re-weighted loss. The predicates in red and green are wrongly and correctly predicted, respectively. (d) Illustration of the re-weighted sample distribution by the re-weighted classifier, where the head predicate samples are less penalized than the tail ones when they are misclassified. This propels the classification boundary to move from the solid line to the dashed line, resulting in massive head predicate samples predicted as the tail predicates.

This work was supported by the National Key Research and Development Project of New Generation Artificial Intelligence under Grant 2018AAA0102502, in part by the National Natural Science Foundation of China under Grant 61772310, Grant 61702300, Grant 62006142, and Grant U1936203, in part by the Natural Science Foundation of Shandong Province under Grant ZR2019JQ23, in part by the Shandong Provincial Key Research and Development Program under Grant 2019JZZY010118, in part by the Shandong Provincial Natural Science Foundation for Distinguished Young Scholars under Grant ZR2021JQ26, in part by the Major Basic Research Project of Natural Science Foundation of Shandong Province under Grant ZR2021ZD15, in part by the Science and Technology Innovation Program for Distinguished Young Scholars of Shandong Province Higher Education In-stitutions under Grant 2021KJ036, and in part by the Innovation Teams in Colleges and Universities in Jinan under Grant 2018GXRC014. (Corresponding author: Liqiang Nie and Xuemeng Song.)

Xianjing Han, Xuemeng Song, Xingning Dong, and Liqiang Nie are with the School of Computer Science and Technology, Shandong University, Qingdao 266237, China (e-mail: hanxianjing2018@gmail.com, sxmustc@gmail.com, pass1463365882@gmail.com, nieliqiang@gmail.com).

Yinwei Wei is with the School of Computing, National University of Singapore, Singapore (e-mail: weiyinwei@hotmail.com).

Meng Liu is with the School of Computer Science and Technology, Shandong Jianzhu University, Jinan 250101, China (e-mail: mengliu.sdu@gmail.com).

over-sampling increases the computation, and under-sampling may lose the valuable samples for feature learning. Differently, the re-weighting strategy works on revising the objective function (*e.g.*, cross-entropy loss) of the predicate prediction by assigning the larger weights to the tail predicate classes as compared with the head ones. Due to its simplicity and efficiency, re-weighting has become a widely used debiasing method. However, despite its remarkable performance in tail predicate prediction, existing re-weighting based methods [14], [15] mainly suffer from the poor performance on the head predicate prediction and thus yield the overly debiased scene graph. As illustrated in Fig. 1 (c), the generated scene graph by Motifs equipped with the re-weighted loss involves several unreasonable predictions, *e.g.*, $\langle arm,\ lying\ on,\ fence \rangle$. This may be due to the less penalty towards the misclassification of the head predicates as compared to the tail ones, as shown in Fig. 1 (d). Therefore, we argue that one key to improve the performance of re-weighing methods is to rescue the misclassified head predicates, while maintaining the tail predicate prediction performance.

In fact, although the unweighted predicate classifier widely used in the biased SGG models usually fails in the tail predicate prediction, it always fits well on the head predicates, owing to their corresponding abundant training samples. Intuitively, we can take advantage of the unweighted predicate classifier (*i.e.*, its superior prediction for head predicates) to compensate for the poor head predicate prediction of SGG models that adopt re-weighted classifiers, and hence alleviate their overly debiased problems.

Towards this end, we propose a novel dual-biased predicate predictor, termed DBiased-P, consisting of a re-weighted primary classifier and an unweighted auxiliary classifier. The former is trained by a re-weighted objective function and biased to the tail predicates, while the latter is optimized by the unweighted objective function and biased to the head predicates. Notably, the re-weighted primary classifier is designed to output the final predicate prediction, while the unweighted classifier is introduced to regularize the former, especially its head predicate prediction. In particular, we propose a head-oriented soft regularization with the KL-divergence between the masked prediction distribution of the re-weighted classifier and that of the unweighted classifier. Based on this regularization, a large number of misclassified head predicate samples can be pulled back to the correct head predicate classes, alleviating the overly debiased prediction of the re-weighted classifier.

The main contributions are summarized as follows:

- We propose an effective DBiased-P for unbiased SGG, which guarantees the tail predicate prediction with the re-weighted classifier and promotes the head predicate prediction via a head-oriented soft regularization from the unweighted classifier. To the best of our knowledge, we are the first to sew the re-weighted and unweighted classifiers to boost the unbiased SGG.
- The proposed DBiased-P is model-free, which can be flexibly applied to the last layer of the predicate classification network of existing SGG models to enhance their unbiased predicate prediction.
- We conducted extensive experiments on Visual Genome [6] and Open Image [18] datasets, and the results indicate that our DBiased-P could achieve a better trade-off between head and tail predicate predictions. We release the source codes and model parameters on GitHub[1].

II. Related Work

A. Scene Graph Generation.

SGG is able to provide a semantic abstract of the image, which has received increasing attention in the computer vision community. Early works [19]–[22] mainly focus on improving the object representation to boost the predicate prediction. Specifically, they resort to the message passing method [6], recurrent sequential structured networks [10], [11], graph neural networks [23]–[25], or attention mechanism [26]–[28] to model the contextual information among objects. For example, Zellers *et al*. [10] emphasized the importance of contextual information among objects by leveraging the statistic to the object and predicate co-occurrence frequency, and introduced the global context based framework to enhance the predicate prediction. Though obvious improvements have been achieved by these efforts, they suffer from the biased prediction due to the long-tailed training data distribution.

B. Unbiased Scene Graph Generation.

Noticing the severe long-tail data distribution in the commonly used SGG dataset [6], Tang *et al*. [11] and Chen *et al*. [29] started to focus on the unbiased SGG and introduced the mean recall of each predicate to evaluate the unbiased SGG. Thereby, various debiasing SGG methods emerged, including causal inference [13], re-sampling [16], [17], and re-weighting [15], [30] based methods. For example, Li *et al*. [16] designed a bi-level data sampling method to adjust the unbalanced training data. Though better performance has been achieved, it suffers from the high calculation cost caused by the image-level over-sampling. Tang *et al*. [13] employed the counterfactual causality to disentangle the bias from the representation. To enhance the prediction of the tail predicate classes, Yan *et al*. [15] and Yu *et al*. [14] adopted the re-weighting based methods to increase the loss penalty of tail predicate classes. Though these debiasing methods improve the performance on the recall of tail predicates, they lose much performance on the head predicate classes. Therefore, in this work, we aim to compensate for the performance of head predicate classes to achieve a better trade-off prediction between head and tail predicate classes.

C. Re-weighting.

Re-weighting is a branch of cost sensitive learning [31], which is employed to balance the biased prediction based on the unbalanced dataset by adjusting the loss cost for different classes. As a classic re-weight method, weighting by inverse class frequency [32] has been commonly adopted in many

[1] https://github.com/hanxjing/Dbiased-P.

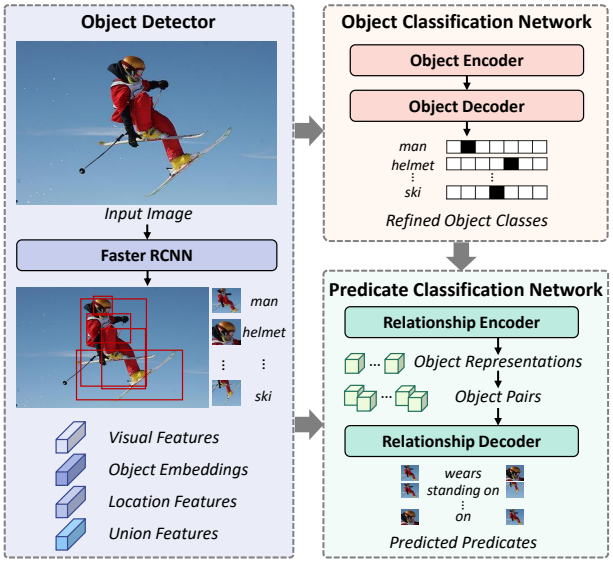


Fig. 2. The pipeline of current SGG models consists of three key components, based on which we construct our overall framework.

fields, such as computer vision [33] and natural language processing [34]. Its smoothed version of the inverse square root of class frequency [32] is also widely adopted in many tasks [35], [36]. Due to the concern that as the number of samples increases, the additional benefit of a newly added data point will diminish, Cui *et al*. [37] introduced the concept of the effective number of samples, based on which adjust the class frequency to assign the loss weight for different classes.

