Copyright ⓒ 2023 KSII

A Novel Two-Stage Training Method for Unbiased Scene Graph Generation via Distribution Alignment

Dongdong Jia[1], Meili Zhou[1], Wei WEI[2], Dong Wang[1] and Zongwen Bai[1*]
[1] School of physics and electronics information, Yanan University
Yanan, Shaanxi 716000 CN
[e-mail: jiadd321@qq.com, zml@yau.edu.cn, dongwang@yau.edu.cn]
[2] School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048; Shaanxi Key Laboratory for Network Computing and Security Technology,China.
[E-mail: weiwei@xaut.edu.cn]
[*]Corresponding author:Meili Zhou, Zongwen Bai

Received October 1, 2023; revised October 26, 2023; accepted December 5, 2023;
published December 31, 2023

Abstract

Scene graphs serve as semantic abstractions of images and play a crucial role in enhancing visual comprehension and reasoning. However, the performance of Scene Graph Generation is often compromised when working with biased data in real-world situations. While many existing systems focus on a single stage of learning for both feature extraction and classification, some employ Class-Balancing strategies, such as Re-weighting, Data Resampling, and Transfer Learning from head to tail. In this paper, we propose a novel approach that decouples the feature extraction and classification phases of the scene graph generation process. For feature extraction, we leverage a transformer-based architecture and design an adaptive calibration function specifically for predicate classification. This function enables us to dynamically adjust the classification scores for each predicate category. Additionally, we introduce a Distribution Alignment technique that effectively balances the class distribution after the feature extraction phase reaches a stable state, thereby facilitating the retraining of the classification head. Importantly, our Distribution Alignment strategy is model-independent and does not require additional supervision, making it applicable to a wide range of SGG models. Using the scene graph diagnostic toolkit on Visual Genome and several popular models, we achieved significant improvements over the previous state-of-the-art methods with our model. Compared to the TDE model, our model improved mR@100 by 70.5% for PredCls, by 84.0% for SGCls, and by 97.6% for SGDet tasks.

Keywords: Scene Graph Generation, Transformer-based Architecture, Distribution Alignment, Model-independent, Visual Genome Dataset.

http://doi.org/10.3837/tiis.2023.12.009 **ISSN : 1976-7277**

1. Introduction

The ultimate goal of computer vision is to develop intelligent systems that can extract valuable insights from images, videos, and other visual data with the same level of mastery as humans. Capturing the relationships between objects in a scene is often crucial for achieving higher-level visual comprehension and reasoning tasks [1-4], serving as a driving force for such endeavors. Scene Graph Generation (SGG) aims to address this challenge by employing data structures known as scene graphs. These graphs describe the instances of objects within a scene and their relationships, effectively encoding images into abstract semantic components. As early as 2015, researchers proposed leveraging the visual features of objects in an image and their relationships as a means to accomplish various computer vision tasks, include Visual Question Answering [1, 3-5], Image Captions [2, 6, 7], and other related jobs in the field of Computer Vision [8-10]. Visual relationship mining has been demonstrated to significantly enhance the performance of relevant computer vision tasks, facilitating improved understanding and reasoning of visual scenes by machines.

However, the task of SGG is currently facing practical challenges primarily due to a high number of low-semantic predictions, which limits the application of scene graphs. There are several factors contributing to this issue. Firstly, in widely used representative datasets, the distribution of relational samples is highly imbalanced, with a long-tailed distribution [11, 12]. For instance, in the widely used Visual Genome (VG) dataset [10, 13], the sum of the top and middle relations comprises more than half of the total relations. Consequently, the network exhibits a strong preference for commonly occurring relations, while struggling to accurately predict relations in less frequent classes or tail classes. For example, there are only a few low-semantic head predicates (such as "on" and "has") that have a large and diverse set of instances, which dominate the training process. On the other hand, a small number of highly informative tail predicates (like "riding" and "watching") often tend to be misclassified into head classes, resulting in insufficient accuracy or reliability for downstream tasks. Moreover, due to visual similarity and sparse training data, distinguishing fine-grained relations (such as "standing on," "sitting on," and "flying on") from one another can be more challenging. Nonetheless, we should not attribute the biased training solely to the distribution of samples. In reality, the distribution of actual samples typically follows a long-tail distribution, where head categories have a higher number of instances compared to tail categories [14]. The majority of biased comments can aid the model in learning valuable contextual priors [15, 16] to streamline candidate searches and eliminate unnecessary ones.

To tackle this issue, extensive research has focused on developing unbiased models through approaches such as Re-weighting [17], enhancing network structures [18, 19], and distinguishing unbiased from biased representations [14]. These approaches primarily focus on the collaborative training of feature extractors and classifiers. However, it remains unclear how this joint learning scheme enhances relational predictive power. Does it achieve this by learning more effective features or by adjusting classifier decision boundaries to better process the data? To address this question, we divide the SGG process into two distinct processes: feature extraction and classification. We utilize transformer-based structures for feature learning, followed by the Distribution Alignment (DA) method [20-22] to adjust each classification probability distribution. Initially, we train the feature extractor and classifier uniformly on the original dataset. Once the model stabilizes, we fix the feature extractor parameters and retrain the classification head utilizing a DA algorithm [20-22]. This strategy calibrates the output of the classifier by comparing it to a reference distribution of classes that promotes balanced predictions. By incorporating class priors and data inputs, we can employ this alignment approach to systematically learn class decision boundaries.

Our contributions can be summarized as follows:

(1) We provide evidence that the pronounced bias in the SGG results is primarily driven by imbalanced decision boundaries rather than the process of feature learning. To demonstrate this, we decouple the feature extraction and classification stages of the SGG process and achieve surprising results.

