Multi-Prototype Space Learning for Commonsense-Based Scene Graph Generation

Lianggangxu Chen[1*], Youqi Song[1*], Yiqing Cai[1], Jiale Lu[1], Yang Li[1], Yuan Xie[1], Changbo Wang[1†], Gaoqi He[1, 2†]

[1]School of Computer Science and Technology, East China Normal University, Shanghai, China
[2]Chongqing Key Laboratory of Precision Optics, Chongqing Institute of East China Normal University, Chongqing, China
{lgxchen, youqisong, yqcai, jllu}@stu.ecnu.edu.cn, yli@cs.ecnu.edu.cn, xieyuan8589@foxmail.com
{cbwang, gqhe}@cs.ecnu.edu.cn

Abstract

In the domain of scene graph generation, modeling commonsense as a single-prototype representation has been typically employed to facilitate the recognition of infrequent predicates. However, a fundamental challenge lies in the large intra-class variations of the visual appearance of predicates, resulting in subclasses within a predicate class. Such a challenge typically leads to the problem of misclassifying diverse predicates due to the rough predicate space clustering. In this paper, inspired by cognitive science, we maintain multi-prototype representations for each predicate class, which can accurately find the multiple class centers of the predicate space. Technically, we propose a novel multi-prototype learning framework consisting of three main steps: prototype-predicate matching, prototype updating, and prototype space optimization. We first design a triple-level optimal transport to match each predicate feature within the same class to a specific prototype. In addition, the prototypes are updated using momentum updating to find the class centers according to the matching results. Finally, we enhance the inter-class separability of the prototype space through iterations of the inter-class separability loss and intra-class compactness loss. Extensive evaluations demonstrate that our approach significantly outperforms state-of-the-art methods on the Visual Genome dataset.

Introduction

Scene graph generation (SGG) plays a vital role in visual scene understanding, which aims to detect objects and represent their relationships using predicates in images (Sun et al. 2023). Consequently, SGG can provide valuable assistance for subsequent computer vision tasks, including visual question answering (Lei et al. 2023) and image captioning (Yang, Liu, and Wang 2022).

Existing SGG methods can be roughly categorized into two groups by whether to use external knowledge: 1) ***Visual contextual based methods*** successively pass visual features through a given network, such as graph neural networks (Li et al. 2018) and transformers (Dhingra, Ritter, and Kunz 2021), to update the feature representations of predicates. However, the long-tail distribution of predicate classes limits the performance of existing visual contextual based methods (Chen et al. 2019b; Tang et al. 2020). As Zareian et al. point out (Zareian et al. 2020), head predicates, such as **on**, crowd out rarer but more informative tail predicates, such as **standing on**. 2) ***Commonsense based methods*** extract commonsense from multiple external knowledge bases (Sharifzadeh et al. 2022; Zareian 2020; Sharifzadeh, Baharlou, and Tresp 2021) to refine object and predicate features to improve recall for the tail predicate classes. Commonsense based methods typically represent the class center of each predicate as a single prototype (Zareian 2020; Zheng et al. 2023a), where the single prototype is embedded into a word vector trained from external knowledge bases. The visual features of predicate instances are mapped into the prototype space, and then each predicate instance is assigned a corresponding predicate class by searching the nearest prototype (Figure 1(a)).

Despite the progress made by existing commonsense-based scene graph generation (C-SGG) methods, the performance of diverse predicates is still not satisfied in the single-prototype setting. In Figure 1(a), the predicate instance **man-riding-board** is closer to the prototype **lying on** than to the prototype **riding** ($d_2 < d_1$) due to the visual similarity. As a result, the predicate instance **man-riding-board** is misclassified as **man-lying on-board**. Specifically, predicates with large intra-class diversity, such as **riding**, **standing on** and **lying on**, are more likely to be misclassified.