Although the re-weighting strategy has been employed in SGG [13]–[15], the performance is far from satisfactory due to the serious sacrifice on the head predicate prediction when pursuing the improvement on tail predicates. Towards this end, in this work, we aim to improve the head predicate prediction of re-weighting methods in SGG.

III. Methodology

In this section, we first give the overview of the common SGG pipeline we adopted, and then present our DBiased-P, which is deployed on the last layer of the predicate classification network in the SGG pipeline.

A. Pipeline of SGG Models

In this work, we adopt the mainstream pipeline used by existing methods [10], [15], [25], [27], as shown in Fig. 2, which consists of three components: 1) object detector, 2) object classification network, and 3) predicate classification network. For a given image I, we can get a set of object proposal bounding boxes $B = \{b_i\}_{i=1}^{N}$ by the object detector, the object predictions $O = \{o_i | o_i \in \mathcal{O}\}_{i=1}^{N}$ from the object classification network, and the relationship $R = \{r_{ij} | r_{ij} \in \mathcal{R}\}$ of different object pairs through the predicate classification network. Formally, we use $\mathcal{O}$ and $\mathcal{R}$ to represent the set of objects and predicate classes, respectively. We then generate the scene graph $G = \{(o_i, r_{ij}, o_j)\}$ of the image I according to the following probability model:

$$Pr(G|I) = Pr(B|I)Pr(O|B,I)Pr(R|O,B,I), \quad (1)$$

where $P_r(B|I)$, $P_r(O|B,I)$, and $P_r(R|O,B,I)$ denote the object detector, object classification network, and predicate classification network, respectively.

Object Detector. The pre-trained Faster R-CNN [38] is commonly adopted as the object detector in existing SGG studies [10], [14], where each detected object o_i can be represented with a visual feature $\mathbf{v}_i$, an object embedding of the initial detected object class $\mathbf{e}_i$, and a location feature $\mathbf{b}_i$ (*i.e*., the coordinates of the object bounding box).

Object Classification Network. Existing SGG studies [10], [11] usually adopt the encoder-decoder based object classification network. The object encoder $Encoder_o$ targets at encoding the rich object contextual information into the object representations, while the object decoder $Decoder_o$ works on predicting the refined object class, which can be formulated as follows,

$$\begin{cases} \hat{\mathbf{X}} = Encoder_o([\mathbf{v}_i; \mathbf{e}_i; \mathbf{b}_i]_{i=1,2,\cdots,N}), \\ \hat{\mathbf{O}} = Decoder_o([\hat{\mathbf{x}}_i]_{i=1,2,\cdots,N}), \end{cases} \quad (2)$$

where $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \cdots, \hat{\mathbf{x}}_N]$ and $\hat{\mathbf{O}} = [\hat{\mathbf{o}}_1, \hat{\mathbf{o}}_2, \cdots, \hat{\mathbf{o}}_N]$ denote the encoded object representations and refined object label vectors of all the objects in the given image, respectively. In existing SGG studies [10], [14], BiLSTMs [39] and multi-head self-attention [40] are widely employed in $Encoder_o$ and $Decoder_o$. We denote the object embedding of the refined object class as $\hat{\mathbf{e}}_i$, which is employed in the following predicate classification.

Predicate Classification Network. Similar to the object classification network, the predicate classification network also includes a relationship encoder and a relationship decoder, where the relationship encoder $Encoder_r$ focuses on encoding the object contextual information with the refined object label, and the relationship decoder $Decoder_r$ works on classifying the predicate for object pairs with their union feature $\mathbf{u}_{ij}$. The predicate logit $\mathbf{r}_{ij}$ of object pair (o_i, o_j) is calculated as follows,

$$\begin{cases} \mathbf{X} = Encoder_r([\hat{\mathbf{x}}_i; \hat{\mathbf{e}}_i]_{i=1,2,\cdots,N}), \\ \mathbf{r}_{ij} = Decoder_r(\mathbf{x}_i, \mathbf{x}_j, \mathbf{u}_{ij}), \end{cases} \quad (3)$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$ denotes the encoded object representations of all the objects in the given image. In existing SGG studies [10], [14], the $Encoder_r$ usually has similar network structure to $Encoder_o$, and $Decoder_r$ usually adopts the fully-connected layers.

B. Dual-Biased Predicate Predictor

Affected by the unbalanced dataset in SGG task, the general unweighted classifier is more likely to be biased to the head predicate prediction. To promote the tail predicate prediction, the re-weighted classifier adjusts the loss function by assigning the larger weights to the tail predicate classes as compared with the head ones. The re-weighted classifier increases the misclassified cost of tail predicate classes and thus is more likely to be biased to the tail predicate prediction. Intuitively, they complement each other, and can be sewed together to promote the unbiased SGG. In a sense, we can use one type

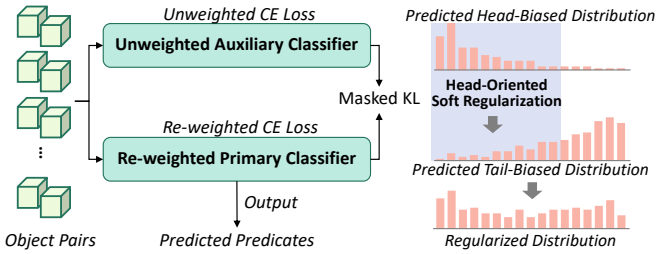


Fig. 3. The framework of our dual-biased predicate predictor, which consists of a re-weighted primary classifier and an unweighted auxiliary classifier. A head-oriented soft regularization is conducted on the distributions of the two classifier by the masked KL-divergence.

of classifier as the primary classifier for the final predicate classification, and the other type of classifier as the auxiliary classifier, whose output can be used to regularize the output of the primary one. Compared with the quantity of the misclassified tail predicate samples of the unweighted classifier, the number of the misclassified head predicate samples of the re-weighted classifier is more prominent. Therefore, it is more promising to use the head predicate predictions of the unweighted classifier to regularize the re-weighting one, rather than the inverse. Accordingly, as shown in Fig. 3, we devise the DBiased-P with the re-weighted classifier as the primary classifier, and the unweighted classifier as the auxiliary one to provide head-oriented regularization.

Re-weighted Primary Classifier. We build the primary classifier with a fully-connected layer. In a sense, we can use different existing re-weighting methods to design the objective function of the primary classifier. Here, we take the frequently adopted one [32] that re-weights samples according to the inverse class frequency as an example. Formally, the re-weighted classifier can be formulated as follows,

$$\begin{cases} \mathbf{r}_{rw} = f_{rw}(\mathbf{x}_i, \mathbf{x}_j, u_{ij}), \\ s^p_{rw} = \dfrac{\exp\left(r^p_{rw}\right)}{\sum_{i=1}^{P} \exp\left(r^i_{rw}\right)}, \\ L_{rw}(\mathbf{r}_{rw}) = -\sum_{p=1}^{P} w^p y^p \log(s^p_{rw}), \end{cases} \tag{4}$$

where f_{rw} denotes the fully connected layer for a re-weighted classifier, and $\mathbf{r}_{rw} = [r^1_{rw}, r^2_{rw}, \cdots, r^P_{rw}] \in \mathbb{R}^P$ is the predicted predicate logits of object pair (o_i, o_j)[2]. s^p_{rw} is the predicted probability of the p-th predicate class by the re-weighted classifier, while $y^p \in \{0, 1\}$ is the corresponding ground truth label. $P = |\mathcal{R}|$ is the number of the predicate classes. L_{rw} is the re-weighted cross-entropy loss and w^p is the loss weight of the p-th predicate class, which is defined as follows,

$$w^p = P \frac{q_p^{-1}}{\sum_{i=1}^{P} q_i^{-1}}, \tag{5}$$

where q_p denotes the class frequency of the p-th predicate.

