Prototype-based Embedding Network for Scene Graph Generation

Chaofan Zheng* Xinyu Lyu* Lianli Gao† Bo Dai
Jingkuan Song
School of Computer Science and Engineering,
University of Electronic Science and Technology of China, China
zheng_chaofan@foxmail.com, lianli.gao@uestc.edu.cn

Abstract

Current Scene Graph Generation (SGG) methods explore contextual information to predict relationships among entity pairs. However, due to the diverse visual appearance of numerous possible subject-object combinations, there is a large ***intra-class variation*** *within each predicate category,* e.g., *"man-eating-pizza, giraffe-eating-leaf", and the severe* ***inter-class similarity*** *between different classes,* e.g., *"man-holding-plate, man-eating-pizza", in model's latent space. The above challenges prevent current SGG methods from acquiring robust features for reliable relation prediction. In this paper, we claim that the predicate's category-inherent semantics can serve as class-wise prototypes in the semantic space for relieving the challenges. To the end, we propose the* ***Prototype-based Embedding Network (PE-Net)****, which models entities/predicates with prototype-aligned compact and distinctive representations and thereby establishes matching between entity pairs and predicates in a common embedding space for relation recognition. Moreover,* ***Prototype-guided Learning (PL)*** *is introduced to help PE-Net efficiently learn such entity-predicate matching, and* ***Prototype Regularization (PR)*** *is devised to relieve the ambiguous entity-predicate matching caused by the predicate's semantic overlap. Extensive experiments demonstrate that our method gains superior relation recognition capability on SGG, achieving new state-of-the-art performances on both Visual Genome and Open Images datasets. The codes are available at* https://github.com/VL-Group/PENET.

1. Introduction

Scene Graph Generation (SGG) is a fundamental computer vision task that involves detecting the entities and predicting their relationships in an image to generate a scene graph, where nodes indicate entities and edges indicate relationships between entity pairs. Such a graph-structured representation is helpful for downstream tasks such as Visual Question Answering [5, 15, 41], Image Captioning [4, 38, 42, 45], and Image Retrieval [10, 25, 40].

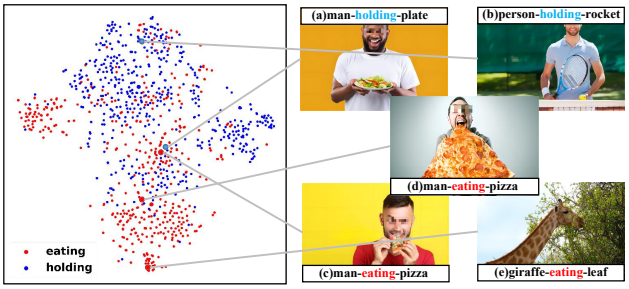


Figure 1. The illustration of relation representations with large intra-class variation and severe inter-class similarity. **Left:** the feature distribution of "eating" (in red) and "holding" (in blue) obtained by Motifs [39]. **Right:** some instances of "eating" and "holding". Examples (c) and (e) illustrate that relation instances from the same class have diverse appearance. Moreover, examples (a) and (c) demonstrate that similar-looking relation instances may belong to different categories.

Existing SGG models [1, 3, 7, 13, 18, 29, 39] typically start with an object detector that generates a set of entity proposals and corresponding features. Then, entity features are enhanced by exploring the contextual information taking advantage of message-passing modules. Finally, these refined entity features are used to predict pairwise relations. Although many works have made great efforts to explore the contextual information for robust relation recognition, they still suffer from biased-prediction problems, preferring common predicates (*e.g.*,"on", "of") instead of fine-grained ones (*e.g.*,"walking on", "covering"). To address the problem, various de-biasing frameworks [6, 14, 21, 28, 34, 37, 46, 47] have been proposed to obtain balanced prediction results. While alleviating the long-tailed issue to some extent, most of them only achieve a trade-off between head and tail predicates. In other words, they sacrifice the robust representations learned on head

*Equal contribution.
†Corresponding author.

predicates for unworthy improvements in the tail ones [46], which do not truly improve the model's holistic recognition ability for most of the relations.

The origin of the issue lies in the fact that current SGG methods fail to capture compact and distinctive representations for relations. For instance, as shown in Fig. 1, the relation representation, derived from Motifs' latent space, is heavily discrete and intersecting. Hence, it makes existing SGG models hard to learn perfect decision boundaries for accurate predicate recognition. Accordingly, we summarize the issue as two challenges: large **Intra-class variation** within the same relation class and severe **Inter-class similarity** between different categories.

Intra-class variation. The intra-class variation arises from the diverse appearance of entities and various subject-object combinations. Specifically, entities' visual appearances change greatly even though they belong to the same class. Thus, represented as the union feature containing subject and object entities, relation representations significantly vary with the appearances of entity instances, *e.g.*, various visual representations for "pizza" in Fig. 1(c) *vs.* Fig. 1(d). Besides, the numerous subject-object combinations of predicate instances further increase the variation within each predicate class, *e.g.*, "man-eating-pizza" *vs.* "giraffe-eating-leaf" in Fig. 1(c) and Fig. 1(e).

Inter-class similarity. The inter-class similarity of relations originates from similar-looking interactions but belongs to different predicate classes. For instance, as shown in Fig. 1(a) and Fig. 1(c), the similar visual appearance of interactions between "man-pizza" and "man-plate" make current SGG models hard to distinguish "eating" from "holding", even if they are semantic irrelevant to each other.

The above challenges motivate us to study two problems: 1) For the intra-class variation, how to capture category-inherent features, producing compact representations for entity/predicate instances from the same category. Moreover, 2) for the inter-class similarity, how to derive distinctive representations for effectively distinguishing similar-looking relation instances between different classes. Our key intuition is that semantics is more reliable than visual appearance when modeling entities/predicates. Intuitively, although entities/predicates of the same class significantly vary in visual appearance, they all share the representative semantics, which can be easily captured from their class labels. Dominated by the representative semantics, the representations of entities and predicates have smaller variations within their classes in the semantic space. Besides, the class-inherent semantics is discriminative enough for visual-similar instances between different categories. Therefore, in conjunction with the above analysis, modeling entities and predicates in the semantic space can provide highly compact and distinguishable representations against intra-class variation and inter-class similarity challenges.