Naturally, we investigated the underlying reasons behind the misclassification of diverse predicates. The reason is that the large intra-class variation is still not well addressed in the single-prototype setting. For example, as illustrated in Figure 1(a), there is a substantial diversity of visual appearance for **bicycle riding**, **horseback riding**, and **wave riding** even if they respectively represents a single predicate category of **riding**. This intra-class variation results in subclasses clearly identified within each predicate category. Modern prototype category theory of cognitive science argues that categories do not have single, central prototype, but rather a polycentric structure with multiple prototypes (Spivak 2014). These prototypes are not fixed, but rather are defined by a set of representative features induced by existing instances (Lakoff 2007). Modern prototype category theory motivated us to rethink C-SGG from the polycen-

*These authors contributed equally.
†Changbo Wang and Gaoqi He are the corresponding authors.

tric view of categories. Each predicate category has multiple class centers, which are represented by multiple-prototypes. In training, multiple-prototypes can be learned by clustering similar predicate features. Therefore, modern prototype category theory enables us to better account for **intra-class diversity** within categories.

In this paper, to address the problem of misclassifying diverse predicates in C-SGG, we propose a novel multi-prototype learning (MPL) framework. Firstly, the pre-trained model (Zareian 2020) is utilized to extract unclassified predicate features and initial prototypes of commonsense. Then, the triple-level optimal transport (TOT) is applied to match unclassified predicate features within the same class to the initial prototypes of that class. TOT calculates the relational distance of object-predicate and subject-predicate to avoid the incorrect matching between predicates and prototypes. The multi-prototypes of each class are subsequently updated using momentum updating, which allows the model to find multiple class centers of predicates accurately. For example, in Figure 1(b), in addition to the **horseback riding** prototype with distance d_1, there is also a **wave riding** prototype with a smaller distance d_3 ($d_3 < d_2$). As a result, the predicate instance **man-riding-board** is correctly classified as **riding**. Furthermore, to improve the structure of the prototype space, we introduce an intra-class compactness loss that encourages prototypes of the same class to form tight clusters. Additionally, we propose an inter-class separability loss that forces prototypes of different classes to be far apart. During testing, each unclassified predicate is assigned to the same class as the nearest prototype (Figure 1(b)).

The contributions can be summarized as follows:

- A novel MPL framework is proposed for C-SGG. The proposed framework matches each predicate to the most appropriate prototype. MPL improves the accuracy of predicate inference and enhances the interpretability of the model. To the best of our knowledge, this is the first exploration to realize multi-prototype modeling for predicate inference.
- A triple-level optimal transport method is proposed to avoid incorrect matching between predicates and prototypes. TOT achieves a trade-off between predicate-wise comparison and triple-wise comparison when computing the optimal transport matrix.
- The loss of both inter-class separability and intra-class compactness are proposed to optimize multi-prototype space. These two losses directly optimize both the prototype-prototype and prototype-predicate distance, which efficiently encourage diversity between prototypes and avoid inter-class overlaps, respectively.