Head-oriented Regularization. We resort to the unweighted auxiliary classifier to provide the head-oriented regularization for the prediction of the re-weighted primary classifier. Similarly, we use a fully-connected layer to construct the unweighted auxiliary classifier. Differently, we use the unweighted cross-entropy loss L_{uw} as the objective function of the unweighted classifier, which can be formulated as follows,

$$\begin{cases} \mathbf{r}_{uw} = f_{uw}(\mathbf{x}_i, \mathbf{x}_j, u_{ij}), \\ s^p_{uw} = \dfrac{\exp\left(r^p_{uw}\right)}{\sum_{i=1}^{P} \exp\left(r^i_{uw}\right)}, \\ L_{uw}(\mathbf{r}_{uw}) = -\sum_{p=1}^{P} y^p \log(s^p_{uw}), \end{cases} \tag{6}$$

where f_{uw} denotes the fully connected layer for the auxiliary unweighted classifier. $\mathbf{r}_{uw}$ denotes the predicted predicate logits of the unweighted classifier, and s^p_{uw} is the predicted probability of the p-th predicate class by the unweighted classifier.

Since the unweighted classifier fits well to the head predicate classes, we take advantage of its prediction on head predicates to regularize the re-weighted classifier. Specifically, we resort to the KL-divergence to force the head predicate prediction of the re-weighted classifier close to that of the unweighted classifier. Towards this end, we first modify the logits generated by these two classifiers by adding a mask to their tail predicate predictions as follows,

$$\begin{cases} \tilde{\mathbf{r}}_{rw} = \mathbf{r}_{rw} + \mathbf{m}, \\ \tilde{\mathbf{r}}_{uw} = \mathbf{r}_{uw} + \mathbf{m}, \end{cases} \tag{7}$$

where $\mathbf{m} = [m^1, m^2, \cdots, m^P] \in \mathbb{R}^P$ is the mask to set tail predicate logits to $-\infty$, whose p-th entry is defined as:

$$m^p = \begin{cases} -\infty, & \text{if } p \in \mathcal{P}_{tail}, \\ 0, & \text{otherwise}, \end{cases} \tag{8}$$

where $\mathcal{P}_{tail}$ is the set of tail predicate classes, and we take the same tail predicate definition as [16]. By adding the mask, the tail predicate probabilities derived by softmax over the modified logits $\tilde{\mathbf{r}}_{rw}$ and $\tilde{\mathbf{r}}_{uw}$ will be the same (*i.e.*, zero). Namely, we have:

$$\begin{cases} \tilde{\mathbf{s}}_{rw} = \text{softmax}(\tilde{\mathbf{r}}_{rw}), \\ \tilde{\mathbf{s}}_{uw} = \text{softmax}(\tilde{\mathbf{r}}_{uw}), \\ L_{kl}(\tilde{\mathbf{r}}_{rw}, \tilde{\mathbf{r}}_{uw}) = \sum_{p=1}^{P} \tilde{s}^p_{uw} \log \dfrac{\tilde{s}^p_{uw}}{\tilde{s}^p_{rw}}, \end{cases} \tag{9}$$

where $\tilde{\mathbf{s}}_{rw}$ and $\tilde{\mathbf{s}}_{uw}$ are the modified predicate probabilities. The tail predicate probabilities of both $\tilde{\mathbf{s}}_{rw}$ and $\tilde{\mathbf{s}}_{uw}$ are zero. Therefore, we can directly adopt the KL-divergence loss L_{kl} over $\tilde{\mathbf{s}}_{rw}$ and $\tilde{\mathbf{s}}_{uw}$ as the head-oriented regularization, to force the head predicate probabilities of $\tilde{\mathbf{s}}_{rw}$ to be similar to that of $\tilde{\mathbf{s}}_{uw}$, and hence improve the head predicate prediction of the re-weighted classifier.

Ultimately, the objective function of the DBiased-P is written as follows,

$$L = L_{uw}(\mathbf{r}_{uw}) + L_{rw}(\mathbf{r}_{rw}) + \lambda L_{kl}(\tilde{\mathbf{r}}_{rw}, \tilde{\mathbf{r}}_{uw}), \tag{10}$$

where λ is the hyperparameter to control the head-oriented regularization degree.

[2]For the convenient presentation and understanding, we omit the subscript of $_{ij}$ in the following presentation.

Gradient Discussion. To intuitively understand the dual-biased mechanism of our DBiased-P, we discuss the gradient of the final objective function with respect to the logits $\mathbf{r}_{rw}$, which is used for final predicate classification.

Given an object pair sample of the p-th predicate, we can obtain the gradient of L_{rw} on r^p_{rw} as follows,

$$\frac{\partial L_{rw}}{\partial r^p_{rw}} = w^p(s^p_{rw} - 1). \tag{11}$$

Then if the given sample is misclassified, *i.e.*, s_p close to 0, it would mainly penalize the classifier according to the predicate weight w_p. Since the samples of tail predicates are assigned with the large weights, *i.e.*, w_p in Eqn. (5), their misclassification would penalize the classifier more heavily, as compared to the head ones. Accordingly, the prediction of the original re-weighted classifier without head-oriented regularization tilts to the tail predicate classes, leading a large amount of head predicate samples misclassified into the tail predicate classes.

Regarding the regularization L_{kl}, we give its gradient on r^p_{rw} as follows,

$$\frac{\partial L_{kl}}{\partial r^p_{rw}} = \begin{cases} 0, & \text{if } p \in \mathcal{P}_{tail}, \\ \tilde{s}^p_{rw} - \tilde{s}^p_{uw}, & \text{otherwise.} \end{cases} \tag{12}$$

The gradient shows that the KL-divergence promotes the classifier to mitigate the difference between the head predicate probabilities of the re-weighted and unweighted classifiers. Specifically, given a sample, if the unweighted classifier predicts it as the p-th head predicate class with the higher probability, *i.e.*, $\tilde{s}^p_{uw}$ is larger, then the gradient would largely encourage the re-weighted classifier to increase the corresponding predicted probability, *i.e.*, $\tilde{s}^p_{rw}$, vice versa. As the unweighted classifier is biased to the head predicate classes, it usually predicts all the samples with higher probabilities on head predicate classes. Moreover, according to our observations, in most case of the unweighted classifier, the head predicate probabilities of the head predicate samples is usually higher and more distinct than that of the tail predicate samples. Accordingly, as shown in Fig. 4, the KL-divergence based regularization on head predicate samples should be larger than that on tail predicate samples. Thereby, our head-oriented soft regularization is able to adaptively improve the head predicate prediction and keep the tail predicate prediction.

In a word, our DBiased-P adaptively increases the regularization towards the head predicates to balance their smaller loss weights in the original re-weighted scenario.

IV. Experiments

In this section, we conducted experiments to demonstrate the effectiveness of our proposed DBiased-P.

A. Experiment Settings

Datasets. We present experimental results on two datasets: Visual Genome (VG) [41] and Open Image [18].

VG is the most widely used benchmark for scene graph generation. We use the pre-processed version of the dataset from [6], which consists of 108K images, 150 object categories, and 50 predicate categories. We adopt the same experimental settings with [10], [13], [14], *i.e.*, 70% of the images for training, 30% of that for testing, as well as sample 5K images from training set for validation.

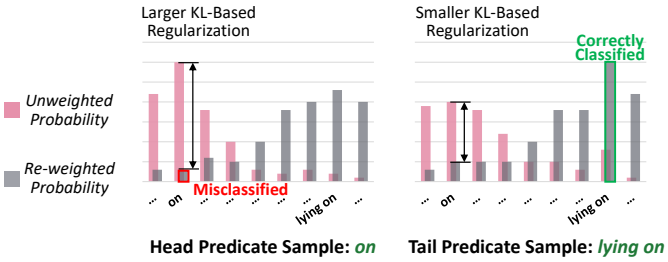


Fig. 4. An intuitive example of the KL-divergence based regularization. For the head predicate sample *on*, the higher predicted probability of unweighted classifier leads to a larger KL-divergence based regularization, which can heavily regularize the incorrect prediction in the re-weighted classifier. For the tail predicate sample *lying on*, the KL-divergence based regularization is relatively smaller, which may not affect the correct prediction in the re-weighted classifier.