(2) Building upon these findings, we propose a two-stage training approach that is model-independent. In the second stage, we utilize a DA algorithm [20-22] to realign the decision boundaries of the SGG predicate classifier.

(3) Extensive experiments are conducted on the existing models, demonstrating that our DA strategy [20-22] consistently and significantly enhances performance. As presented in **Fig. 1**, the Motifs [16] method shows notable improvement across all three tests at mean Recall@100.

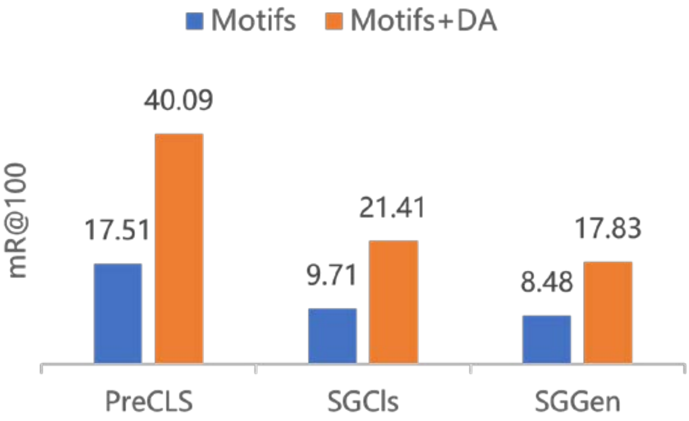


Fig. 1. MR@100 improvement on three tasks over Motifs.

2. Related Work

2.1 Scene Graph Generation

SGG generates visual representations of images in the form of graphical abstractions, facilitating visual relational reasoning and comprehension across various downstream tasks [11, 23-25]. Previous research [3, 4] concentrated on object recognition and relationship detection using independent networks while ignoring the rich contextual information. To incorporate global visual features, recent works have used more powerful feature refinement modules to encode rich contextual information, such as message-passing mechanisms [8, 26], various LSTMs [16, 18], graph neural networks [2, 11], and self-attention networks [27, 28]. Due to the significant bias present in the dataset, although object recognition achieves high accuracy, relationship detection falls far short of providing satisfactory support for downstream vision and language tasks. To mitigate biased relation prediction, several methods have been proposed, including debiasing strategies such as Re-sampling [29] or Re-weighting [30], separating unbiased representations from biased relations [14], and filtering out irrelevant predicates using tree structures [18, 19]. However, these methods often prioritize overfitting the head class at the expense of the tail class. In this paper, we propose a novel two-stage training strategy that separates the SGG process into feature extraction and classification stages to effectively tackle the problem of long-tail distribution. Additionally, we employ DA [20-22] to recalibrate the probability distribution of each class, eliminating the need for sampling strategies, complex loss functions, or additional storage modules.

2.2 Unbiased Classification

In previous studies, classification methods on highly biased training data were extensively investigated to mitigate the long-tailed distribution problem in visual tasks. This can be classified into three categories: (1) balancing data distribution through data augmentation and resampling [31-35], removing bias from learning strategies through Re-weighting losses [35] and well-designed network structures [18, 19, 36, 37]; (2) distinguishing biased and unbiased models for prediction [14, 38, 39]; and (3) separating the classifier from the end-to-end learning approach [12, 22, 40-42] and then rebalancing the classifier to improve long-tail prediction. Our scheme falls into the third category but differs from existing methods in that we use a two-stage training strategy to decouple feature extraction and classifier, and we only adjust the classification head in the second stage using a DA algorithm [20-22].

3. APPROACH

A scene graph is a structured data representation that encodes object instances as nodes and relations between objects as edges, capturing the content of an image scene. In IMP [26], the objective of Scene Graph Generation (SGG) is to accurately create a graph that maps the image in a correct manner. A scene graph can be mathematically defined as $\mathrm{G} = \{B, O, R\}$, where B represents bounding boxes, O represents object labels, and R represents relationship labels. The probability distribution of the scene graph $\mathrm{P}(G|I)$ is typically decomposed into three components when given an image I, as shown in (1).

$$P(G \mid I) = P(B \mid I) P(O \mid B, I) P(R \mid O, B, I) \tag{1}$$

To begin with, the probability distribution $\mathrm{P}(B|I)$ is modeled using the commonly used pre-trained Faster R-CNN [43], which generates a set of bounding box suggestions. Subsequently, the object detection model $P(O|B,I)$ predicts the object labels for each bounding box based on the potential proposals. Finally, the relationship prediction model $P(R|O,B,I)$ is employed to infer the relationship between each pair of objects and generate the scene graph for the given image based on the object detection results. In our two-stage learning model, we follow the aforementioned framework in the first stage. In the second stage, $P(R|O,B,I)$ is decomposed into $P(L|O,B,I)$ and $P(R|L)$ (where L represents the probability distribution of the predicates). Then, the classification head $P(R|L)$ is adjusted using a Distribution Alignment (DA) algorithm [20-22], which can be mathematically expressed as (2):

$$P(G \mid I) = P(B \mid I) P(O \mid B, I) P(L \mid O, B, I) P(R \mid L) \tag{2}$$