Inspired by that, we propose a simple but effective method, Prototype-based Embedding Network (PE-Net), which produces compact and distinctive entity/predicate representations for relation recognition. To achieve that, the PE-Net models entity and predicate instances with compact and distinguishable representations in the semantic space, which are closely aligned to their semantic prototypes. Practically, the prototype is defined as the representative embedding for a group of instances from the same entity/predicate class. Then, the PE-Net establishes matching between entity pairs (*i.e.*, subject-object $(\boldsymbol{s}, \boldsymbol{o})$) and their corresponding predicates $(\boldsymbol{p})$ for relation recognition (*i.e.*, $\mathcal{F}(\boldsymbol{s}, \boldsymbol{o}) \approx \boldsymbol{p}$). Besides, a Prototype-guided Learning strategy (PL) is proposed to help PE-Net efficiently learn this entity-predicate matching. Additionally, to alleviate the ambiguous entity-predicate matching caused by the semantic overlap between predicates (*e.g.*, "walking on" and "standing on"), Prototype Regularization (PR) is proposed to encourage inter-class separation between predicate prototypes for precise entity-predicate matching. Finally, we introduce two metrics, *i.e.*, Intra-class Variance (IV) and Intra-class to Inter-class Variance Ratio (IIVR), to measure the compactness and distinctiveness of entity/predicate representations, respectively.

In summary, the main contributions of our work are three folds:

- We propose a simple yet effective method, *i.e.*, Prototype-based Embedding Network (PE-Net), which produces compact and distinctive entity/predicate representations and then establishes matching between entity pairs and predicates for relation recognition.
- Moreover, Prototype-guided Learning (PL) is introduced to help PE-Net efficiently learn such entity-predicate matching, and Prototype Regularization (PR) is devised to relieve the ambiguous entity-predicate matching caused by the predicate's semantic overlap.
- Evaluated on the Visual Genome and Open Images datasets, our method significantly increases the relation recognition ability for SGG, achieving new state-of-the-art performances.

2. Related Work

We categorize the related works of SGG into the following fields: Vanilla Scene Graph Generation Model and Unbiased Scene Graph Generation Framework.

Vanilla Scene Graph Generation Model. Numerous models have been proposed to solve the scene graph generation task from different perspectives in recent years. Early methods [20] attempted to detect objects and relations with independent networks, ignoring the rich contextual information.

Afterward, [33] firstly proves that the contextual information can significantly improve the relation prediction and hence introduces an iterative message-passing mechanism to refine the features of objects and relations. [39] further emphasizes the importance of contextual information between objects and utilizes the BiLSTM to encode the object and edge contextual information. Moreover, to avoid suffering from noisy information during message passing, [29] and [36] design sparse structures to improve the model's context modeling capability. In addition, prior knowledge is also helpful for relation prediction. [39] explores the statistical patterns of object co-occurrence for refining relation predictions. Besides, [7] encodes the commonsense knowledge into the model to improve the few-shot recognition ability. However, due to the imbalanced data distribution, the vanilla SGG models struggle to recognize the fine-grained tail predicates.

Unbiased Scene Graph Generation Framework. Recently, various de-biasing SGG frameworks have been proposed to tackle the biased predictions problem. [28] proposes a counterfactual causality method to remove the effect of context bias. [37] constructs a hierarchical tree structure from the cognitive perspective to make the tail predicates receive more attention. [13] compensates the disadvantages of over-sampling and under-sampling and proposes a bi-level sampling method. [6] creates a balanced learning process by constructing a balanced predicate learning space and semantic adjustment. [11] explicitly cleans the noisy annotations on the datasets to balance the data distribution. [21] introduces a predicate lattice to figure out the fine-grained predicate pairs that are hard to distinguish. Despite alleviating the biased problem to some extent, these methods improve the prediction performance of tail predicates at the expense of head ones, which do not truly improve the model's holistic recognition ability.

Our work generates compact and distinctive entity/predicate representations by utilizing a prototype-based modeling method and cleverly-designed learning strategies, which achieves superior relation recognition performance on both head predicates and tail ones with a simple but effective framework.

3. Method

The whole pipeline of our Prototype-based Relation Embedding (PE-Net) is illustrated in Fig. 2. Following the previous works [27, 28], we utilize an object detector (*e.g.*, Faster R-CNN [24]) to generate a set of entity proposals with corresponding features. Moreover, the features extracted from the union box between two entities are used to represent their corresponding predicates. Given entity and predicate features, the Prototype-based Embedding Network (PE-Net) models subject ($\boldsymbol{s}$), object ($\boldsymbol{o}$), and predicate ($\boldsymbol{p}$) instances with prototype-based compact and distinguishable representations. Then, the PE-Net matches subject-object pairs ($(\boldsymbol{s}, \boldsymbol{o})$) with the corresponding predicates (*i.e.*, $\mathcal{F}(\boldsymbol{s}, \boldsymbol{o}) \approx \boldsymbol{p}$) in the common embedding space for relation recognition. To achieve that, we propose a Prototype-guided Learning (PL), to help PE-Net learn the entity-predicate matching. Furthermore, to relieve the ambiguous matching problem caused by the predicate's semantic overlap, Prototype Regularization (PR) is proposed to encourage inter-class distinction for accurate entity-predicate matching.

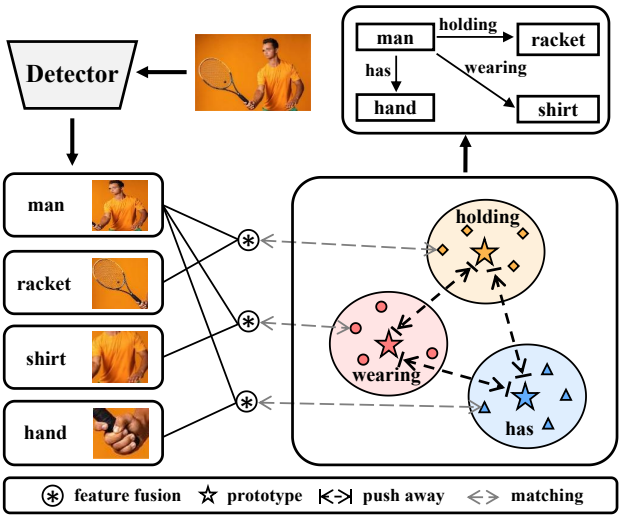


Figure 2. The main process of our proposed PE-Net from the input image to the scene graph.

3.1. Prototype-based Embedding Network

The procedure of Prototype-based Embedding Network (PE-Net) can be divided into two steps: 1) Prototype-based Modeling for producing compact and distinctive entity and predicate representations. 2) Prototype-guided Entity-Predicate Matching for relation recognition.

Prototype-based Modeling. The Prototype-based Embedding Network (PE-Net) models entity/predicate instances with prototype-based compact and discriminative representations shown in Fig. 3.