Open Image is a large-scale dataset proposed by Google, which provides the superior annotation for the scene graph generation. Specifically, we adopt the Open Image V6, which includes 134K images, 300 object categories, and 30 predicate categories. We follow the same data split with [16], [27], and obtain 126K images for training, 2K images for validation, and 5K images for testing.

Similar to BGNN [16], in VG [41] dataset, we take the predicate classes that contain less than 0.5K samples as the tail classes. Thereby, VG dataset includes 28 head and 22 tail predicate classes. While in Open Image [18] dataset, we take the predicate classes that contain less than 0.2K samples as the tail classes. Thereby, Open Image dataset includes 18 head and 12 tail predicate classes.

Evaluation Tasks. Following the previous works [6], [27], [29], [44], we mainly adopt the following three tasks to evaluate our model: 1) Predicate Classification (**PredCls**) takes the ground truth object labels and bounding boxes to predict the predicate classes; 2) Scene graph classification (**SGCls**) takes the ground truth bounding boxes to predict the object and predicate classes; and 3) Scene graph detection (**SGDet**) predicts the scene graphs from images.

Evaluation Metrics. For VG dataset, following the existing works [11], [16], [29], we adopt three types of metrics: 1) Recall@K (**R@K**) is the recall of all samples, which is usually dominated by head classes in the long-tailed dataset; 2) Mean recall@K (**mR@K**) is the average R@K of all predicate classes, which validates the unbiased scene graph generation and is mainly dominated by tail classes; and 3) the mean of these two types of recall, denoted as **Mean**, which reflects the balance among the correct and unbiased generation. We adopt $K \in \{50, 100\}$ in our experiments. Mean=(R@50+R@100+mR@50+mR@100)/4. As for Open Image dataset, we adopt the same evaluation metrics in [12], [16], including mR@50, R@50, weighted mean AP of relationships (wmAP_{rel}), weighted mean

TABLE I
PERFORMANCE COMPARISON IN PREDCLS, SGCLS, AND SGDET TASKS WITH VG DATASET IN TERMS OF MR@50/100, R@50/100, AND THEIR MEAN. † DENOTES THAT THE METHOD EMPLOYS FASTER R-CNN WITH VGG-16. ∗ MEANS RE-SAMPLING IS APPLIED IN THIS MODEL. u INDICATES THAT THE METHOD AIMS AT UNBIASED SGG. THE BEST RESULTS OF METHODS WITH THE SAME ENCODING METHOD ARE HIGHLIGHTED IN BOLDFACE.

Model	PredCls			SGCls			SGDet		
	mR@50/100	R@50/100	Mean	mR@50/100	R@50/100	Mean	mR@50/100	R@50/100	Mean
IMP† [6]	9.8 / 10.5	59.3 / 61.3	35.2	5.8 / 6.0	34.6 / 35.4	20.5	3.8 / 4.8	20.7 / 24.5	13.5
Motifs† [10]	14.0 / 15.3	65.2 / 67.1	40.4	7.7 / 8.2	35.8 / 36.5	22.1	5.7 / 6.6	27.2 / 30.3	17.5
KERN† [29]	17.7 / 19.2	65.8 / 67.6	42.6	9.4 / 10.0	36.7 / 37.4	23.4	6.4 / 7.3	27.1 / 29.8	17.7
VCTree$^{\dagger}_{u}$ [11]	17.9 / 19.4	66.4 / 68.1	43.0	10.1 / 10.8	38.1 / 38.8	24.5	6.9 / 8.0	27.9 / 31.3	18.5
GPS-Net$^{\dagger}_{u}$ [27]	21.3 / 22.8	66.9 / 68.8	45.0	11.8 / 12.6	39.2 / 40.1	25.9	8.7 / 9.8	28.4 / 31.7	19.7
Schemata$^{\dagger}_{u}$ [42]	19.1 / 20.7	66.9 / 68.4	43.8	10.1 / 10.9	39.1 / 39.8	25.0	-	-	-
PCPL$^{\dagger}_{u}$ [15]	35.2 / 37.8	50.8 / 52.6	44.1	18.6 / 19.6	27.6 / 28.4	23.6	9.5 / 11.7	14.6 / 18.6	13.6
BGNN$^{*}_{u}$ [16]	30.4 / 32.9	59.2 / 61.3	46.0	14.3 / 16.5	37.4 / 38.5	26.7	10.7 / 12.6	31.0 / 35.8	22.5
DT2-ACBS$^{*}_{u}$ [17]	35.9 / 39.7	23.3 / 25.6	31.1	24.8 / 27.5	16.2 / 17.6	21.5	22.0 / 24.4	15.0 / 16.3	19.4
Motifs [10]	4.6 / 15.8	**66.1 / 68.0**	41.1	8.0 / 8.5	**39.3 / 40.1**	24.0	5.8 / 7.8	**32.5 / 37.3**	20.7
Motifs-EBM$_u$ [43]	18.0 / 19.5	65.2 / 67.3	42.5	10.2 / 11.0	39.2 / 40.0	25.1	7.7 / 9.3	31.7 / 36.3	21.3
Motifs-TDE$_u$ [13]	25.5 / 29.1	46.2 / 51.4	38.1	13.1 / 14.9	27.7 / 29.9	21.4	8.2 / 9.8	16.9 / 20.3	13.8
Motifs-CogTree$_u$ [14]	26.4 / 29.0	35.6 / 36.8	32.0	14.9 / 16.1	21.6 / 22.2	18.7	10.4 / 11.8	20.0 / 22.1	16.1
Motifs-DBiased$_u$ (ours)	**34.7 / 36.6**	58.8 / 60.7	**47.7**	**20.3 / 21.2**	36.5 / 37.4	**28.9**	**14.9 / 17.5**	29.4 / 33.9	**24.0**
VCTree [11]	14.9 / 16.1	**66.2 / 68.1**	41.3	7.5 / 7.9	40.5 / 41.4	24.3	5.7 / 6.9	**31.5 / 36.2**	20.1
VCTree-EBM$_u$ [43]	18.2 / 19.7	64.0 / 65.8	41.9	12.5 / 13.5	**44.7 / 45.8**	**29.1**	7.7 / 9.1	31.4 / 35.9	21.0
VCTree-TDE$_u$ [13]	25.4 / 28.7	47.2 / 51.6	38.2	12.2 / 14.0	25.4 / 27.9	19.9	9.3 / 11.1	19.4 / 23.2	15.8
VCTree-CogTree$_u$ [14]	27.6 / 29.7	44.0 / 45.4	36.7	18.8 / 19.9	30.9 / 31.7	25.3	10.4 / 12.1	18.2 / 20.4	15.3
VCTree-DBiased$_u$ (ours)	**34.5 / 36.4**	59.1 / 61.0	**47.8**	**20.4 / 21.3**	36.8 / 37.7	**29.1**	**14.3 / 17.0**	29.5 / 34.1	**23.7**
SG [14]	18.5 / 20.2	**65.0 / 66.9**	42.7	11.5 / 12.3	**39.1 / 39.9**	25.7	7.7 / 9.0	**30.3 / 33.3**	20.1
SG-CogTree$_u$ [14]	28.4 / 31.0	38.4 / 39.7	34.4	15.7 / 16.7	22.9 / 23.4	19.7	11.1 / 12.7	19.5 / 21.7	16.3
SG-DBiased$_u$ (ours)	**37.7 / 40.2**	55.5 / 57.4	**47.7**	**22.0 / 22.9**	34.1 / 34.9	**28.5**	**16.4 / 19.7**	27.0 / 31.4	**23.6**

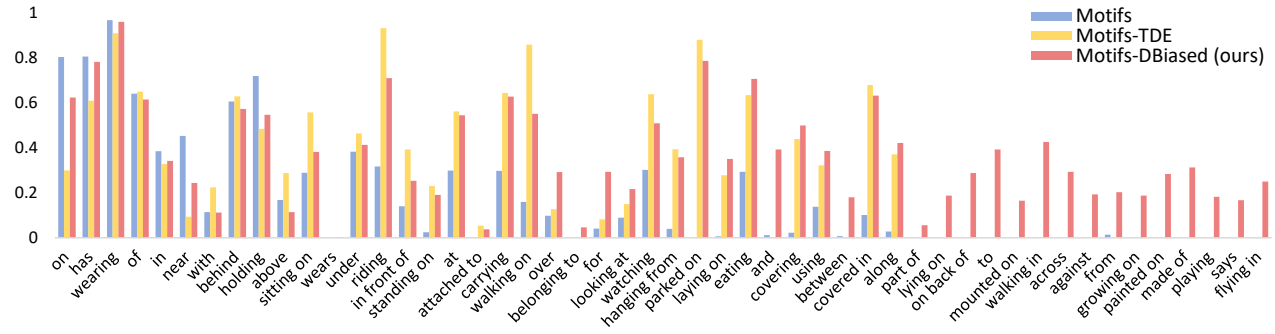


Fig. 5. R@100 of each predicate in Motifs, Motifs-TDE, and Motifs-DBiased in PredCls task on VG dataset.