3.1 Overall Framework

Our two-stage learning model, depicted in **Fig. 2**, can be summarized in four stages. (a) We employ a pre-trained Faster RCNN-FPN [43, 44] to extract K targets, denoted as $\mathrm{O} = \{o_i\}^k$, where each target has visual features $o_i \in \mathbb{R}^{4096}$ and spatial features $b_i \in \mathbb{R}^4$. Additionally, we extract G object pairs, denoted as $\mathrm{U} = \{u_{ij}\}^G$, with visual features $u_{ij} \in \mathbb{R}^{4096}$. (b) We generate object features using a transformer-based Object Encoder that leverages both the visual features o_i and spatial features b_i of the target. This allows us to dynamically collect multiple contextual information for each object without the constraints of sequential input. The Object Decoder comprises of a fully connected layer followed by a Softmax layer, which refines object predictions and bounding boxes. To capture the contextual semantics and generate edge features, we concatenate the visual features u_{ij} of the object pair, the label embedding $l_{ij} \in \mathbb{R}^{400}$ produced by the Object Decoder, and the spatial feature embedding $b_{ij} \in \mathbb{R}^{256}$. We then pass this concatenated input through another transformer-based encoder, known as the Relation Encoder. The Relation Encoder is also built on a transformer encoder and is responsible for generating edge features. The Relation Decoder is a fully connected layer used to generate relation features $r \in \mathbb{R}^{2052}$. (c) To predict the probability distribution of predicates, we concatenate the linguistic prior features calculated using the object pair labels l_{ij}, relation features r, and visual features u_{ij}. This concatenated input is then fed to the Predicate Encoder, and the Predicate Decoder consists of a fully connected layer followed by a Softmax layer. (d) We have redesigned the classifier head to distinguish between feature extraction and classification processes. The new classifier head consists of two calibration parameters and an adaptive calibration function. These parameters are used to linearly modify the scores of each category, and the adaptive function combines the original and modified category values in an adaptive manner. Once the model has stabilized, we will retrain the classifier head using DA techniques [20-22] to adjust the classifier's decision boundaries.

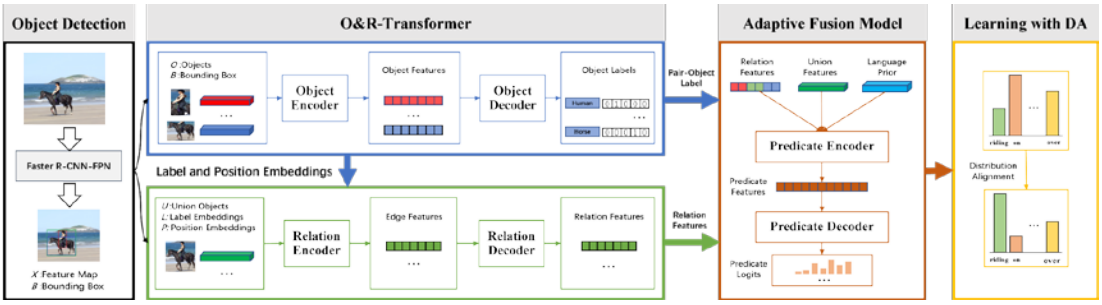


Fig. 2. The overview of our model.

3.2 Object and Relation Transformer

The fact that certain objects consistently appear together in specific scenes provides valuable prior knowledge and serves as an informative cue for expanding the semantic understanding of these objects. In previous works such as Motifs [16] and VCTree [18], LSTMs/TreeLSTMs were used to encode and decode the co-occurrence relationships among objects. However, in this paper, we employ a transformer-based encoder to capture diverse contextual information specific to each object in an adaptive manner, without being constrained by sequential input limitations. To achieve this, we concatenate the visual features o_i of the object with the object label embedding l_i^0 and the spatial feature embedding b_i. These concatenated features are then passed through a fully connected layer and serve as the input to the Object Encoder, as illustrated in (3).

In addition to possessing a local structure, scene graphs also exhibit a higher-order structure. For instance, if the “eye of cat” is present, it is highly likely that the “ear of cat” will also be present. Similarly, when an image features the “head of elephant”, it is probable that the “legs of elephant”, “trunk of elephant”, and “ears of elephant” will follow suit. Many patterns observed in the VG dataset [13] involve combinations of parts or objects, which are typically grouped together if they appear together at least 50 times and are at least 10 times more likely to co-occur than to occur separately. Over 50% of the images in the dataset showcase a parent motif comprising at least two object-relationship-object combinations [16]. Contrary to previous research that solely encoded the visual features of subjects and objects, we discovered that the representation of relationships is also enhanced by the union visual features of object pairs, denoted as u_{ij}. Consequently, we merge the label embedding of the object pair l_{ij}, the location embedding b_{ij}, and the union visual feature u_{ij}, as illustrated in (4).

$$Input_{OT} = f\left(cat\left(o_i, l_i^0, b_i\right)\right) \tag{3}$$

$$Input_{RT} = f\left(cat\left(u_{ij}, l_{ij}, s_{ij}\right)\right) \tag{4}$$

where f represents the fully connected layer. The structure of the Object and Relation Encoder, shown in **Fig. 3**, enables dynamic capturing of contextual semantics. The edge features $e \in \mathbb{R}^{2052}$ are then passed through a fully connected layer. Both the Object and Relation Transformers employ fully-connected layers as decoders.