Concretely, the representations of subject ($\boldsymbol{s}$), object ($\boldsymbol{o}$), and predicate ($\boldsymbol{p}$) are modeled below:

$$\begin{aligned} \boldsymbol{s} &= \mathbf{W}_s \boldsymbol{t_s} + \boldsymbol{v_s}, \\ \boldsymbol{o} &= \mathbf{W}_o \boldsymbol{t_o} + \boldsymbol{v_o}, \\ \boldsymbol{p} &= \mathbf{W}_p \boldsymbol{t_p} + \boldsymbol{u_p}, \end{aligned} \tag{1}$$

where $\mathbf{W_s}$, $\mathbf{W_o}$, and $\mathbf{W_p}$ are learnable parameters. Moreover, $\mathbf{W}_s\boldsymbol{t_s}$, $\mathbf{W}_o\boldsymbol{t_o}$ and $\mathbf{W}_p\boldsymbol{t_p}$ are the class-specific semantic prototypes obtained from their class labels' word embedding (GloVe [23]), *i.e.*, $\boldsymbol{t_s}$, $\boldsymbol{t_o}$ and $\boldsymbol{t_p}$. Based on the class-specific prototypes, the instance-varied semantic contents $\boldsymbol{v_s}$, $\boldsymbol{v_o}$ and $\boldsymbol{u_p}$ are utilized to model the diversity of each

instance from the same subject, object, and predicate class. Practically, $\boldsymbol{v_s}$ are obtained as:

$$\begin{aligned} \boldsymbol{g_s} &= \sigma\left(f((\mathbf{W_s}\boldsymbol{t_s}) \oplus h(\boldsymbol{x_s}))\right) \\ \boldsymbol{v_s} &= \boldsymbol{g_s} \odot h(\boldsymbol{x_s}), \end{aligned} \tag{2}$$

where $f(\cdot)$ is a fully connected layer, $h(\cdot)$ is the visual-to-semantic function used to transform the visual feature into semantic space, and $\oplus$ is the concatenation operation. Moreover, $\sigma(\cdot)$ is the sigmoid activation function, $\odot$ is the element-wise product, and $\boldsymbol{x_s}$ is the visual features of subject instances from the detector. Utilizing the gate mechanism in Eq. (2), the class-irrelevant information is eliminated from the original visual feature $\boldsymbol{x_s}$ producing consistent representations within class. In addition, we derive $\boldsymbol{v_o}$ in the same way following Eq. (2).

Similarly, predicate's instance-varied semantic content $\boldsymbol{u_p}$ is defined as:

$$\begin{aligned} \boldsymbol{g_p} &= \sigma\left(f\left(\mathcal{F}(\boldsymbol{s},\boldsymbol{o}) \oplus h(\boldsymbol{x_u})\right)\right) \\ \boldsymbol{u_p} &= \boldsymbol{g_p} \odot h(\boldsymbol{x_u}), \end{aligned} \tag{3}$$

where $\boldsymbol{x_u}$ is the union feature of subject and object, and $\mathcal{F}(\cdot,\cdot)$ denotes the feature fusion function.

Prototype-guided Entity-Predicate Matching. Then, for relation recognition, we match subject instance ($\boldsymbol{s}$) and object instance ($\boldsymbol{o}$) with the corresponding predicate instance ($\boldsymbol{p}$) in the common semantic space. Practically, the entity-predicate matching is shown below:

$$\mathcal{F}(\boldsymbol{s},\boldsymbol{o}) \approx \boldsymbol{p}, \tag{4}$$

where $\mathcal{F}(\boldsymbol{s},\boldsymbol{o})$ is defined as: $\text{ReLU}(\boldsymbol{s}+\boldsymbol{o}) - (\boldsymbol{s}-\boldsymbol{o})^2$.

However, the predicate representation varies with the subject-object pair, which prevents PE-Net from efficiently learning the matching. Therefore, we perform an equivalent transformation on Eq. (4), deriving a deterministic matching objective as follows:

$$\mathcal{F}(\boldsymbol{s},\boldsymbol{o}) - \boldsymbol{u_p} \approx \mathbf{W}_p\boldsymbol{t_p}, \tag{5}$$

where $\mathcal{F}(\boldsymbol{s},\boldsymbol{o}) - \boldsymbol{u_p}$ is defined as relation representation $\boldsymbol{r}$, which should be matched to its corresponding predicate prototype $\mathbf{W}_p\boldsymbol{t_p}$ (represented as $\boldsymbol{c}$ in the following sections).

3.2. Prototype-guided Learning

To help PE-Net efficiently match relation representations with corresponding predicates in Eq. (5), we devise a learning strategy, *i.e.*, Prototype-guided Learning (PL), which makes relation representations close to their corresponding prototypes. In practice, PL consists of two constraints: cosine similarity and Euclidean distance.

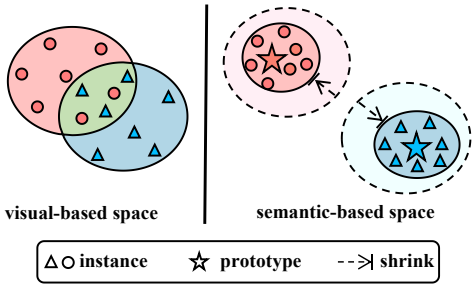


Figure 3. The illustration of Prototype-based Modeling *vs*. Visual-based Modeling. **Left:** Modeling instances with the appearance in visual-based space suffers from large intra-class variation and severe inter-class overlap. **Right:** Our Prototype-based Modeling method gathers instances around their semantic prototypes in the semantic-based space producing compact and distinctive representations.

Firstly, we have to increase the cosine similarity between the relation representation $\boldsymbol{r}$ and its corresponding prototype $\boldsymbol{c_t}$, which is implemented as the following loss function:

$$\mathcal{L}_{e\_sim} = -\log \frac{\exp(\langle \overline{\boldsymbol{r}}, \overline{\boldsymbol{c_t}} \rangle / \tau)}{\sum_{j=0}^{N} \exp(\langle \overline{\boldsymbol{r}}, \overline{\boldsymbol{c_j}} \rangle / \tau)}, \tag{6}$$

where $\bar{\cdot}$ denotes the unitary operation, τ is a learnable temperature hyper-parameter, t is subscript for the ground truth class, and N is the number of predicate categories.

Since the cosine similarity only considers angle-based relative distance, it may fail to make relation representations and corresponding prototypes close to each other in the Euclidean space. To the end, we further impose the Euclidean distance constraint. It encourages the relation representation $\boldsymbol{r}$ close to its corresponding prototypes $\boldsymbol{c_t}$ while keeping the distances with others in Euclidean space. Practically, we first calculate the distances between the relation representation $\boldsymbol{r}$ and each class prototype $\boldsymbol{c_i}$ obtaining the distances set $\boldsymbol{G} = \{g_i\}_{i=0}^{N}$, where g_i is computed as:

$$g_i = \|\boldsymbol{r} - \boldsymbol{c_i}\|_2^2. \tag{7}$$

Then, we sort the distance set $\boldsymbol{B} = \boldsymbol{G} \setminus \{g_t\}$ (excluding g_t) in increasing order and obtain the sorted distance set as $\boldsymbol{B'} = \{b'_i\}_{i=0}^{N-1}$. Furthermore, the top k_1 smallest distances of $\boldsymbol{B'}$ are averaged as the distance g^- to negative prototypes:

$$g^- = \frac{1}{k_1} \sum_{i=0}^{k_1-1} b'_i. \tag{8}$$

Together with the distance to the positive prototype $g^+ = g_t$, we further construct the triplet loss:

$$\mathcal{L}_{e\_euc} = \max(0, g^+ - g^- + \gamma_1), \tag{9}$$

where γ_1 is a hyper-parameter to adjust the distance margins between relation representations and the negative prototypes.