TABLE II
PERFORMANCE COMPARISON IN SGDET TASK WITH OPEN IMAGE DATASET IN TERMS OF MR@50, R@50, WMAP$_{rel}$, WMAP$_{phr}$, AND SCORE$_{wtd}$. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLDFACE.

model	mR@50	R@50	wmAP		score$_{wtd}$
			rel	phr	
VCTree [11]	33.9	74.1	34.2	33.1	40.2
Motifs [10]	32.7	71.6	29.9	31.6	38.9
TDE [13]	35.5	69.3	30.7	32.8	39.3
BGNN [16]	40.5	**75.0**	33.5	34.2	42.1
Motifs-DBiased	**42.1**	74.6	**34.3**	**34.4**	**42.3**

AP of phrase (wmAP$_{phr}$), and the final weighted score (score$_{wtd}$), to evaluate the generated scene graph. According to the Open Image challenge [18], score$_{wtd}$ = 0.2×R@50+0.4×wmAP$_{rel}$+0.4×wmAP$_{phr}$.

Implementation Details. We use a pre-trained Faster R-CNN with ResNeXt-101-FPN provided by [13] and [16] as object detectors for VG and Open Image datasets, respectively. Object class embeddings are 200-D word embeddings generated by Glove [45]. Following the SGG method CogTree [14], we adopt the re-weighting method of inverse effective frequency (IEF) [37] in our DBiased-P, where the re-weighting factor [37] β is set to 0.99996. We employ SGD with a momentum of 0.9 as the optimizer. Batch size and initial learning rate are consistently set to 10 and 0.001 for all three tasks, respectively. We adopt the same warm-up and decayed strategy as [13], and each training procedure lasts for 50,000 steps. The hyperparameter λ is set to 0.5. All our experiments are conducted with a RTX2080 Ti GPU.

As our DBiased-P is model-free and is designed to deploy on the last layer of the predicate classification network of existing SGG models, the object encoder $Encoder_o$, object decoder $Decoder_o$, and relationship encoder $Encoder_r$ of our method depend on the adopted SGG backbone for the sake of fair comparison. Specifically, we deploy our DBiased-P on three different SGG models, including Motifs [10], VCTree [11], and SG [14], denoted as Motifs-DBiased, VCTree-DBiased, and SG-Debiased, respectively. As the BiLSTM [39], TreeLSTM [46], and Transformer [40] networks are respectively employed in Motifs, VCTree, and SG, we accordingly adopt the BiLSTM, TreeLSTM, and Transformer networks as the $Encoder_o$, $Decoder_o$, and $Encoder_r$ of Motifs-DBiased, VCTree-DBiased, and SG-Debiased, respectively. In addition,

TABLE III
PERFORMANCE COMPARISON WITH DIFFERENT RE-WEIGHTING METHODS ON VG DATASET. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLDFACE.

Model	PredCls			SGCls			SGDet		
	R@50/100	mR@50/100	Mean	R@50/100	mR@50/100	Mean	R@50/100	mR@50/100	Mean
Motifs	**66.1** / **68.0**	4.6 / 15.8	41.1	**39.3** / **40.1**	8.0 / 8.5	24.0	**32.5** / **37.3**	5.8 / 7.8	20.7
Motifs-IF	39.7 / 41.9	37.0 / 38.9	39.4	24.8 / 25.9	21.3 / 22.5	23.6	19.4 / 23.1	14.7 / 17.4	18.7
Motifs-DBiased(IF)	46.7 / 49.0	**37.3** / **39.2**	43.1	29.2 / 30.2	**21.8** / **22.8**	26.0	21.2 / 25.2	**15.5** / **18.0**	20.0
Motifs-ISF	58.4 / 60.4	30.2 / 32.2	45.3	36.2 / 37.1	18.6 / 19.6	27.9	28.1 / 32.9	14.3 / 17.0	23.1
Motifs-DBiased(ISF)	60.6 / 62.5	30.1 / 32.2	46.4	37.7 / 38.6	18.5 / 19.5	28.6	29.7 / 34.2	14.2 / 16.7	23.7
Motifs-IEF	56.9 / 58.8	34.7 / 36.5	46.7	35.0 / 35.9	19.9 / 21.1	28.0	28.1 / 32.6	14.9 / 17.6	23.3
Motifs-DBiased(IEF)	58.8 / 60.7	34.7 / 36.6	**47.7**	36.5 / 37.4	20.3 / 21.2	**28.9**	29.4 / 33.9	14.9 / 17.5	**24.0**

same with Motifs, VCTree, and SG, the general cross-entropy losses are employed in object classification networks to refine the object class.

B. Comparison to Existing Methods

We compare our DBiased-P with the existing SGG models to evaluate the ability of unbiased scene graph generation. Besides, we also compare with the mainstream re-weighting methods to demonstrate the effectiveness of our DBiased-P on them.

Comparison with SGG Models. For fair comparison, we compare our model with the model-free unbiased methods based on the same encoding method, including EBM [43], TDE [13], and CogTree [14]. We also compare our model with other biased and unbiased SGG methods, such as BGNN [16] and DT2-ACBS [17], which are state-of-the-art re-sampling based methods. Table I and Table II show the performance comparison of our methods and baseline methods on VG and Open Image datasets, respectively. In Table I, we report the results of baseline methods according to their papers. In Table II, we report the results of baseline methods according to [16].

To begin with, it is worth noting that the decrease on the overall recall (R@K) is hard to avoid when pursuing the increase on the mean recall of all predicates (mR@K) on a biased dataset. From Table 1, we have the following comparisons and observations: **1)** Compared with the three model-free debiasing methods (*i.e.*, TDE, CogTree, EBM) on all three encoding baseline methods (*i.e.*, Motifs, VCTree, and SG), our DBiased-P (*i.e.*, Motifs-DBiased, VCTree-DBiased, and SG-DBiased) achieves the best mR@50/100 and Mean. This reflects that our DBiased-P can largely improve mR@50/100, and keep satisfactory results on R@50/100 at the same time. **2)** Compared with all other baseline methods, our DBiased-P achieves the best Mean, which demonstrates the superiority of our DBiased-P in balancing the correct and unbiased SGG. And **3)** compared with DT2-ACBS, which achieves better performance in terms of mR@50/100 in SGCls and SGDet tasks, our DBiased-P has obvious advantages in terms of R@50/100 and Mean. Moreover, DT2-ACBS achieves the worst R@50/100 in all three evaluation tasks. The underlying reason is that the sampling strategies of DT2-ACBS sacrifice much overall recall to increase the mean recall. Consequently, though DT2-ACBS largely improves the tail predicate prediction, the prediction of the head predicate classes that occupy most samples in the dataset is decreased. In other word, the accuracy of the generated scene graph (*i.e.*, R@50/100) by DT2-ACBS is inferior to our DBiased-P. In addition, as DT2-ACBS employs different sampling strategies with two training stages in predicate prediction, our DBiased-P is more effective and simpler to implement.