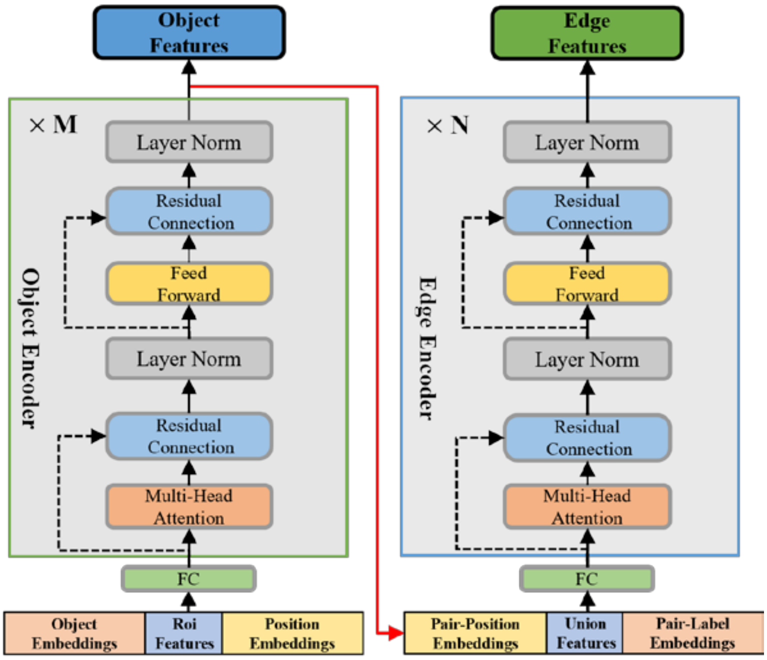


Fig. 3. The structure of Object and Relation Encoder.

3.3 Adaptive Fusion Model.

We only use the adjusted object pair labels from the Object Transformer and calculate the relationships between the target pairs using language prior knowledge, mapping the results to $\mathbb{R}^{2052}$. The joint visual features u_{ij} are projected into the $\mathbb{R}^{2052}$ space. To dynamically integrate the joint visual features u_{ij}, relation features r_{ij}, and language prior features f_{ij}, we propose a Predicate Encoder equipped with an attention score function and a fusion function, as illustrated in (5) and (6).

$$\alpha_u, \alpha_r, \alpha_f = \sigma\left(cat\left(u_{ij}, r_{ij}, f_{ij}\right)\right) \tag{5}$$

$$p_{ij} = \left(1+\alpha_u\right)\cdot u_{ij} + \left(1+\alpha_r\right)\cdot r_{ij} + \left(1+\alpha_f\right)\cdot f_{ij} \tag{6}$$

where $\sigma(\cdot)$ denotes the sigmoid function. For each object pair, the fused features p_{ij} are inputted into the Predicate Decoder, which consists of a fully connected layer, to generate a probability distribution L_{ij}^0. over the predicates.

3.4. Classifier Head

The SGG process can be divided into two parts: feature extraction $F(\cdot)$ and classification $H(\cdot)$. In the initial step, we jointly train $F(\cdot)$ and $H(\cdot)$. Once the model stabilizes, we fix the parameters of $F(\cdot)$ and retrain $H(\cdot)$ using DA in the subsequent phase. In the feature extraction phase, we obtain the predicate features p_{ij} and the initial predicate probability distribution L_{ij}^0. We then fine-tune the probability distribution of each predicate. We introduce an adaptive calibration strategy based on two calibration parameters and an adaptive calibration function. Specifically, we denote the class fraction in L_{ij}^0 as $L_{ij}^0 = [L_1^0, \cdots, L_C^0]$ and introduce a class-specific linear transformation to modify the fraction, which is described as (7).

$$L_j = \alpha_j \cdot L_j^0 + \beta_j, \forall j \in C \tag{7}$$

where the calibration parameters α_j and β_j are learned from the data set for each class. As mentioned earlier, we define a confidence score function $\sigma(p_{ij})$, which is an adaptive combination of the original probability distribution and the transformed class scores (Next, we replace p_{ij} with x), as illustrated in (8).

$$\hat{L}_j = \sigma\left(x\right)\cdot L_j + \left(1-\sigma\left(x\right)\right)\cdot L_j^0 = \left(1+\sigma\left(x\right)\alpha_j\right)\cdot L_j^0 + \sigma\left(x\right)\cdot \beta_j \tag{8}$$

The confidence score $\sigma(x)$ is computed for all inputs x through a linear layer followed by a nonlinear activation function, such as the sigmoid function. The purpose of $\sigma(x)$ is to determine the amount of calibration required for a given input x. Using the calibrated class fraction, we utilize the softmax function to create a predictive distribution for our model, as illustrated in (9).

$$p_m\left(y=j\middle|x\right) = \frac{exp\left(\hat{z}_j\right)}{\sum_{k=1}^{C} exp\left(\hat{z}_k\right)} \tag{9}$$

3.4. Distribution Alignment

We propose a method that predicts $p_m(\cdot)$ in a balanced way by aligning it with reference distribution of the category, given the training set. Formally, the category reference distribution is denoted as $p_r(y|x)$, and our goal is to minimize the expectation of the KL-dispersion between $p_r(y|x)$ and the model prediction $p_m(y|x)$, as illustrated in (10).

$$\begin{aligned}\mathcal{L} &= \mathbb{E}_{\mathcal{D}_{tr}}\left[\mathcal{KL}\left(p_r\left(y|\ x\right) || \ p_m\left(y|\ x\right)\right)\right] \\ &\approx -\frac{1}{N}\sum_{i=1}^{N}\left[\sum_{y\in\mathcal{C}} p_r\left(y|\ x_i\right) log\left(p_m\left(y|\ x_i\right)\right)\right] + C\end{aligned} \tag{10}$$

The expectation is approximated by the empirical mean on the training dataset $\mathcal{D}_{tr}$, where C is a constant. We apply a method that assigns different weights to different classes to align the distribution of the training data with the reference distribution. We use a weighted empirical distribution as the reference distribution, as illustrated in (11).

$$p_r\left(y = c|\ x_i\right) = w_c \cdot \mathcal{F}_c\left(y_i\right), \forall c \in C \tag{11}$$

We use $\mathcal{F}_c(y_i)$ to denote a function that is 1 when $y_i = c$ and 0 otherwise, and w_c to represent the weight of class c. We calculate the reference weights from the training data by using the observed frequencies of each class $r = [r_1, \cdots, r_k]$, as illustrated in (12).

$$w_c = \frac{\left(1/r_c\right)^{\rho}}{\sum_{k=1}^{K}\left(1/r_k\right)^{\rho}}, \forall c \in C \tag{12}$$

where ρ is a scaling hyperparameter.