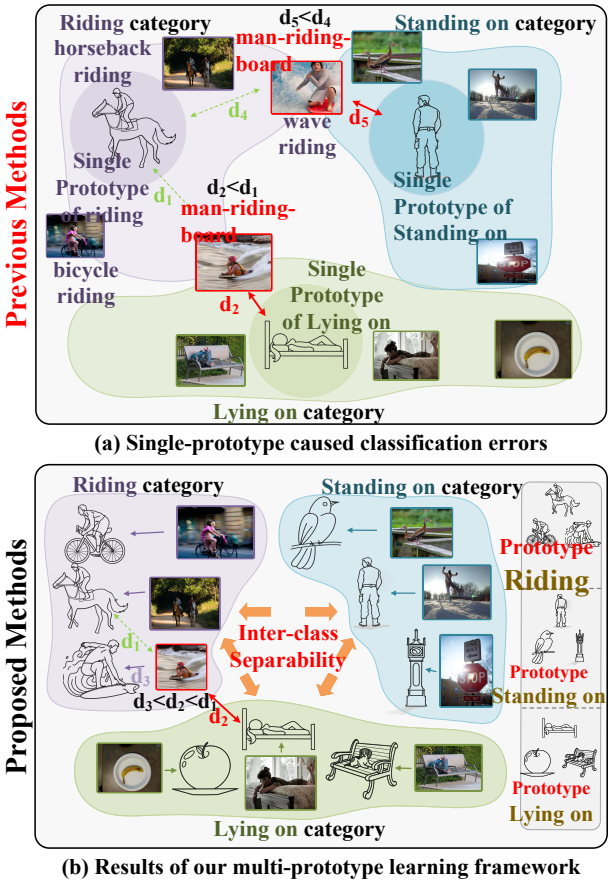

Figure 1: Comparison of previous single-prototype methods and our proposed multi-prototype method. (a) Previous approaches attempt to represent commonsense as a single prototype (Zareian 2020; Zheng et al. 2023a). The large intra-class variance leads to inaccurate representation of class features in single prototype setting, resulting in classification errors. (b) Visualization of the results achieved by our model.

Related Work

SGG with Commonsense

Various methods have been proposed to incorporate commonsense into the C-SGG task, including statistical probability-based approaches, external knowledge-based approaches, graph-based approaches, and natural language text-based approaches.

Statistical probability-based approaches, such as (Chen et al. 2019b; Hou et al. 2019), utilize co-occurrence statistics to model commonsense. However, these methods can be limited by incomplete and biased representations of commonsense due to the availability of training data. To address this, some methods use external knowledge sources like WordNet (Miller 1995) and ConceptNet (Liu and Singh 2004; Gu et al. 2019). Graph-based approaches, like (Zareian 2020; Lin, Zhu, and Liang 2022), model commonsense as a graph but may be limited by incomplete knowledge graphs. Recently, natural language text-based approaches have been introduced that extract commonsense representations from large text corpora (Zhong et al. 2021; Yao et al. 2021; Sharifzadeh et al. 2022). In this paper, we are the first to use multi-prototype to find multiple class cen-

ters in SGG.

Optimal Transport

In the field of computer vision, optimal transport (OT) has been employed to address the low prediction accuracy issue in crowded scenarios by formulating the assigning procedure in object detection as an OT problem (Ge et al. 2021). Additionally, OT has been used in the field of image segmentation and semantic correspondence (Liu et al. 2020). For point cloud analysis, Li *et al.* utilized OT to establish correspondences between two point clouds, which approximates scene flow and overcomes the problem of the scarcity of annotated scene flow data in a self-supervised manner (Li, Lin, and Xie 2021). In the field of natural language processing, OT distance has been utilized in estimating the necessary quantity of alignment for instance Word Mover's Distance (Chen et al. 2019a). In this paper, triple-level optimal transport used as marginal constraints to suppress many-to-one matching of predicates and prototype.

Prototype Learning and Contrastive Learning

Few-shot learning approaches have achieved success in recognizing classes with few training examples, but offline methods prevent end-to-end training of prototypes. For instance, Allen *et al.* (Allen et al. 2019) proposed to learn prototypes to address the challenge of recognizing classes with few training examples. In the context of 3D point cloud deep learning, multi-prototype methods were briefly mentioned in (Zhao, Chua, and Lee 2021), which generated multiple prototypes by farthest point sampling on the embedding space for the support set. Kim *et al.* (Kim et al. 2018) proposed end-to-end deep quadruplet networks to map prototype and real image embeddings into a common space for image classification. Recently, Kang *et al.* (Kang and Ahn 2022) proposed a variational multi-prototype encoder for object recognition, where multiple prototypes are used to represent each object class. However, these methods do not provide concrete evidence of the existence of multiple prototypes. In this work, we allocate prototype representation for predicates through OT and demonstrate the existence of multi-prototypes through visualization experiments.

Problem Formulation and Definitions

SGG: The task of SGG first parses the input image $\mathcal{I}$ into an unclassified scene graph $\mathcal{G}_s = \{\mathcal{V}_s = (\mathcal{V}_o \cup \mathcal{V}_p), \mathcal{E}_s\}$, where $\mathcal{V}_o = \{\mathbf{v}_o^i\}_{i=1}^N, \mathbf{v}_o^i$ is the feature of object node and $\mathcal{V}_p = \{\mathbf{v}_p^i\}_{i=1}^M, \mathbf{v}_p^i$ is the feature of predicate node, where N and M denote the number of objects and predicates, respectively. The undirected edges in $\mathcal{E}_s$ connect predicate and object nodes. In the prediction stage, each object node $\mathbf{v}_o^i$ will be assigned a class label from a set of object classes C_o. Each predicate node $\mathbf{v}_p^i$ will be assigned a class label from a set of predicate classes C_p.