3.3. Prototype Regularization

To alleviate the ambiguous matching caused by the semantic overlap between predicates, we propose a Prototype Regularization (PR) to encourage inter-class separation by enlarging the distinction between prototypes for precise entity-predicate matching. Correspondingly, according to the constraints imposed in Sec. 3.2, we first calculate the cosine similarity between predicate prototypes obtaining the similarity matrix as follows:

$$\boldsymbol{S} = \overline{\boldsymbol{C}} \cdot \overline{\boldsymbol{C}}^{\top} = (s_{ij}) \in \mathbb{R}^{(N+1)\times(N+1)}, \tag{10}$$

where $\boldsymbol{C} = [\boldsymbol{c_0}; \boldsymbol{c_1}; ...; \boldsymbol{c_N}]$ is the predicate prototype matrix, and $\overline{\boldsymbol{C}}$ is obtained by normalizing the vectors in it. Moreover, s_{ij} represents the cosine similarity between prototype $\boldsymbol{c_i}$ and $\boldsymbol{c_j}$. Then, we should reduce each pair of prototypes' cosine similarity to make them distinctive in the semantic space. Therefore, we introduce the $l_{2,1}$-norm of $\boldsymbol{S}$ and minimize it:

$$\mathcal{L}_{r\_sim} = \|\boldsymbol{S}\|_{2,1} = \sum_{i=0}^{N} \sqrt{\sum_{j=0}^{N} s_{ij}^2}. \tag{11}$$

However, only regularized by the cosine similarity, some predicates are still not distinctive enough against others. Thus, we enlarge their distances in the Euclidean space for further distinction. To achieve that, we calculate the Euclidean distance between two prototypes obtaining the distance matrix $\boldsymbol{D} = (d_{ij}) \in \mathbb{R}^{(N+1)\times(N+1)}$ with d_{ij} computed as:

$$d_{ij} = \|\boldsymbol{c_i} - \boldsymbol{c_j}\|_2^2, \tag{12}$$

where d_{ij} indicates the Euclidean distance between prototype $\boldsymbol{c_i}$ and $\boldsymbol{c_j}$. For each prototype, we should distance it from others. Therefore, we sort the elements in each row of matrix $\boldsymbol{D}$ in increasing order, obtaining $\boldsymbol{D'} = (d'_{ij}) \in \mathbb{R}^{(N+1)\times(N+1)}$, and select the top k_2 smallest values of each row to widen them:

$$\begin{aligned} d^{-} &= \frac{1}{(N+1)k_2} \sum_{i=0}^{N} \sum_{j=1}^{k_2} d'_{ij}, \\ \mathcal{L}_{r\_euc} &= \max(0, -d^{-} + \gamma_2), \end{aligned} \tag{13}$$

where γ_2 is another hyper-parameter used to adjust the distance margins.

3.4. Scene Graph Prediction

During the training stage, the final loss function $\mathcal{L}$ for our PE-Net is expressed as:

$$\mathcal{L} = \mathcal{L}_{r\_sim} + \mathcal{L}_{e\_sim} + \mathcal{L}_{r\_euc} + \mathcal{L}_{e\_euc}. \tag{14}$$

During the testing stage, with the relation representation $\boldsymbol{r}$, we choose the class of prototypes with the highest cosine similarity as the prediction result:

$$res_r = \arg\max_i(\{q_i | q_i = \langle \overline{\boldsymbol{r}}, \overline{\boldsymbol{c_i}} \rangle / \tau\}), \tag{15}$$

where q_i indicates the similarity between relation representation $\boldsymbol{r}$ and prototype $\boldsymbol{c_i}$.

4. Experiments

4.1. Datasets

Visual Genome (VG). The Visual Genome (VG) dataset consists of 108,077 images with average annotations of 38 objects and 22 relationships per image. In this paper, we adopt the most widely used split [33], which contains the most frequent 150 object categories and 50 predicate categories. Specifically, the dataset is divided into a training set with 70% of the images, a testing set with the remaining 30%, and 5k images from the training set for validation.

Open Images (OI). We conduct experiments on Open Image V6 dataset, which has 126,368 images for training, and 1813 and 5322 images for validation and testing. It contains 301 object categories and 31 predicate categories.

4.2. Evaluation Protocol

Visual Genome (VG). We evaluate our method on three sub-tasks, including Predicate Classification (**PredCls**), Scene Graph Classification (**SGCls**), and Scene Graph Detection (**SGDet**). Following the recent works [21, 27–29], we take Recall@K (**R@K**) and mean Recall@K (**mR@K**) as the primary evaluation metrics. Moreover, we also report the zero-shot Recall@K (**zs-R@K**) that measures the model's generalization in dealing with the unseen relation triplets during training. Due to the imbalanced data distribution of VG dataset, R@K focuses on the common predicates with abundant samples, and mR@K prefers the tail predicates. Therefore, we introduce the Mean@K (**M@K**), which averages the R@K and mR@K for evaluating the model's overall performance on SGG. In addition, the Intra-class Variance (**IV**) and Intra-class to Inter-class Variance Ratio (**IIVR**) are introduced to measure the compactness and distinctiveness of entity/predicate representations. Intuitively, lower values of **IV** and **IIVR** indicate higher quality for representations.

Open Images (OI). Following the previous works [13, 19, 44], we utilize the Recall@50 (**R@50**), weighted mean AP of relations (**wmAP**$_{rel}$), weighted mean AP of phrase (**wmAP**$_{phr}$) as the evaluation metrics. The **score**$_{wtd}$ is calculated as: score$_{wtd}$ = 0.2 × R@50 + 0.4 × wmAP$_{rel}$ + 0.4 × wmAP$_{phr}$.