4. Experimental Results and Analysis

In this section, we provide a detailed description of the implementation details for our research. Subsequently, we present the quantitative results and conduct a qualitative analysis using the Visual Genome (VG) dataset.

4.1 Implementation

Based on previous literature, we employed a pre-trained Faster R-CNN [43] with Resnet-101-FPN [44, 45] as the underlying detector for our SGG model, and fix its parameters. The input images were scaled to have a longer side of 1k pixels. The Object Transformer consisted of 4 transformer encoders, while the Relation Transformer had 2 transformer encoders, both with 12 attention heads. For the DA using the generalized reweighting, we set the parameter ρ to a fixed value of 1. We use gradual warmup with a starting learning rate of 0.0024, then return to the set learning rate of 0.024 after 500 iterations to continue training, and then reduce the learning rate by a factor of 10 when the loss plateaus, with a minimum of 0.00024. To train our model, we utilized the SGD algorithm with a batch size of 12. All experiments were conducted successfully using PyTorch on two NVIDIA Tesla V100 GPUs.

4.2. Quantitative Results

Dataset: We use the VG dataset [13] to evaluate our model. It has 108K images, 75K object classes, and 37K predicate classes. We follow the standard VG split [26, 46], which selects the top 150 object classes and 50 predicate classes, because 92% of the predicate classes have fewer than 10 samples. The VG split only provides the training and test sets, so we take 5K validation sets from the training set as done in previous work [14, 19].

Tasks and Evaluation: The SGG task is divided into three subtasks based on the prior work[16]: (1) Predicate classification (PredCls) only predicts predicate labels. (2) Scene graph classification (SGCls) forecasts object and predicate labels based on the ground-truth bounding boxes of labels. (3) Scene graph detection (SGDet) predicts bounding boxes, object and predicate labels. We adopt the unbiased metric mean Recall@k (mR@k)[14] to measure the unbiased scene graph, which computes Recall@k for each class individually and averages the Recall@k for all classes separately.

Comparison: We evaluated our debiasing method against the state-of-the-art debiasing methods TDE [14] and CogTree [19] on three basic models: Motifs [16], VCTree [18], and O&R-Transformer. All of the above models share the same pre-trained Faster R-CNN [43] detection. We also compared O&R-Transformer to other biased models, including GT-Transformer [19], Motifs [16], and VCTree [18]. The following are our observations from **Table 1:** (1) Distribution Alignment (DA) is an effective method that can improve the mR@k of all biased models across all evaluation tasks and outperforms other debiasing techniques. (2) On the baseline, our biased model O&R-Transformer outperforms existing SGG models [16, 18, 19], demonstrating that the contextual information gathered by the transformer structure is helpful for distinguishing between relation representations. (3) When compared to GT-Transformer [19], our model performs better in both PredCls and SGDet, demonstrating that the union visual features of object pairs are very beneficial for relation prediction, and slightly worse in the SGCls baseline, indicating that accurate object recognition aids in relation identification.

Table 1. Comparison on Visual Genome dataset

Model	SGDet		SGCls		PredCls	
	mR@50	mR@100	mR@50	mR@100	mR@50	mR@100
Motifs [16]	7.3	8.5	9.2	9.7	16.1	17.5
SG-Transformer [19]	7.6	8.9	10.8	11.5	17.0	18.3
VCTree [18]	7.5	8.7	**11.1**	**11.8**	17.6	18.9
O&R-Transformer(Ours)	**8.0**	**9.5**	10.2	10.7	**18.5**	**20.1**
Motifs + TDE [14, 16]	8.9	10.9	13.2	15.7	21.7	25.6
Motifs + CogTree [16, 19]	11.8	13.5	17.1	18.4	29.5	32.1
Motifs + DA	**14.8**	**17.8**	**20.2**	**21.4**	**37.8**	**40.1**
VCTree + TDE [14, 18]	9.1	11.0	16.0	18.1	24.8	28.1
VCTree + CogTree [18, 19]	10.2	12.1	15.2	16.2	27.0	29.5
VCTree + DA	**13.0**	**15.6**	**25.1**	**26.3**	**36.6**	**38.6**
Ours + TDE	7.0	8.3	10.6	11.9	20.1	23.4
Ours + CogTree	10.9	12.7	15.6	16.4	28.1	30.1
Ours + DA	**13.7**	**16.4**	**20.7**	**21.9**	**37.3**	**39.9**

Fig. 4 also displays Recall@100 for the top 25 common predicates. Motifs + DA outperforms Motifs [16] on the majority of tail classes while decreasing on a few head classes, indicating that the increase in mR@K is due mainly to tail classes rather than head classes. While the decreasing head classes are mostly coarse relations like “on”, “has”, and “near”, the improving tail classes are mostly fine-grained relations like “standing on”, “in front of”, and “sitting on”, demonstrating that DA can successfully detect fine-grained relations.