C-SGG Models: The commonsense can be formulated as a graph $\mathcal{G}_c = (\mathcal{V}_c, \mathcal{E}_c)$, where $\mathcal{V}_c = \{\mathbf{v}_c^i\}_{i=1}^{C_o+C_p}$ and $\mathcal{E}_c$ denote the node set and edge set, respectively. $\mathcal{V}_c$ contains both object nodes and predicate nodes. Each node in $\mathcal{V}_c$ represents the word embeddings of its class label and each class label appears in exactly one node. The edge between two nodes represents the statistical co-occurrence frequency.

In current C-SGG, a predominant strategy for classification is to use a single prototype representation for each class (Zareian 2020). Specifically, each predicate embedding $\mathbf{v}_p^i$ is utilized for C_p-way classification, as described by Eqn. (1):

$$P(\mathbf{v}_p^i \mid p) = \frac{\exp\left(\mathbf{v}_p^{i\top}\mathbf{v}_c^i\right)}{\sum_{\mathbf{v}_c^{i'}} \exp\left(\mathbf{v}_p^{i\top}\mathbf{v}_c^{i\prime}\right)}, \tag{1}$$

where $P(\mathbf{v}_p^i \mid p)$ is the probability that $\mathbf{v}_p^i$ belonging to predicate class $p \in C_p$. $\mathbf{v}_c^i$ represents a commonsense node, and Eqn. (1) computes a pairwise similarity from each predicate instance to all commonsense nodes. Consequently, $\mathbf{v}_c^i$ is treated as a single prototype representation.

Although Eqn. (1) reflects the prevailing notion of single prototype learning in C-SGG, it has the following problems: 1) Eqn. (1) defines each class by a single prototype representation, neglecting intra-class diversity. 2) The optimization of Eqn. (1) through cross-entropy loss ignores the absolute distances between predicates and prototypes, which leads to the inter-class overlaps.

Method

Overview

Our framework is illustrated as Figure 2. Given an unclassified scene graph $\mathcal{G}_s$ and prototype $\mathcal{V}_c$ (Figure 2(a)), optimal transport is performed to obtain the best matching results between prototype and predicate features (Figure 2(b)). According to the matching results, we calculate the average predicate features corresponding to the prototype, and then momentum update the prototype features in each training iteration. The inter-class separability and intra-class compactness loss are utilized to optimize prototype space. Finally, the MPL outputs the multi-prototypes $\mathbf{C}$. In the testing stage, we assign a label to the nearest predicate features $\mathcal{V}_p$ based on $\mathbf{C}$ to generate the final scene graph (Figure 2(c)).

Multi-prototype Learning

In this section, the details of the multi-prototype learning framework are described as follows:

Step 1: Initial Predicate Features and Prototypes Extracting. Given an image $\mathcal{I}$, the object detector is first used to extract a set of bounding box proposals $\mathcal{B}$ and the corresponding region objects feature vectors are $\mathcal{V}_o$. The object detector also generates visual features $\mathcal{V}_p$ for predicates to focus on the closed boxes of any two objects (Xu et al. 2017).

To identify the necessary prototypes that can effectively differentiate a predicate class, the vector of one predicate word vector is diverged into K prototypes. Specifically, given a vector $\mathbf{z} \sim \mathcal{N}(0, 1)$, a small hyper-sphere is defined with radius r centered at the prototype $\mathcal{V}_c$. We random sample vectors $\mathbf{C} = \mathcal{V}_c + \rho$ as multi-prototypes with the sphere, where $\rho \sim \mathcal{U}[-r, r]$ is a randomly sampled vector from a uniform distribution $\mathbf{z}$.