Model	PredCls			SGCls			SGDet		
	R@50/100	mR@50/100	M@50/100	R@50/100	mR@50/100	M@50/100	R@50/100	mR@50/100	M@50/100
SGTR [12]	-	-	-	-	-	-	24.6 / 28.4	12.0 / 15.2	18.3 / 21.8
SS R-CNN [30]	-	-	-	-	-	-	33.5 / 38.4	8.6 / 10.3	21.1 / 24.4
Motifs$^{\diamond}$ [27, 39]	65.3 / 67.2	14.9 / 16.3	40.1 / 41.8	38.9 / 39.8	8.3 / 8.8	23.6 / 24.3	32.1 / 36.8	6.6 / 7.9	19.4 / 22.4
VCTree$^{\diamond}$ [27, 29]	65.5 / 67.4	16.7 / 17.9	41.1 / 42.7	40.3 / 41.6	7.9 / 8.3	24.1 / 25.0	31.9 / 36.0	6.4 / 7.3	19.2 / 21.7
G R-CNN* [13, 36]	65.4 / 67.2	16.4 / 17.2	40.9 / 42.2	37.0 / 38.5	9.0 / 9.5	23.0 / 24.0	29.7 / 32.8	5.8 / 6.6	17.8 / 19.7
KERN* [1, 13]	65.8 / 67.6	17.7 / 19.2	41.8 / 43.4	36.7 / 37.4	9.4 / 10.0	23.1 / 23.7	27.1 / 29.8	6.4 / 7.3	16.8 / 18.6
VTransE$^{\diamond}$ [27, 43]	65.7 / 67.6	14.7 / 15.8	40.2 / 41.7	38.6 / 39.4	8.2 / 8.7	23.4 / 24.1	29.7 / 34.3	5.0 / 6.1	17.4 / 20.2
R-CAGCN [35]	66.6 / 68.3	18.3 / 19.9	42.5 / 44.1	38.3 / 39.0	10.2 / 11.1	24.3 / 25.1	28.1 / 31.3	7.9 / 8.8	18.0 / 20.1
GPS-Net* [13, 17]	65.2 / 67.1	15.2 / 16.6	40.2 / 41.9	37.8 / 39.2	8.5 / 9.1	23.2 / 24.2	31.3 / 35.9	6.7 / 8.6	19.0 / 22.3
RU-Net [19]	67.7 / 69.6	- / 24.2	- / 46.9	**42.4 / 43.3**	- / 14.6	- / 29.0	**32.9 / 37.5**	- / 10.8	- / 24.2
BGNN* [13]	59.2 / 61.3	30.4 / 32.9	44.8 / 47.1	37.4 / 38.5	14.3 / 16.5	25.9 / 27.5	31.0 / 35.8	10.7 / 12.6	20.9 / 24.2
PE-Net(P)	**68.2 / 70.1**	23.1 / 25.4	45.7 / 47.8	41.3 / 42.3	13.1 / 14.8	27.2 / 28.6	32.4 / 36.9	8.9 / 11.0	20.7 / 24.0
PE-Net	64.9 / 67.2	**31.5 / 33.8**	**48.2 / 50.5**	39.4 / 40.7	**17.8 / 18.9**	**28.6 / 29.8**	30.7 / 35.2	**12.4 / 14.5**	**21.6 / 24.9**
Motifs-TDE [28]	46.2 / 51.4	25.5 / 29.1	35.9 / 40.3	27.7 / 29.9	13.1 / 14.9	20.4 / 22.4	16.9 / 20.3	8.2 / 9.8	12.6 / 15.1
Motifs-CogTree [37]	35.6 / 36.8	26.4 / 29.0	31.0 / 32.9	21.6 / 22.2	14.9 / 16.1	18.3 / 19.2	20.0 / 22.1	10.4 / 11.8	15.2 / 17.0
Motifs-BPL-SA [6]	50.7 / 52.5	29.7 / 31.7	40.2 / 42.1	30.1 / 31.0	16.5 / 17.5	23.3 / 24.3	23.0 / 26.9	13.5 / 15.6	18.3 / 21.3
Motifs-NICE [11]	55.1 / 57.2	29.9 / 32.3	42.5 / 44.8	33.1 / 34.0	16.6 / 17.9	24.9 / 26.0	**27.8 / 31.8**	12.2 / 14.4	20.0 / 23.1
Motifs-PPDL [14]	47.2 / 47.6	32.2 / 33.3	39.7 / 40.5	28.4 / 29.3	17.5 / 18.2	23.0 / 23.8	21.2 / 23.9	11.4 / 13.5	16.3 / 18.7
Motifs-GCL [3]	42.7 / 44.4	36.1 / 38.2	39.4 / 41.3	26.1 / 27.1	20.8 / 21.8	23.5 / 24.5	18.4 / 22.0	**16.8 / 19.3**	17.6 / 20.7
Motifs-Reweight [2]	53.2 / 55.5	33.7 / 36.1	43.5 / 45.8	32.1 / 33.4	17.7 / 19.1	24.9 / 26.3	25.1 / 28.2	13.3 / 15.4	19.2 / 21.8
PE-Net-Reweight	**59.0 / 61.4**	**38.8 / 40.7**	**48.9 / 51.1**	**36.1 / 37.3**	**22.2 / 23.5**	**29.2 / 30.4**	26.5 / 30.9	16.7 / 18.8	**21.6 / 24.9**

Table 1. Performance comparison with the state-of-the-art SGG methods on VG dataset. * and $^{\diamond}$ denotes the results reproduced with the codebase provided by [13] and [27]. **PE-Net(P)** refers to the PE-Net only trained with PL. **PE-Net** indicates PE-Net trained with both PL and PR. The **best** and second best methods under each setting are marked according to formats.

4.3. Implementation Details

Following the previous works [18, 27, 28, 35], we adopt the Faster R-CNN [24] with ResNeXt-101-FPN [9, 16, 32] provided by [27] to detect objects in the image. The parameters of the detector are frozen during training. In particular, we set k_1, k_2, γ_1, and γ_2 as 10, 1, 1, and 7. Additionally, the PE-Net is trained by an SGD optimizer with $60k$ iterations. The initial learning rate and the batch size are set to 10^{-3} and 8. All experiments are implemented with PyTorch and trained with an NVIDIA GeForce RTX 3090 GPU.

4.4. Comparisons with State-of-the-art Methods

Visual Genome. To evaluate PE-Net's capability on scene graph generation, we compare it with several state-of-the-art SGG methods on Visual Genome dataset under all three sub-tasks. The results are shown in Tab. 1. Generally, our method achieves superior performance compared to other SGG models. Concretely, PE-Net(P) outperforms the VCTree by 7.5%, 6.5%, and 3.7% at mR@100 and by 2.7%, 0.7%, and 0.9% at R@100 on PredCls, SGCls, and SGDet. It also outperforms the recent SGG model, RU-Net, by 0.5% and 1.2% at R@100 and mR@100 on PredCls. In addition, the full PE-Net outperforms VCTree by 15.9%, 10.6%, 7.2%, and RU-Net by 9.6%, 4.3%, 3.7%, at mR@100 on three subtasks, respectively. The results demonstrate the effectiveness of our model.

Moreover, to explore PE-Net's potential capability of solving the long-tail problem for SGG, we equip it with the advanced re-weighting method [2]. Then, we compare PE-Net-Reweight with Motifs [39] de-biased by several existing state-of-the-art de-biasing methods. The results are summarized in Tab. 1. We find that our PE-Net-Reweight pushes the performance on unbiased SGG to a new level. For instance, compared with Motifs-GCL, we achieve an absolute performance advantage, outperforming it by 17.0%, 10.2%, and 8.9% at R@100 on PredCls, SGCls, and SGDet tasks, and by 2.5%, 1.7% at mR@100 on PredCls, SGCls tasks, respectively. Benefiting from the prototype-aligned distinctive representations, the PE-Net has the potential to tackle the biased problem in SGG.