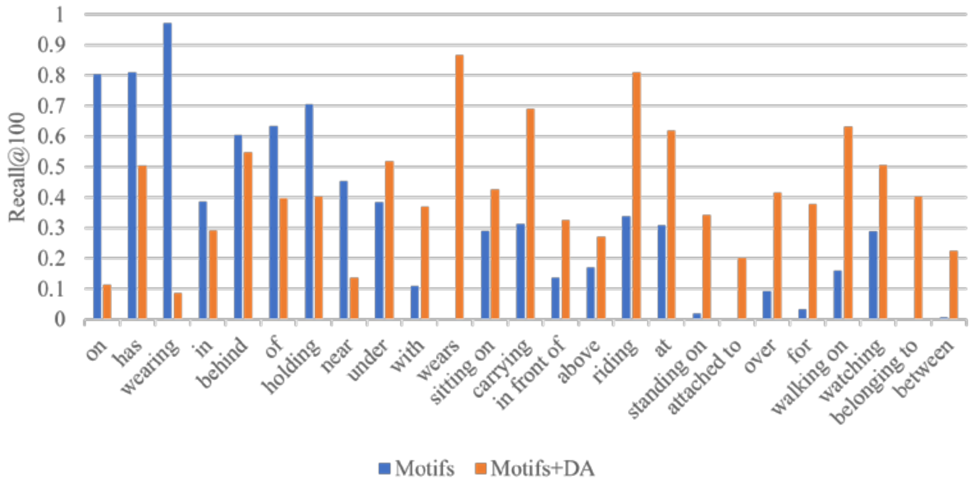


Fig. 4. The predicate of Recall@100 on PredCls.

4.3 Ablation Study

Table 2 presents the ablation results, which aim to validate the contribution of each component in our model. Models 1-3 are used to evaluate the effectiveness of individual components. Model 4, based on the feature fusion method proposed in TDE [14], demonstrates that the combination of union visual features and language prior features provides complementary information for the SGG task compared to Model 1. Model 2 improves upon Model 1 by introducing the Adaptive Fusion Module (AFM), which adaptively combines relational features, joint visual features, and linguistic prior features. This enhancement leads to superior performance compared to baseline models, highlighting the effectiveness of our AFM in seamlessly integrating these features. Model 3 expands upon Model 1 by incorporating a classification head, albeit with just one stage of end-to-end training. The findings reveal that the model's performance, solely based on the classification head, exhibits marginal improvement compared to the baseline model in the PredCls task. This suggests that the decision boundaries established by the classifier, rather than achieving enhanced feature representations, play a crucial role in influencing the long-tail recognition ability.

Table 2. Ablation study of key components in our model.

Model		SGDet		SGCls		PredCls	
		mR@50	mR@100	mR@50	mR@100	mR@50	mR@100
1	O&R-Transformer(base)	7.4	8.8	9.8	10.4	17.6	19.0
2	Base + AFM	8.0	9.5	10.2	10.7	18.5	20.1
3	Base + Classifier Head	7.5	8.8	9.7	10.3	17.8	19.3
4	Base + SUM[14]	8.1	9.5	10.4	11.0	18.3	19.8

4.4 Qualitative Analysis

Fig. 4 shows several instances of PredCls generated by O&R-Transformer (gray) and O&R-Transformer + DA (yellow). Despite having the strongest baseline of the four biased models, O&R-Transformer makes significant improvements when equipped with DA:(1) When applying DA, the model forecasts more precise and differentiated relationships than when only using O&R-Transformer. The baseline is inclined to predict plausible but insignificant head classes, as seen in **Fig. 4** whereas our model correctly identifies more fine-grained and useful relations like walking on, riding on, and in front of. This is primarily due to the fact that DA successfully reassigns cumbersome head class links to more precise fine-grained ones. (2) DA allows the model to visibly and semantically differentiate comparable relations, which is challenging for O&R-Transformer to do. The O&R-Transformer wrongly forecasts the difference between walking on and standing on because it cannot distinguish between these two actions. By shifting the predicate categorization led by visual characteristics, our model can effectively differentiate identical relations and forecast relations to more accurate classes.

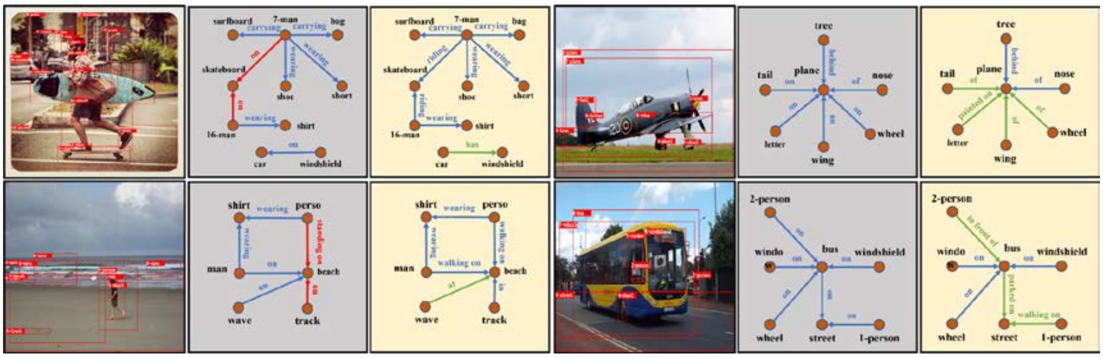


Fig. 4. The scene graphs produced by O&R-Transformer(gray) and O&R-Transformer + DA(yellow).

5. Conclusion

In this paper, we propose a novel two-stage training strategy to address the long-tail distribution problem in Scene Graph Generation. Our method involves decoupling the process of SGG into feature extraction and classification stages, and applies a confidence-aware Distribution Alignment scheme to balance the predicate classes. We demonstrate that our method can significantly improve the performance of SGG on highly biased datasets, achieving extraordinary results on several metrics. Our method is also model-independent and can be easily integrated with existing SGG models. We believe that our method can facilitate the development of more robust and unbiased scene graph generation systems for various applications.

Acknowledgement

This work was support by national key R&D program of China (Grant. NO 2022YFE0138600), national Natural science foundation of China (Grant No. 62266045) and Key projects of Shaanxi Yan'an (Grant No. 2021JB-04).