Step 2: Prototype-Predicate Feature Matching by Triple-level Optimal Transport. After obtaining the initial

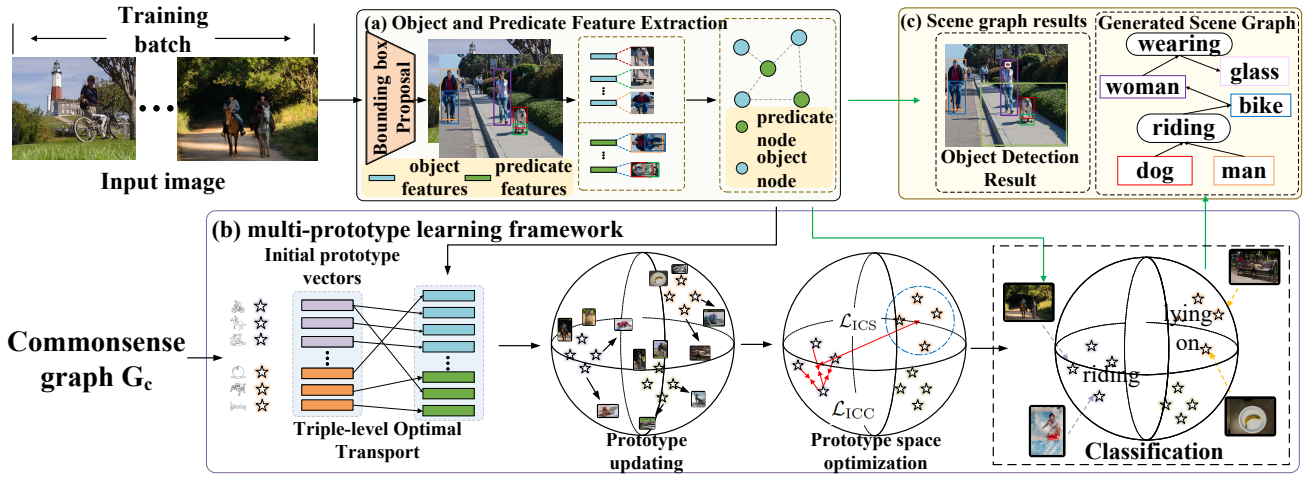

Figure 2: An overview of our full method. (a) Unclassified scene graphs extraction. (b) Our proposed MPL framework. (c) Generated scene graph results. The black arrow represents the training phase, and the green arrow represents the testing phase.

representation of multi-prototypes $\mathbf{C}$, the predicate features in the same class are matched to the prototypes belonging to that class, and then update the prototypes according to the matched predicate features. Step 2 is proposed to solve the problem 1) of Eqn. (1).

Traditional Optimal Transport. Given M predicate feature vectors $\mathbf{P} = \left[\mathbf{v}_p^1, \ldots, \mathbf{v}_p^M\right]$ that belong to a class $p \in C_p$, our goal is to match $\mathbf{P}$ to the multi-prototype $\mathbf{C}^p = \left[\mathbf{v}_c^1, \ldots, \mathbf{v}_c^K\right]$, where $\mathbf{C}^p \in \mathbf{C}$. This matching results are denoted by $\mathbf{Q} = \left[\mathbf{q}^1, \ldots, \mathbf{q}^M\right]$, and $\mathbf{Q}$ are optimized by maximizing the similarity between the predicates and the prototypes, i.e.,

$$\max_{\mathbf{Q}} \operatorname{Tr}(\mathbf{Q}^T \mathbf{C}^{pT} \mathbf{P}), \tag{2}$$

where $\operatorname{Tr}(\cdot)$ is the trace operation (Zhang et al. 2023a).

However, traditional OT can lead to incorrect matching between prototypes and predicates. As illustrated in Figure 3(b), when purely relying on the Wasserstein distance in Eqn. (2) (Lin and Chan 2023), the predicate prototype **standing on** (in red) may be wrongly matched with the image corresponding to **riding** (in blue) because of their visual similarity.