In addition, we report the zero-shot recall results to verify the generalization of our method in handling the unseen relation triplets in the training set. As shown in Tab. 2, our model outperforms the vanilla Motifs and VCTree by 15.53%, 5.40%, 3.49%, and 15.38%, 4.45%, 2.91% at zs-R@100 on PredCls, SGCls, and SGDet. Although TDE significantly improves the zero-shot performance by removing the effect of context bias, Motifs-TDE and VCTree-TDE are still surpassed by our PE-Net with 2.69%, 2.03%, 0.7%, and 3.29%, 2.53%, 0.4% on three tasks, respectively. We owe the strength to the Prototype-based Modeling of our PE-Net, which models entity and predicate in the semantic space, significantly improving the model's analogical reasoning capability on unseen relation triplets.

Open Images. To verify the generality of our method on different datasets, we conduct experiments on Open Images and present the results in Tab. 3. Consistent with the performance on VG, PE-Net also achieves competitive results on

Models	PredCls	SGCls	SGDet
	zs-R@50 / 100	zs-R@50 / 100	zs-R@50 / 100
Motifs [39]	3.24 / 5.36	0.68 / 1.13	0.05 / 0.11
VCTree [29]	3.27 / 5.51	1.17 / 2.08	0.31 / 0.69
Motifs-TDE [28]	14.4 / 18.2	3.4 / 4.5	2.3 / 2.9
VCTree-TDE [28]	14.3 / 17.6	3.2 / 4.0	**2.6** / 3.2
Motifs-EBM [26]	4.87 / -	1.25 / -	0.23 / -
VCTree-EBM [26]	5.36 / -	1.87 / -	0.54 / -
PE-Net	**17.16 / 20.89**	**5.37 / 6.53**	2.31 / **3.60**

Table 2. Comparison with different methods on Zero-shot Recall (zs-R@50/100) under all three sub-tasks on the VG dataset.

Model	R@50	wmAP_{rel}	wmAP_{phr}	score_{wtd}
Motifs [39]	71.6	29.9	31.6	38.9
G R-CNN [36]	74.5	33.2	34.2	41.8
GPS-Net [17]	74.8	32.9	34.0	41.7
VCTree [29]	74.1	34.2	33.1	40.2
BGNN [13]	75.0	33.5	34.2	42.1
RU-Net [19]	**76.9**	<u>35.4</u>	<u>34.9</u>	<u>43.5</u>
PE-Net	<u>76.5</u>	**36.6**	**37.4**	**44.9**

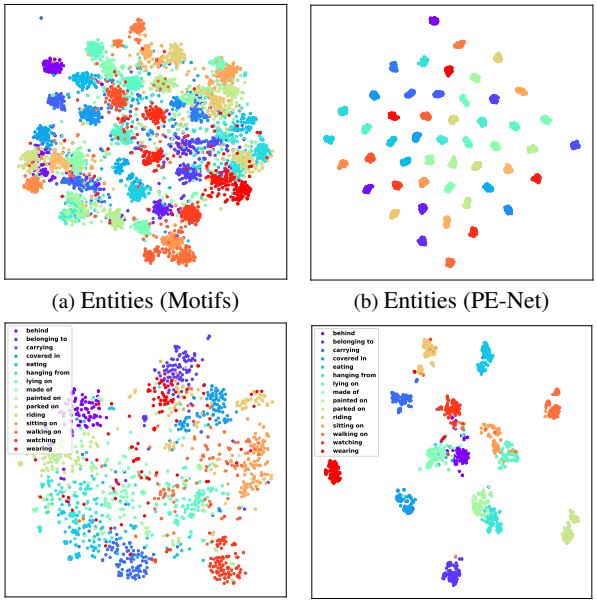
Table 3. Comparison with the state-of-the-art methods on Open-Images V6. We adopt the same evaluation metric as in [13]. The **best** and <u>second best</u> methods under each setting are marked according to formats.

Open Images dataset. Specifically, our method exceeds the BGNN with a large margin of 2.7% on average at four metrics, and outperforms RU-Net by 1.2%, 2.5%, and 1.4% at wmAP_{rel}, wmAP_{phr}, and score_{wtd}, respectively. It powerfully confirms PE-Net's generalization on handling relation recognition under different data distributions.

4.5. Measuring Representation Modeling of PE-Net

To certify the assumption that our PE-Net is capable of producing compact and distinctive representations for entity and predicate, we conduct both quantitative and qualitative studies in Tab. 5 and Fig. 4, respectively. Notably, we only conduct experiments on the PredCls task, which eliminates the impact of entities' mis-classification made by detectors.

Quantitative Analysis. To quantitatively evaluate the quality of the entity's and predicate's representations (*i.e.*, degree of intra-class compactness and the inter-class distinctiveness), we evaluate and make comparisons between PE-Net and previous methods [17, 27, 29, 36, 39] with IV (Intra-class Variance) and IIVR (Intra-class to Inter-class Variance Ratio). Moreover, the experimental results are shown in Tab. 5. Firstly, our PE-Net yields more compact entity and predicate representations than previous methods, *e.g.*, 0.74 *vs*. 9.73 on IV-O and 1.06 *vs*. 1.41 on IV-R compared with Motifs. That illustrates the effectiveness of our Prototype-based Modeling in PE-Net. Also, the representations learned by our model are more distinguishable, *e.g.*, 0.24 *vs*. 1.93 on IIVR-O and 1.67 *vs*. 2.72 on IIVR-R compared with Motifs. We owe it to the effectiveness of our PR, which significantly alleviates the ambiguous



(a) Entities (Motifs) (b) Entities (PE-Net)

(c) Relations (Motifs) (d) Relations (PE-Net)

Figure 4. The comparison of t-SNE visualization results on entity and predicate feature distributions between PE-Net and Motifs under the VG dataset.

entity-predicate matching by encouraging predicate prototypes away from each other.

Qualitative Analysis. For an intuitive illustration of PE-Net's capability of yielding compact and distinguishable representations, we visualize the feature distribution of entities and predicates taking advantage of the t-SNE technique, shown in Fig. 4. Comparing Fig. 4(a) with Fig. 4(b), we observe that PE-Net produces more compact and distinctive entity representations than Motifs, intuitively illustrating the advantages of modeling instances in the semantic space than from visual appearances. In addition, the relation feature distribution of Motifs shown in Fig. 4(c) is of large intra-class variance and severe inter-class overlap. In this case, it is hard for SGG models to learn a perfect decision boundary for accurate relation recognition. On the contrary, the relation representations learned by our PE-Net are of high-level inter-class distinctiveness and intra-class compactness, which intuitively demonstrates our method's superiority and explains why our method achieves excellent relation prediction performance.