Besides, from Table II, we observe that our DBiased-P achieves better performance than the baseline methods in terms of almost all metrics, indicating that our model can achieve better performance on various datasets. However, compared with BGNN, our DBiased-P achieves better mR@50 and worse R@50 on both VG and Open Image datasets. One possible explanation is that the over-sampling strategy used in BGNN are somehow weaker regarding the tail performance, but stronger in retaining the head performance, as compared to our DBiased-P. Overall, our DBiased-P performs better than BGNN in terms of the Mean metric. Notably, different from BGNN that has the multi-stage graph refinement and over-sampling of the image, DBiased-P is simple to implement and model-free.

To obtain a deep insight, we compare our DBiased-P with both biased Motifs and unbiased Motifs-TDE methods in each predicate class. From Fig. 5, we observe that: **1)** compared with the biased Motifs, our Motifs-DBiased is able to significantly improve the performance on most of the tail predicate classes with a slight decrease on that of the head ones; and **2)** compared with the unbiased Motifs-TDE, our Motifs-DBiased performs better on most of the predicate classes, including both the head predicate classes and some tail predicate classes. One possible reason is that although TDE can enhance the tail predicate prediction by disentangling the environmental bias, it may also remove some useful bias that benefits for the head predicate prediction, resulting the decrease of the head predicate prediction in Motifs-TDE. This reflects that our DBiased-P is more effective than removing the environmental bias in boosting the unbiased SGG. These two observations indicate that our DBiased-P is able to achieve a better trade-off between head and tail predicate prediction.

Comparison with Re-weighting Methods. To demonstrate the generalization capability of our DBiased-P on re-weighting methods, we adopt three commonly used re-weighting methods, including inverse class frequency (IF) [32], inverse square root of class frequency (ISF) [32], and inverse effective number (IEF) [37]. Then, based on Motifs, we have three variants, denoted as Motifs-DBiased(IF), Motifs-DBiased(ISF), and Motifs-DBiased(IEF), respectively. For comparison, we introduce three baseline methods: Motifs-IF, Motifs-ISF, and Motifs-IEF, which are directly deployed on the Motifs, respectively.

TABLE IV
ABLATION STUDY ON VG DATASET IN TERMS OF R@50/100, MR@50/100, AND MEAN IN SGDET, SGCLS, AND PREDCLS TASKS.

Model	PredCls			SGCls			SGDet		
	R@50/100	mR@50/100	Mean	R@50/100	mR@50/100	Mean	R@50/100	mR@50/100	Mean
Motifs	**66.1** / **68.0**	4.6 / 15.8	41.1	39.3 / 40.1	8.0 / 8.5	24.0	**32.5** / **37.3**	5.8 / 7.8	20.7
Motifs-DBiased-w/o-L_{kl}	57.2 / 59.0	34.6 / 36.6	46.9	35.0 / 36.1	20.0 / 21.2	28.1	28.5 / 32.8	14.9 / 17.8	23.5
Motifs-HardR	50.2 / 52.1	**38.7** / **40.8**	45.4	35.0 / 35.9	**21.3** / **22.3**	28.6	28.0 / 32.2	**15.6** / **18.6**	23.6
Motifs-TailR	65.1 / 66.8	17.8 / 19.4	42.3	**39.8** / **40.6**	10.7 / 11.3	25.6	31.3 / 35.9	10.1 / 11.0	22.1
Motifs-DBiased	58.8 / 60.7	34.7 / 36.6	**47.7**	36.5 / 37.4	20.3 / 21.2	**28.9**	29.4 / 33.9	14.9 / 17.5	**24.0**

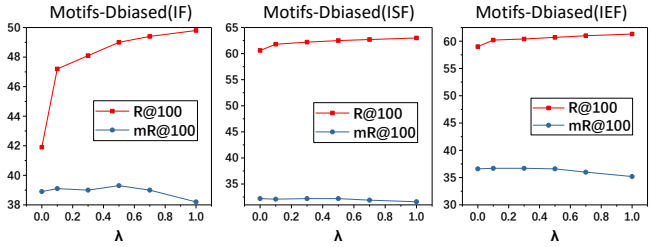


Fig. 6. R@100 and mR@100 of Motifs-DBiased(IF), Motifs-DBiased(ISF), and Motifs-DBiased(IEF) in PredCls task with different λ.

Table III shows the performance comparison, and we find that: **1)** for each re-weighting method, combined with our DBiased-P, Motifs achieves better performance on R@50/10, while maintaining the similar performance on mR50/100, which leads to the better performance on the Mean. As the R@50/100 and mR@50/100 are mainly dominated by the head and tail predicate classes predictions, respectively, we can draw the conclusion that our proposed head-oriented soft regularization is able to enhance the prediction of the head predicate classes without hurting that of the tail ones of the re-weighted classifier. **2)** Although Motifs-IF achieves the best mR@50/100, compared with Motifs-ISF and Motifs-IEF, its performance on R@50/100 is the worst. The reason may be that compared with ISF and IEF, IF re-weights the predicates most severely, which benefits the tail predicate prediction, but results in more misclassifications of head predicate samples in Motifs-IF. **3)** Our DBiased-P achieves the largest improvement for the re-weighting method IF. This indicates that our model is more effective when the proportion of the misclassified head predicate samples in the tail predicate classes is large. And **4)** Motifs-DBiased(ISF), equipped with only a naive re-weighting method, also surpasses all the baseline methods in Table I in terms of Mean. This confirms the effectiveness of our head-oriented regularization in balancing the head and tail predicate prediction.

C. Ablation Study

To thoroughly investigate the head-oriented soft regularization in our DBiased-P, we introduce the following three baseline methods:

1) Motifs-DBiased-w/o-L_{kl}: to investigate the impact of the head-oriented soft regularization, we disable it from our DBiased-P, by removing the term L_{kl} in the final objective function in Eqn.(10).

2) Motifs-HardR: to justify the effectiveness of the soft regularization, we replace the logits $\mathbf{r}_{uw}$ with the one-hot distribution, where only the ground truth predicate has the probability of one, otherwise zeros, to provide the head-oriented hard regularization.

3) Motifs-TailR: opposite to our head-oriented regularization towards the re-weighted classifier, this method uses the tail predicate predictions of the re-weighted classifier to regularize that of the unweighted classifier.

The results are shown in Table IV, from which we observe that: **1)** our Motifs-DBiased consistently achieves better performance than Motifs-DBiased-w/o-L_{kl} in terms of the Mean, which demonstrates the effectiveness of the head-oriented regularization. Meanwhile, we also find that the performance of Motifs-DBiased-w/o-L_{kl} is close to that of Motifs-IEF (see Table III), which only consists of a re-weighted classifier. This reflects that without the regularization conducted by L_{kl}, the unweighted classifier hardly affects the re-weighted one. **2)** It is unexpected that compared with Motifs-DBiased-w/o-L_{kl}, Motifs-HardR unexpectedly worsens the R@50/100, reflecting that Motifs-HardR further aggravates the head predicate predictions, which goes against our target of enhancing the head predicate predictions. The underlying reason may be that the zero elements of the one-hot ground truth label vector may largely depress the prediction of the other head predicate classes besides the ground truth one, which may affect the overall head predicate predictions of the classifier. And **3)** the improvement of Motifs-TailR over Motifs on mR@50/100 is less than that of Motifs-DBiased over Motifs. This implies that the tail-oriented regularization from the re-weighted classifier to the unweighted classifier is not as effective as the opposite one used in our DBiased-P.

D. Parameter Analysis

To investigate the effect of the regularization degree (*i.e.*, λ), we conduct experiments with different λ on our DBiased-P configured with three kinds of re-weighting methods (*i.e.*, Motifs-DBiased(IF), Motifs-DBiased(ISF), and Motifs-DBiased(IEF)).

The results are shown in Fig. 6, and we observe that: **1)** with the growing of the value of λ, the performance of R@100 in the three methods gradually increases, while that of mR@100 keep stable before $\lambda = 0.5$ and then decrease. It demonstrates that with the reasonable regularization degree, *i.e.*, $\lambda = 0.5$, our proposed head-oriented soft regularization is able to enhance the head predicate prediction and hardly hurt that of the tail predicate. And **2)** the improvement of Morifs-DBiased(IF) on R@100 is more obvious than that of Morifs-DBiased(ISE) and Morifs-DBiased(IEF) on R@100, which reconfirms that our DBiased-P is more effective when the proportion of the misclassified head predicate samples in the tail predicate classes is large.