Triple-level Optimal Transport. To address the above-mentioned mismatch, we propose the Triple-level Optimal Transport. In particular, TOT is an optimization problem consisting of a Wasserstein distance term and a triple distance term:

$$\begin{aligned} &\max_{\mathbf{Q}} \underbrace{(1-\beta)\operatorname{Tr}(\mathbf{Q}^T \mathbf{C}^{pT} \mathbf{P})}_{\text{Wasserstein term}} \\ &+ \underbrace{\beta(-\boldsymbol{D}_{oc} \boldsymbol{Q} \boldsymbol{D}_{op}^T - \boldsymbol{D}_{sc} \boldsymbol{Q} \boldsymbol{D}_{sp}^T)}_{\text{triple-distance term}} \\ &+ \varepsilon \left(\mathrm{KL}\left(Q 1_J \| \frac{1}{I} \mathbf{1}_I\right) + \mathrm{KL}\left(Q^T \mathbf{1}_I \| \frac{1}{J} \mathbf{1}_J\right)\right), \end{aligned} \tag{3}$$

where $\boldsymbol{D}_{op} = d(\mathbf{v}_{ob}^i \in \mathbf{v}_o^i, \mathbf{v}_p^i)$ and $\boldsymbol{D}_{sp} = d(\mathbf{v}_{sb}^i \in \mathbf{v}_o^i, \mathbf{v}_p^i)$ are the object-predicate distance and subject-predicate distance in visual triplets, respectively. Similarly, $\boldsymbol{D}_{oc}$ and $\boldsymbol{D}_{sc}$ are the object-predicate distance and subject-predicate distance in the commonsense triplet, respectively. TOT achieves a trade-off between predicate-wise comparison and triple-wise comparison when computing the optimal transport matrix, in which the significance of the two terms is controlled by the hyperparameter $\beta \in [0, 1]$. For the marginals of the transport matrix, instead of imposing strict equality constraints (Zhou et al. 2023), we add two regularizers to penalize the KL-divergences between them and uniform distributions (Wu et al. 2023). The terms $\frac{1}{I}\mathbf{1}_I$ and $\frac{1}{J}\mathbf{1}_J$ in Eqn. (3) represent uniform distribution that assigns the equal probability to each elements in matrix I and J. The significance of the two regularizers is controlled by the hyperparameter ε (Demetci et al. 2020). The regularizers allow us to learn the significance and the assignment of the predicates while avoiding trivial solutions (Figure 3(a)).

The matrix value of $\mathbf{Q}^*$ stores the matching results of prototypes and features. $\mathbf{Q}^*$ can be easily calculated throughout iterative matrix multiplication using the Sinkhorn-Knopp algorithm (Cuturi 2013).

Step 3: Prototype Updating. Following the matching results of each predicate $\mathbf{v}_p^i$ to its corresponding k-th prototype for class p, the multi-prototypes $\mathbf{C}^p = \left\{\mathbf{v}_c^k\right\}_{k=1}^K$ are updated as centers of the matched predicate features (Fickinger et al. 2021). Specifically, after each training batch, each prototype is updated using the following equation:

$$\mathbf{v}_c^k = \mu \mathbf{v}_c^k + (1-\mu)\bar{\mathbf{v}}_p, \tag{4}$$

where $\mu = 0.999$ is a momentum coefficient and $\bar{\mathbf{v}}_p$ represents the mean vector of the predicate features matched to prototype $\mathbf{v}_c^k$ in $\mathbf{Q}^*$.

Step 4: Prototype Space Optimization. Step 4 aims to solve the problem 2) of Eqn. (1). It is essential to balance the

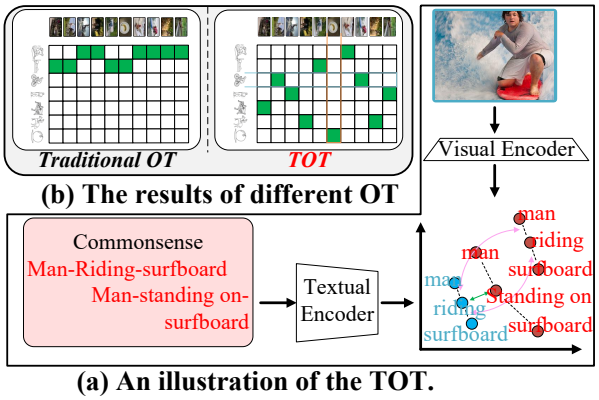

Figure 3: (a) Given the commonsense and the visual features, the TOT not only considers their predicate-wise distance (the green arrows) but also considers the triple distance of object-predicate (man-riding) and subject-predicate (riding-surfboard). Even if the predicate-wise distance is large, the triple distance can be small (i.e., the lengths of the black dotted lines are similar). (b) Many predicate instances are matched to one prototype by traditional OT algorithm (Hartigan and Wong 1979). Right: TOT suppresses the many-to-one matching.