References

[1] C. Zhang, W.-L. Chao, and D. Xuan, "An Empirical Study on Leveraging Scene Graphs for Visual Question Answering, " *arXiv preprint arXiv:1907.12133*, Jul. 2019. Article (CrossRef Link).

[2] Y. Zhong, L. Wang, J. Chen, D. Yu, and Y. Li, "Comprehensive Image Captioning via Scene Graph Decomposition," in *Proc. of Computer Vision–ECCV 2020: 16th European Conference*, pp. 211–229, 2020. Article (CrossRef Link).

[3] Z. Bai, Y. Li, M. Woźniak, M. Zhou, and D. Li, "DecomVQANet: Decomposing visual question answering deep network via tensor decomposition and regression," *Pattern Recognition*, vol. 110, p. 107538, 2021. Article (CrossRef Link).

[4] D. Teney, L. Liu, and A. Van Den Hengel, "Graph-Structured Representations for Visual Question Answering," in *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2017. Article (CrossRef Link).

[5] Z. Bai et al., "Bilinear Semi-Tensor Product Attention (BSTPA) model for visual question answering," in *Proc. of 2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Jul. 2020. Article (CrossRef Link).

[6] J. Gu, S. Joty, J. Cai, H. Zhao, X. Yang, and G. Wang, "Unpaired Image Captioning via Scene Graph Alignments," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, IEEE, pp. 10322–10331, Oct. 2019. Article (CrossRef Link).

[7] N. Xu, A.-A. Liu, J. Liu, W. Nie, and Y. Su, "Scene graph captioner: Image captioning based on structural visual representation," *J. Vis. Commun. Image Represent*., vol. 58, pp. 477–485, 2019. Article (CrossRef Link).

[8] Y. Teng, L. Wang, Z. Li, and G. Wu, "Target Adaptive Context Aggregation for Video Scene Graph Generation," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, IEEE, pp. 13668–13677, Oct. 2021. Article (CrossRef Link).

[9] M. Zhou, X. Liu, T. Yi, Z. Bai, and P. Zhang, "A superior image inpainting scheme using Transformer-based self-supervised attention GAN model," *Expert Syst. Appl.*, vol. 233, p. 120906, Dec. 2023. Article (CrossRef Link).

[10] Y. Guo, J. Chen, H. Zhang, and Y.-G. Jiang, "Visual relations augmented cross-modal retrieval," in *Proc. of the 2020 International Conference on Multimedia Retrieval*, pp. 9–15, 2020. Article (CrossRef Link).

[11] G. Zhu et al., "Scene Graph Generation: A Comprehensive Survey," *arXiv preprint arXiv:2201.00443*, Jun. 2022. Article (CrossRef Link).

[12] K. Tang, J. Huang, and H. Zhang, "Long-Tailed Classification by Keeping the Good and Removing the Bad Momentum Causal Effect," *Advances in Neural Information Processing Systems*, 33, 2023.

[13] R. Krishna et al., "Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations," *Int. J. Comput. Vis.*, vol. 123, no. 1, pp. 32–73, May 2017. Article (CrossRef Link).

[14] K. Tang, Y. Niu, J. Huang, J. Shi, and H. Zhang, "Unbiased Scene Graph Generation From Biased Training," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 3713–3722, Jun. 2020. Article (CrossRef Link).

[15] J. Jung and J. Park, "Visual Relationship Detection with Language prior and Softmax," in *Proc. of 2018 IEEE International Conference on Image Processing, Applications and Systems (IPAS)*, pp. 143–148, 2018. Article (CrossRef Link).

[16] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi, "Neural Motifs: Scene Graph Parsing with Global Context," in *Proc. of the IEEE conference on computer vision and pattern recognition*, IEEE, pp. 5831–5840, Jun. 2018. Article (CrossRef Link).

[17] B. Kang, Z. Liu, X. Wang, F. Yu, J. Feng, and T. Darrell, "Few-Shot Object Detection via Feature Reweighting," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, Seoul, IEEE, pp. 8419–8428, Oct. 2019. Article (CrossRef Link).

[18] K. Tang, H. Zhang, B. Wu, W. Luo, and W. Liu, "Learning to Compose Dynamic Tree Structures for Visual Contexts," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 6612–6621, Jun. 2019. Article (CrossRef Link).

[19] J. Yu, Y. Chai, Y. Wang, Y. Hu, and Q. Wu, "CogTree: Cognition Tree Loss for Unbiased Scene Graph Generation," *arXiv preprint arXiv:2009.07526*, 2020. Article (CrossRef Link).

[20] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Strong-Weak Distribution Alignment for Adaptive Object Detection," in *Proc. of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 6949–6958, Jun. 2019. Article (CrossRef Link).

[21] D. Berthelot et al., "ReMixMatch: Semi-Supervised Learning with Distribution Alignment and Augmentation Anchoring," *arXiv preprint arXiv:1911.09785*, Feb. 2020. Article (CrossRef Link).

[22] S. Zhang, Z. Li, S. Yan, X. He, and J. Sun, "Distribution Alignment: A Unified Framework for Long-tail Visual Recognition," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 2361–2370, Jun. 2021. Article (CrossRef Link).

[23] R. Y. Zakari, J. W. Owusu, H. Wang, K. Qin, Z. K. Lawal, and Y. Dong, "VQA and Visual Reasoning: An Overview of Recent Datasets, Methods and Challenges," *arXiv preprint arXiv:2212.13296*, Dec. 2022. Article (CrossRef Link).

[24] H. Senior, G. Slabaugh, S. Yuan, and L. Rossi, "Graph Neural Networks in Vision-Language Image Understanding: A Survey," *arXiv preprint arXiv:2303.03761*, Mar. 2023. Article (CrossRef Link).