4.6. Ablation Studies

To verify the effectiveness of each component of the proposed PE-Net, we conduct ablation studies on PL and PR under the VG dataset, and the results are summarized in Tab. 4. Exp 1, PE-Net is trained without PL and PR, which directly uses a linear classifier to classify the relation representation defined in Eq. (5). Exp 2, PE-Net is trained with

Exp	Component		PredCls			SGCls			SGDet		
	PL	PR	R@50/100	mR@50/100	M@50/100	R@50/100	mR@50/100	M@50/100	R@50/100	mR@50/100	M@50/100
1	✗	✗	66.5 / 68.2	18.5 / 20.0	42.5 / 44.1	39.5 / 40.4	9.9 / 10.5	24.7 / 25.5	32.3 / 36.8	7.8 / 9.3	20.1 / 23.1
2	✓	✗	**68.2 / 70.1**	23.1 / 25.4	45.7 / 47.8	**41.3 / 42.3**	13.1 / 14.8	27.2/ 28.6	**32.4 / 36.9**	8.9 / 11.0	20.7 / 24.0
3	✓	✓	64.9 / 67.2	**31.5 / 33.8**	**48.2 / 50.5**	39.4 / 40.7	**17.8 / 18.9**	**28.6 / 29.8**	30.7 / 35.2	**12.4 / 14.5**	**21.6 / 24.9**

Table 4. Ablation study on each component of PE-Net. PL and PR denote the Prototype-guided Learning and Prototype Regularization.

Models	IV-O ↓	IIVR-O ↓	IV-R ↓	IIVR-R ↓
Motifs [27, 39]	9.73	1.93	1.41	2.72
VCTree [27, 29]	8.31	2.11	1.50	2.78
Transformer [27, 31]	9.08	2.05	1.44	2.76
G-RCNN [13, 36]	8.76	1.99	1.46	2.81
GPS-Net [13, 17]	9.36	2.07	1.53	2.69
PE-Net	**0.74**	**0.24**	**1.06**	**1.67**

Table 5. Quantitative results on representation quality compared with classical SGG models under the PredCls task on VG dataset. The lower values indicate representations with higher quality.

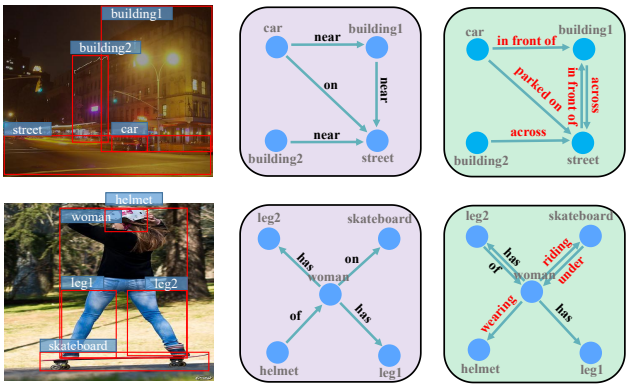


Figure 5. Visualization Results of Motifs (in purple) and PE-Net (in green) on the PredCls task.

PL, which discards the $\mathcal{L}_{r\_sim}$ and $\mathcal{L}_{r\_euc}$ in Eq. (14). Exp 3, PE-Net is trained with both PL and PR, *i.e.*, Eq. (14). When constrained by PL in Exp 2, the model outperforms the baseline (*i.e.*, Exp 1) on all metrics under three sub-tasks (*e.g.*, 25.4% *vs.* 20.0% at mR@100, and 70.1% *vs.* 68.2% at R@100 on PredCls). This verifies that PL effectively helps PE-Net to establish matching between entities and predicates for accurate relation recognition. Furthermore, after being integrated with PR in Exp 3, our PE-Net obtains significant gains on mR@K (*e.g.*, 33.8% *vs.* 25.4% at mR@100 on PredCls), which demonstrates PR's effectiveness in enlarging the distinction between prototypes achieving reliable entity-predicate matching. However, we observe that the improvement of mR@K brings a slight drop at R@K. That is because our model can reasonably classify some predicates (head classes) into their corresponding fine-grained ones (tail classes), *i.e.*, from "on" to "standing on/laying on/walking on". And the drops of Recall on those head predicates are inevitable, which is also observed in fine-grained classification [22] and long-tailed tasks [8].

4.7. Visualization Results

To verify that our proposed PE-Net is capable of making reliable relation recognition, we make a comparison between scene graphs generated by Motifs [39] (in purple) and our method (in green) in Fig. 5. In the first example, our method predicts informative relations such as "car-parked on-street" and "building1-across-street" instead of "car-on-street" and "building1-near-street". Similarly, in the second example, our method generates fine-grained predicates, *e.g.*, "wearing" and "riding". These results demonstrate that our method has a stronger predicate recognition ability than Motifs, which generates accurate relations for comprehensive scene understanding.

5. Conclusion

In this work, we propose a novel Prototype-based Embedding Network (PE-Net), which produces compact and distinctive entity/predicate representations for SGG task. Towards this end, the PE-Net models entity and predicate instances with prototype-based representations and then matches entity pairs with predicates for relation recognition. Moreover, we propose a Prototype-guided Learning strategy (PL) and Prototype Regularization (PR) to help PE-Net efficiently learn entity-predicate matching. Finally, our method achieves new state-of-the-art performances on both Visual Genome and Open Images datasets, which demonstrates the effectiveness of our methods.

Broader Impact and Limitations. Our work presents a powerful and efficient SGG method, which predicts relations between entities without message-passing module. The merit greatly reduces the computational complexity and enables SGG to be widely used in real-world applications, such as autonomous driving and intelligent robotics. However, our method is sensitive to the detector's recognition ability for entities, which limits its performance on SGCls and SGDet subtasks. Therefore, a more robust modeling method should be explored in further work.

Acknowledgment. This study is supported by grants from National Key R&D Program of China (2022YFC2009903/2022YFC2009900), the National Natural Science Foundation of China (Grant No. 62122018, No. 62020106008, No. 61772116, No. 61872064), Fok Ying-Tong Education Foundation(171106), SongShan Laboratory YYJC012022019, and Open Research Projects of Zhejiang Lab (No. 2019KD0AD01/011).