emphasis on prototype diversity to avoid equally distanced prototypes that can harm separability in predicate classes (Kang and Ahn 2022). To achieve this, we derive an inter-class separability $\mathcal{L}_{\text{ICS}}$ for prototype space optimization:

$$\mathcal{L}_{\text{ICS}} = -\log \frac{\exp\left(\mathbf{v}_p^\top \mathbf{v}_c^k/\tau\right)}{\exp\left(\mathbf{v}_p^\top \mathbf{v}_c^k/\tau\right) + \sum_{\gamma \in \mathbf{C}} \exp\left(\mathbf{v}_p^\top \gamma/\tau\right)}, \tag{5}$$

where γ represents the set of other $KC_p - 1$ prototypes, and τ is a control parameter. Eqn. (5) enforces each predicate instance to be closer to its matched prototype and dissimilar to other irrelevant prototypes.

However, Eqn. (5) does not take into account the goal of making predicate features that belong to the same prototype more compact. Therefore, we use an intra-class compactness loss $\mathcal{L}_{\text{ICC}}$ to minimize the distance between each predicate and its matched prototype:

$$\mathcal{L}_{\text{ICC}} = \delta^2 \left(\sqrt{1 + (\frac{\mathbf{v}_c^k - \mathbf{v}_p}{\delta})^2} - 1 \right), \tag{6}$$

where δ is a hyperparameter that controls the shape of Pseudo-Huber loss curve. Compared to Euclidean distance, Pseudo-Huber loss is more robust to outliers and thus more suitable for long-tailed scenarios (Barron 2019).

Finally, our complete loss function is formed by the following combination of loss functions:

$$\mathcal{L}_{\text{C-SGG}} = \mathcal{L}_{\text{focal}} + \lambda_1 \mathcal{L}_{\text{ICS}} + \lambda_2 \mathcal{L}_{\text{ICC}}, \tag{7}$$

where $\mathcal{L}_{\text{focal}}$ is the focal loss for predicate classification and object classification (Lin et al. 2017).

Training Details

The Adam (Kingma and Ba 2014) optimizer is used with batch size 4 during training. The initial learning rate is 0.0001 for the backbone and 0.0003 for our MPL framework, which decays by a factor of 10 for every 5 epochs. The weight decay is set as 0.0001. The training of our full method takes approximately 15-16 hours. The prototype number K is set to 3. λ_1 and λ_2 were set to 0.1 and 0,1, respectively, by cross-validation. τ and δ are both set to 0.1. ε is set to 0.05. The radius r is set to 0.0001. The trade-off hyperparameter β is set to 0.1. Importantly, our initial prototype features are generated from commonsense. We use a pre-trained graph convolution network (GCN) in (Zareian 2020) to make the initial prototype feature have context information.

Experiments

Dataset and Evaluation Settings

Experiment results using the large-scale Visual Genome (VG) dataset are presented (Krishna et al. 2017). The training set of VG includes 70% images, with 5K images used as a validation subset. The testing set is composed of the remaining 30% of images. Three standard evaluation modes are used (Lu et al. 2016): (i) Scene Graph Detection (SGDET); (ii) Scene Graph Classification (SGCLS); (iii) Predicate Classification (PREDCLS). The image-wise Recall (R) and Mean Recall (mR) is used, which measure the fraction of ground truth visual predicates appearing in top-20, top-50, and top-100 confident predictions (Chen et al. 2019b; Tang et al. 2019).