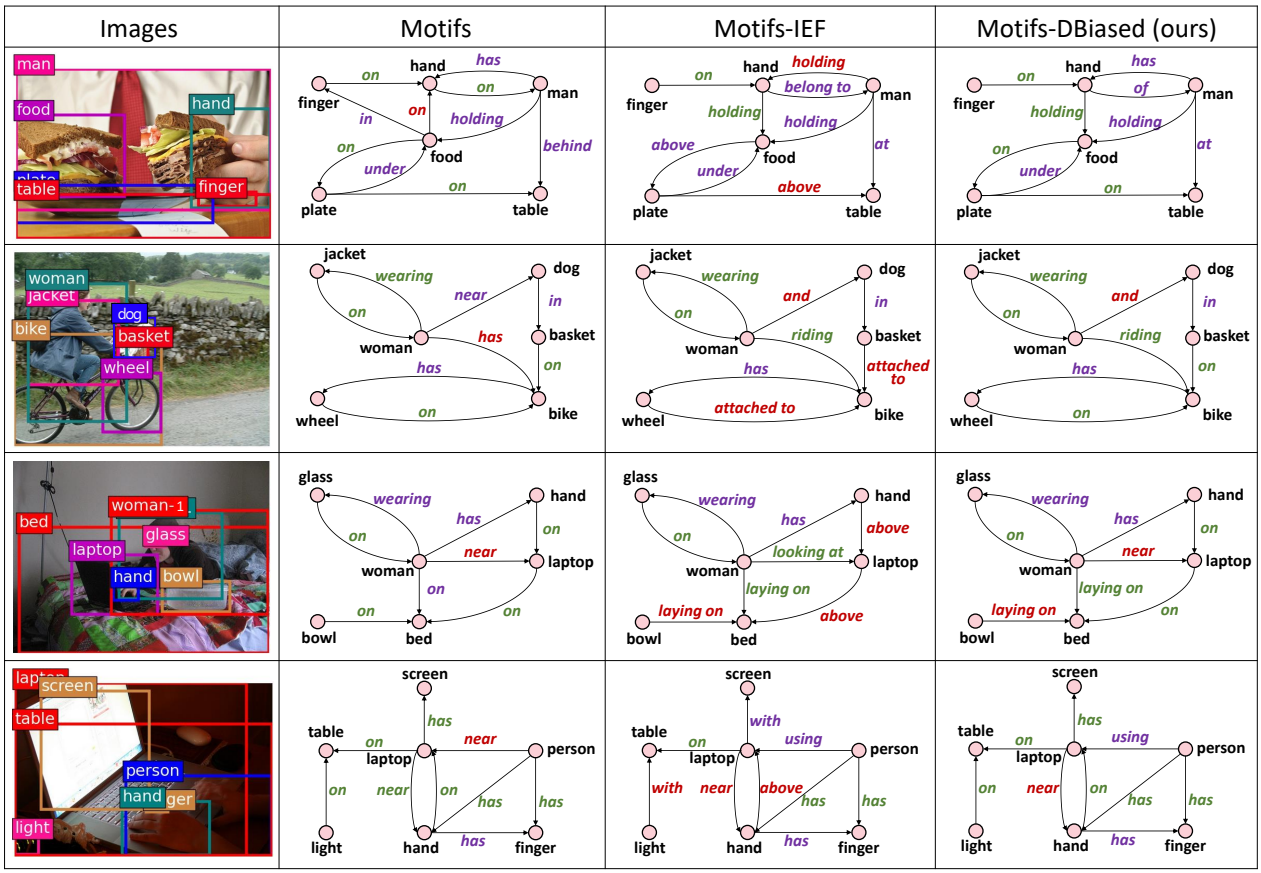


Fig. 7. Scene graph examples generated by Motifs, Motifs-IEF, and Motifs-DBiased in PredCls task with respect to R@20. Predicates in green indicate that the corresponding relationships are captured by the top 20 predicted places. Predicates in red denote the misclassified or uncaught ground truth relationships. Predicates in purple denote the reasonable captured relationships, but are not annotated as the ground truth.

E. Qualitative Results

To obtain a deep insight, we visualize four qualitative results in Fig. 7. To conduct a fair comparison, we equip our DBiased-P in Motifs (*i.e.*, Motifs-DBiased) and qualitatively compare with the general biased SGG method (*i.e.*, Motifs) as well as the advanced re-weighting SGG method (*i.e.*, Motifs-IEF). From Fig. 7, we can observe that: **1)** the scene graphs generated by our Motifs-DBiased contain more informative predicate predictions than those generated by Motifs, *e.g.*, $\langle hand,\ holding,\ food\rangle$ in the first example, $\langle women,\ riding,\ bike\rangle$ in the second example, and $\langle woman,\ laying\ on,\ bed\rangle$ in the third example. **2)** The scene graphs generated by our Motifs-DBiased have more accurate head predicate predictions than those generated by Motifs-IEF, *e.g.*, $\langle basket,\ on,\ bike\rangle$ in the second example and $\langle light,\ on,\ table\rangle$ in the fourth example. This reconfirms that our DBiased-P can improve the head predicate prediction of the re-weighting method. And **3)** as for the limitation, although our DBiased-P can improve the head predicate prediction of the re-weighting method, the improvement is limited and the head predicate prediction of DBiased-P is still inferior to that of Motifs. For example, $\langle bowl,\ on,\ bed\rangle$ in the third example and $\langle hand,\ near,\ laptop\rangle$ in the fourth example are still not captured by Motifs-DBiased. The underlying reason is that our DBiased-P is based on the re-weighting method, where the head predicate prediction is intrinsically weak. Moreover, we set the head-oriented regularization degree (*i.e.*. λ) to control our DBiased-P enhancing the head predicate prediction of the re-weighting method and also maintaining their tail predicate prediction, which makes the improvement on the head predicate prediction limited.

V. Conclusion and Future Work

In this work, we propose a DBiased-P to boost the predicate prediction for unbiased SGG, which guarantees the tail predicate prediction with the re-weighted classifier and promote the head predicate prediction via a head-oriented soft regularization from the unweighted classifier. Experiments conducted on VG and Open Image datasets indicate that our DBiased-P could achieve a better prediction trade-off between head and tail predicate classes. In addition, experiments conducted with different re-weighting methods demonstrate that our proposed head-oriented soft regularization is able to enhance the head predicate prediction without hurting the tail ones of the re-weighted classifier. In this work, we mainly rely on the metric Mean to measure the balance between the correct and unbiased SGG. In the future, we plan to design a more reasonable evaluation metric to explicitly define the trade-off between head and tail predicate predictions.

REFERENCES

[1] X. Li and S. Jiang, "Know more say less: Image captioning based on scene graphs," *IEEE Transactions on Multimedia*, vol. 21, no. 8, pp. 2117–2130, 2019.

[2] M. Yang, W. Zhao, W. Xu, Y. Feng, Z. Zhao, X. Chen, and K. Lei, "Multitask learning for cross-domain image captioning," *IEEE Transactions on Multimedia*, vol. 21, no. 4, pp. 1047–1061, 2018.

[3] J. Wu, T. Chen, H. Wu, Z. Yang, G. Luo, and L. Lin, "Fine-grained image captioning with global-local discriminative objective," *IEEE Transactions on Multimedia*, vol. 23, pp. 2413–2427, 2020.

[4] N. Ouyang, Q. Huang, P. Li, C. Yi, B. Liu, H.-f. Leung, and Q. Li, "Suppressing biased samples for robust vqa," *IEEE Transactions on Multimedia*, 2021.

[5] J. Yu, W. Zhang, Y. Lu, Z. Qin, Y. Hu, J. Tan, and Q. Wu, "Reasoning on the relation: Enhancing visual representation for visual question answering and cross-modal retrieval," *IEEE Transactions on Multimedia*, vol. 22, no. 12, pp. 3196–3209, 2020.

[6] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei, "Scene graph generation by iterative message passing," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5410–5419.