References

[1] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation. In *CVPR*, 2019. 1, 6

[2] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge J. Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, 2019. 6

[3] Xingning Dong, Tian Gan, Xuemeng Song, Jianlong Wu, Yuan Cheng, and Liqiang Nie. Stacked hybrid-attention and group collaborative learning for unbiased scene graph generation. In *CVPR*, 2022. 1, 6

[4] Lianli Gao, Yu Lei, Pengpeng Zeng, Jingkuan Song, Meng Wang, and Heng Tao Shen. Hierarchical representation network with auxiliary tasks for video captioning and video question answering. *TIP*, 31, 2021. 1

[5] Wenya Guo, Ying Zhang, Jufeng Yang, and Xiaojie Yuan. Re-attention for visual question answering. *TIP*, 30, 2021. 1

[6] Yuyu Guo, Lianli Gao, Xuanhan Wang, Yuxuan Hu, Xing Xu, Xu Lu, Heng Tao Shen, and Jingkuan Song. From general to specific: Informative scene graph generation via balance adjustment. In *ICCV*, 2021. 1, 3, 6

[7] Yuyu Guo, Jingkuan Song, Lianli Gao, and Heng Tao Shen. One-shot scene graph generation. In *ACM MM*, 2020. 1, 3

[8] Agrim Gupta, Piotr Dollár, and Ross B. Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 8

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6

[10] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *CVPR*, 2015. 1

[11] Lin Li, Long Chen, Yifeng Huang, Zhimeng Zhang, Songyang Zhang, and Jun Xiao. The devil is in the labels: Noisy label correction for robust scene graph generation. In *CVPR*, 2022. 3, 6

[12] Rongjie Li, Songyang Zhang, and Xuming He. Sgtr: End-to-end scene graph generation with transformer. In *CVPR*, 2022. 6

[13] Rongjie Li, Songyang Zhang, Bo Wan, and Xuming He. Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *CVPR*, 2021. 1, 3, 5, 6, 7, 8

[14] Wei Li, Haiwei Zhang, Qijie Bai, Guoqing Zhao, Ning Jiang, and Xiaojie Yuan. Ppdl: Predicate probability distribution based loss for unbiased scene graph generation. In *CVPR*, 2022. 1, 6

[15] Xiangpeng Li, Jingkuan Song, Lianli Gao, Xianglong Liu, Wenbing Huang, Xiangnan He, and Chuang Gan. Beyond rnns: Positional self-attention with co-attention for video question answering. In *AAAI*, 2019. 1

[16] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 6

[17] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *CVPR*, 2020. 6, 7, 8

[18] Xin Lin, Changxing Ding, Yibing Zhan, Zijian Li, and Dacheng Tao. Hl-net: Heterophily learning network for scene graph generation. In *CVPR*, 2022. 1, 6

[19] Xin Lin, Changxing Ding, Jing Zhang, Yibing Zhan, and Dacheng Tao. Ru-net: Regularized unrolling network for scene graph generation. In *CVPR*, 2022. 5, 6, 7

[20] Cewu Lu, Ranjay Krishna, Michael S. Bernstein, and Fei-Fei Li. Visual relationship detection with language priors. In *ECCV*, 2016. 2

[21] Xinyu Lyu, Lianli Gao, Yuyu Guo, Zhou Zhao, Hao Huang, Heng Tao Shen, and Jingkuan Song. Fine-grained predicates learning for scene graph generation. In *CVPR*, 2022. 1, 3, 5

[22] Xinyu Lyu, Lianli Gao, Pengpeng Zeng, Heng Tao Shen, and Jingkuan Song. Adaptive fine-grained predicates learning for scene graph generation. *arXiv*, 2022. 8

[23] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *ACL*, 2014. 3

[24] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 3, 6

[25] Brigit Schroeder and Subarna Tripathi. Structured query-based image retrieval using scene graphs. In *CVPR*, 2020. 1

[26] Mohammed Suhail, Abhay Mittal, Behjat Siddiquie, Chris Broaddus, Jayan Eledath, Gérard G. Medioni, and Leonid Sigal. Energy-based learning for scene graph generation. In *CVPR*, 2021. 7

[27] Kaihua Tang. A scene graph generation codebase in pytorch, 2020. https://github.com/KaihuaTang/Scene-Graph-Benchmark.pytorch. 3, 5, 6, 7, 8

[28] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *CVPR*, 2020. 1, 3, 5, 6, 7

[29] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *CVPR*, 2019. 1, 3, 5, 6, 7, 8

[30] Yao Teng and Limin Wang. Structured sparse r-cnn for direct scene graph generation. In *CVPR*, 2022. 6

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 8

[32] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017. 6

[33] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *CVPR*, 2017. 3, 5

[34] Shaotian Yan, Chen Shen, Zhongming Jin, Jianqiang Huang, Rongxin Jiang, Yaowu Chen, and Xian-Sheng Hua. PCPL: predicate-correlation perception learning for unbiased scene graph generation. In *ACM MM*, 2020. 1

[35] Gengcong Yang, Jingyi Zhang, Yong Zhang, Baoyuan Wu, and Yujiu Yang. Probabilistic modeling of semantic ambiguity for scene graph generation. In *CVPR*, 2021. 6

[36] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph R-CNN for scene graph generation. In *ECCV*, volume 11205, 2018. 3, 6, 7, 8

[37] Jing Yu, Yuan Chai, Yujing Wang, Yue Hu, and Qi Wu. Cogtree: Cognition tree loss for unbiased scene graph generation. In *IJCAI*, 2021. 1, 3, 6

[38] Jin Yuan, Shuai Zhu, Shuyin Huang, Hanwang Zhang, Yaoqiang Xiao, Zhiyong Li, and Meng Wang. Discriminative style learning for cross-domain image captioning. *TIP*, 31, 2022. 1

[39] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, 2018. 1, 3, 6, 7, 8

[40] Pengpeng Zeng, Lianli Gao, Xinyu Lyu, Shuaiqi Jing, and Jingkuan Song. Conceptual and syntactical cross-modal alignment with cross-level consistency for image-text matching. In *ACM MM*, 2021. 1

[41] Pengpeng Zeng, Haonan Zhang, Lianli Gao, Jingkuan Song, and Heng Tao Shen. Video question answering with prior knowledge and object-sensitive learning. *TIP*, 31, 2022. 1

[42] Pengpeng Zeng, Haonan Zhang, Jingkuan Song, and Lianli Gao. S2 transformer for image captioning. In *IJCAI*, 2022. 1

[43] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. Visual translation embedding network for visual relation detection. In *CVPR*, 2017. 6

[44] Ji Zhang, Kevin J Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. Graphical contrastive losses for scene graph generation. In *CVPR*, 2019. 5

[45] Mingxing Zhang, Yang Yang, Hanwang Zhang, Yanli Ji, Heng Tao Shen, and Tat-Seng Chua. More is better: Precise and detailed image captioning using online positive recall and missing concepts mining. *TIP*, 28, 2019. 1

[46] Chaofan Zheng, Lianli Gao, Xinyu Lyu, Pengpeng Zeng, Abdulmotaleb El Saddik, and Heng Tao Shen. Dual-branch hybrid learning network for unbiased scene graph generation. *arXiv*, 2022. 1, 2

[47] Chaofan Zheng, Xinyu Lyu, Yuyu Guo, Pengpeng Zeng, Jingkuan Song, and Lianli Gao. Learning to generate scene graph from head to tail. In *ICME*, 2022. 1