[25] A. U. Khan et al., "Learning Situation Hyper-Graphs for Video Question Answering," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, May 2023. Article (CrossRef Link).

[26] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei, "Scene Graph Generation by Iterative Message Passing," in *Proc. of the IEEE conference on computer vision and pattern recognition*, 2017. Article (CrossRef Link).

[27] R. Li, S. Zhang, and X. He, "SGTR: End-to-end Scene Graph Generation with Transformer," in *Proc. of* the *IEEE/CVF conference on computer vision and pattern recognition*, Mar. 2022. Article (CrossRef Link).

[28] X. Dong, T. Gan, X. Song, J. Wu, Y. Cheng, and L. Nie, "Stacked Hybrid-Attention and Group Collaborative Learning for Unbiased Scene Graph Generation," in *Proc. of* the *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 19405–19414, Jun. 2022. Article (CrossRef Link).

[29] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh, "Graph R-CNN for Scene Graph Generation," in *Proc. of the European Conference on Computer Vision (ECCV)*, pp. 670–685, 2018. Article (CrossRef Link).

[30] J. Zhang, K. J. Shih, A. Elgammal, A. Tao, and B. Catanzaro, "Graphical Contrastive Losses for Scene Graph Parsing," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 11527–11535, Jun. 2019. Article (CrossRef Link).

[31] R. Li, S. Zhang, B. Wan, and X. He, "Bipartite Graph Network with Adaptive Message Passing for Unbiased Scene Graph Generation," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 11104–11114, Jun. 2021. Article (CrossRef Link).

[32] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness," *arXiv preprint arXiv:1811.12231*, Nov. 2022. Article (CrossRef Link).

[33] Y. Li, Y. Li, and N. Vasconcelos, "RESOUND: Towards Action Recognition Without Representation Bias," in *Proc. of the European Conference on Computer Vision (ECCV)*, pp. 520–535, 2018. Article (CrossRef Link).

[34] Y. Li and N. Vasconcelos, "REPAIR: Removing Representation Bias by Dataset Resampling," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 9564–9573, Jun. 2019. Article (CrossRef Link).

[35] H. He and E. A. Garcia, "Learning from Imbalanced Data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, 2009. Article (CrossRef Link).

[36] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, "Learning Fair Representations," in *Proc. of International conference on machine learning*, vol. 28, pp. 325–333, Jun. 2013.

[37] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection", in *Proc. of the IEEE international conference on computer vision*, pp. 2980-2988, 2017. Article (CrossRef Link).

[38] R. Cadene and C. Dancette, "RUBi: Reducing Unimodal Biases for Visual Question Answering," *Advances in neural information processing systems*, 2019.

[39] I. Misra, C. L. Zitnick, M. Mitchell, and R. Girshick, "Seeing through the Human Reporting Bias: Visual Classifiers from Noisy Human-Centric Labels," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 2930–2939, Jun. 2016. Article (CrossRef Link).

[40] B. Kang et al., "Decoupling Representation and Classifier for Long-Tailed Recognition," *arXiv preprint arXiv:1910.09217*, Feb. 2020. Article (CrossRef Link).

[41] T. Wang et al., "The Devil Is in Classification: A Simple Framework for Long-Tail Instance Segmentation," in *Proc. of Computer Vision--ECCV 2020: 16th European Conference*, pp. 728–744, 2020. Article (CrossRef Link).

[42] X. Wang, T. E. Huang, T. Darrell, J. E. Gonzalez, and F. Yu, "Frustratingly Simple Few-Shot Object Detection," *arXiv preprint arXiv:2003.06957*, Mar. 2020. Article (CrossRef Link).

[43] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017. Article (CrossRef Link).

[44] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125, Jul. 2017. Article (CrossRef Link).

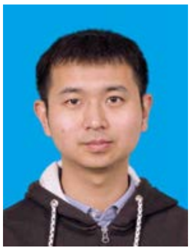
[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 770–778, Jun. 2016. Article (CrossRef Link).

[46] L. Chen, H. Zhang, J. Xiao, X. He, S. Pu, and S.-F. Chang, "Counterfactual Critic Multi-Agent Training for Scene Graph Generation," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, IEEE, pp. 4612–4622, Oct. 2019. Article (CrossRef Link).

Dongdong Jia received his B.S. degree from Xidian University in 2018, and is currently a master student at Yan'an University. His research interests include signal processing and Computer Vision, with a focus on Image Processing.

Meili Zhou received the M.S. degree in signal and information processing from the yanan university in 2008. She is an associate Professor with the School of physics and electronic information, yanan University. Her Interests cover signal processing, Computer Vision and Image Processing.

Wei Wei received the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 2005 and 2011, respectively. He is currently an Associate Professor with the School of Computer Science and Engineering, Xi'an University of Technology, Xi'an. His current research interests include the area of wireless networks, wireless sensor networks application, image processing, mobile computing. He has published around 100 research papers in international conferences and journals.



Dong Wang received the Ph.D. degree from the School of Computer Science, Northwestern Polytechnical University, Xi'an, China, in 2023.He is currently a Lecture with the School of Physics and Electronic Information, Yan'an University, Yan'an, China. His research interests include pansharpening, hyperspectral image superresolution, and few-shot learning.

Zongwen Bai is with the Shaanxi Provincial Key Lab of bigdata of energy and intelligence processing, School of physics and electronic information. He is currently pursuing the Ph. D. degree with the School of Computer Science, Northwestern Polytechnical University, Xi'an. His research interests cover Computer Vision, Nature Language Processing and Deep Learning.