[7] R. Yu, A. Li, V. I. Morariu, and L. S. Davis, "Visual relationship detection with internal and external linguistic knowledge distillation," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1974–1982.

[8] J. Gu, H. Zhao, Z. Lin, S. Li, J. Cai, and M. Ling, "Scene graph generation with external knowledge and image reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1969–1978.

[9] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh, "Graph r-cnn for scene graph generation," in *Proceedings of the European conference on computer vision*, 2018, pp. 670–685.

[10] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi, "Neural motifs: Scene graph parsing with global context," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5831–5840.

[11] K. Tang, H. Zhang, B. Wu, W. Luo, and W. Liu, "Learning to compose dynamic tree structures for visual contexts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6619–6628.

[12] J. Zhang, K. J. Shih, A. Elgammal, A. Tao, and B. Catanzaro, "Graphical contrastive losses for scene graph parsing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 535–11 543.

[13] K. Tang, Y. Niu, J. Huang, J. Shi, and H. Zhang, "Unbiased scene graph generation from biased training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3716–3725.

[14] J. Yu, Y. Chai, Y. Hu, and Q. Wu, "Cogtree: Cognition tree loss for unbiased scene graph generation," *arXiv preprint arXiv:2009.07526*, 2020.

[15] S. Yan, C. Shen, Z. Jin, J. Huang, R. Jiang, Y. Chen, and X.-S. Hua, "Pcpl: Predicate-correlation perception learning for unbiased scene graph generation," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 265–273.

[16] R. Li, S. Zhang, B. Wan, and X. 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*, 2021, pp. 11 109–11 119.

[17] A. Desai, T.-Y. Wu, S. Tripathi, and N. Vasconcelos, "Learning of visual relations: The devil is in the tails," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 404–15 413.

[18] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov *et al.*, "The open images dataset v4," *International Journal of Computer Vision*, vol. 128, no. 7, pp. 1956–1981, 2020.

[19] Y. Bin, Y. Yang, C. Tao, Z. Huang, J. Li, and H. T. Shen, "Mr-net: exploiting mutual relation for visual relationship detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 8110–8117.

[20] Z. Cui, C. Xu, W. Zheng, and J. Yang, "Context-dependent diffusion network for visual relationship detection," in *Proceedings of the 26th ACM international conference on Multimedia*, 2018, pp. 1475–1482.

[21] M. Tajrobehkar, K. Tang, H. Zhang, and J. H. Lim, "Align r-cnn: A pairwise head network for visual relationship detection," *IEEE Transactions on Multimedia*, 2021.

[22] W. Liao, B. Rosenhahn, L. Shuai, and M. Ying Yang, "Natural language guided visual relationship detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 444–453.

[23] H. Zhou, C. Zhang, and C. Hu, "Visual relationship detection with relative location mining," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 30–38.

[24] M. Qi, W. Li, Z. Yang, Y. Wang, and J. Luo, "Attentive relational networks for mapping images to scene graphs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3957–3966.

[25] A. Zareian, S. Karaman, and S.-F. Chang, "Bridging knowledge graphs to generate scene graphs," in *European Conference on Computer Vision*, 2020, pp. 606–623.

[26] A. Kolesnikov, A. Kuznetsova, C. Lampert, and V. Ferrari, "Detecting visual relationships using box attention," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 1749–1753.

[27] X. Lin, C. Ding, J. Zeng, and D. Tao, "Gps-net: Graph property sensing network for scene graph generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3746–3753.

[28] A. Zareian, H. You, Z. Wang, and S.-F. Chang, "Learning visual commonsense for robust scene graph generation," *arXiv preprint arXiv:2006.09623*, 2020.

[29] T. Chen, W. Yu, R. Chen, and L. Lin, "Knowledge-embedded routing network for scene graph generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6163–6171.

[30] V. S. Chen, P. Varma, R. Krishna, M. Bernstein, C. Re, and L. Fei-Fei, "Scene graph prediction with limited labels," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2580–2590.

[31] C. Elkan, "The foundations of cost-sensitive learning," in *International joint conference on artificial intelligence*, vol. 17, no. 1. Lawrence Erlbaum Associates Ltd, 2001, pp. 973–978.

[32] K. M. Ting, "An instance-weighting method to induce cost-sensitive trees," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 3, pp. 659–665, 2002.

[33] C. Huang, Y. Li, C. C. Loy, and X. Tang, "Learning deep representation for imbalanced classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5375–5384.

[34] V. Lertnattee and T. Theeramunkong, "Analysis of inverse class frequency in centroid-based text classification," in *Communications and Information Technology, 2004. ISCIT 2004. IEEE International Symposium on*, 2004.

[35] D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. Van Der Maaten, "Exploring the limits of weakly supervised pretraining," in *Proceedings of the European conference on computer vision*, 2018, pp. 181–196.

[36] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[37] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9268–9277.

[38] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[39] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[41] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," 2016. [Online]. Available: https://arxiv.org/abs/1602.07332

[42] S. Sharifzadeh, S. M. Baharlou, and V. Tresp, "Classification by attention: Scene graph classification with prior knowledge," *arXiv preprint arXiv:2011.10084*, 2020.

[43] M. Suhail, A. Mittal, B. Siddiquie, C. Broaddus, J. Eledath, G. Medioni, and L. Sigal, "Energy-based learning for scene graph generation," in

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 13 936–13 945.

[44] W. Wang, R. Wang, S. Shan, and X. Chen, "Exploring context and visual pattern of relationship for scene graph generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8188–8197.

[45] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[46] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," *arXiv preprint arXiv:1503.00075*, 2015.

Xianjing Han received the B.E. degree from Northeastern University of China in 2017. She currently is a Ph.D. student from the School of Computer Science and Technology at Shandong University, supervised by Liqiang Nie and Xuemeng Song. Her research interest comprises multimedia computing and computer vision. She has published several papers in the top venues, such as ACM SIGIR, MM and IEEE TIP.

Xuemeng Song received the B.E. degree from University of Science and Technology of China in 2012, and the Ph.D. degree from the School of Computing, National University of Singapore in 2016. She is currently an assistant professor of Shandong University, Jinan, China. Her research interests include the information retrieval and social network analysis. She has published several papers in the top venues, such as ACM SIGIR, MM and TOIS. In addition, she has served as reviewers for many top conferences and journals.

Xingning Dong received the B.E. degree from the School of Computer Science and Communication Engineering, Jiangsu University, in 2020, where he is currently pursuing the master's degree with the School of Computer Science and Technology, Shandong University. His research interests include computer vision, multimodal pretraining, and scene understanding.

Yinwei Wei received his MS degree from Tianjin University and Ph.D. degree from Shandong University, respectively. Currently, he is a research fellow with NExT, National University of Singapore. His research interests include multimedia computing and recommendation. Several works have been published in top forums, such as ACM MM, IEEE TMM and TIP. Dr. Wei has served as the PC member for several conferences, such as MM, AAAI, and IJCAI, and the reviewer for TPAMI, TIP, and TMM.

Meng Liu is currently a Professor with the School of Computer Science and Technology, Shandong Jianzhu University. Various parts of her work have been published in top forums and journals, such as SIGIR, MM, and IEEE TRANSACTIONS ON IMAGE PROCESSING (TIP). Her research interests include multimedia computing and information retrieval. She has served as a Reviewer and a Subreviewer for various conferences and journals, such as MMM, MM, PCM, JVCI, and INS.

Liqiang Nie is currently a professor with the Shandong University. Meanwhile, he is the adjunct dean with the Shandong AI institute. He received his B.Eng. and Ph.D. degree from Xi'an Jiaotong University and National University of Singapore (NUS), respectively. After PhD, Dr. Nie continued his research in NUS as a research follow for three and half years. His research interests lie primarily in multimedia analysis and search. Dr. Nie has published more than 150 papers and received around 13,000 Google scholar citations. He is an AE of Information Science, IEEE TKDE, IEEE TMM, and ACM ToMM. He received many awards, like SIGIR best paper honorable mention in 2019, ACM MM best paper finalist in 2019, SIGIR best student paper in 2021, SIGMM Rising Star in 2020, TR35 China, and DAMO Academy Young Fellow in 2020.