Comparisons with the State-of-the-art Methods

Table 1 shows a comparison of our approach with other commonsense-based methods considered in Recall. Our method outperforms all the other models on all three tasks, achieving 54.2 on average. We train MPL on the Open Images V6 dataset (Kuznetsova et al. 2020) and compare it with other complex methods, as shown in Table 2. The experimental results show that MPL achieved SOTA under all settings (Following (Cong, Yang, and Rosenhahn 2023)), which demonstrates the powerful performance.

Ablation Studies of Different Modules

Effectiveness of MPL Framework: As shown in Table 3, Exp 1 denotes our full method. Three variant experiments are designed to validate the effect of the MPL framework. 1) Exp 2: We remove step 1 and step 2 in MPL framework while a fixed commonsense graph is used. It denotes that commonsense regresses to the single prototype state without multi-prototype evolution for predicate inference. From Exp 2, the performance consistently improves after the multi-prototype learning has been applied. 2) Exp 3: We replace step 2 with traditional OT. From Exp 3, we see a substantial performance gain after our optimal transport has been applied. This shows that the matching method based on TOT is superior to the traditional clustering algorithm. 3) Exp 4-6: We investigate our overall training objective in Eqn. (7). Adding $\mathcal{L}_{\text{ICS}}$ or $\mathcal{L}_{\text{ICC}}$ individually brings gains, revealing

Model	SGDET			SGCLS			PREDCLS			
	R@20	R@50	R@100	R@20	R@50	R@100	R@20	R@50	R@100	mean
BGNN (Li et al. 2021) (2021)	-	31.0	35.8	-	37.4	38.5	-	59.2	61.3	43.9
NICE (Li et al. 2022)	-	27.0	30.8	-	37.8	39.0	-	55.0	56.9	41.1
SMP (Zhang et al. 2023b)	-	32.6	36.9	-	39.9	40.7	-	66.3	68.0	47.4
GB-Net (Zareian 2020)	20.3	26.4	30.0	34.9	38.0	38.8	60.4	66.6	68.2	44.7
Motif + DNS (Yao et al. 2021)	-	30.9	35.1	-	38.4	39.3	-	64.4	66.4	45.8
TXM (Sharifzadeh et al. 2022)	-	-	-	-	39.0	39.9	-	66.7	68.3	-
KEM (Zheng et al. 2023b)	-	34.5	37.9	-	47.1	47.9	-	72.6	73.5	52.3
VS^3 (Zhang et al. 2023c)	27.8	36.6	41.5	-	-	-	-	-	-	-
PE-Net (Zheng et al. 2023a)	-	30.7	35.2	-	39.4	40.7	-	64.9	67.2	46.4
Our method	**28.8**	**37.6**	**42.2**	**39.8**	**48.0**	**49.2**	**66.4**	**73.3**	**75.1**	**54.2**

Table 1: Comparisons with state-of-the-arts on the VG dataset. Because some methods were not tested on R@20, we only compute the mean of the two tasks of R@50 and R@100.

Model	mR@50	R@50	wmAP rel	wmAP phr	score wtd
RelDN	33.98	73.08	32.16	33.39	40.84
VCTree	33.91	74.08	34.16	33.11	40.21
MOTIFS	32.68	71.63	29.91	31.59	38.93
TDE	35.47	69.30	30.74	32.80	39.27
GPS-Net	35.26	74.81	32.85	33.98	41.69
BGNN	40.45	74.98	33.51	34.15	42.06
BGNN+SCR (Kang and Yoo 2023)	42.43	75.21	33.98	35.13	42.66
SGTR (Li, Zhang, and He 2022)	-	59.91	36.98	38.73	42.28
RelTR (Cong, Yang, and Rosenhahn 2023)	-	71.66	34.19	37.46	42.99
Our method	**43.98**	**76.34**	**37.11**	**40.55**	**44.43**

Table 2: Comparison with other state-of-the-art methods on the Open Images V6 (Kuznetsova et al. 2020) test set.

Exp	Module	SGDET	
		mR@50	mR@100
1	our full method	**21.2**	**22.3**
2	w/o step 1 and step 2	18.8	19.8
3	Replace step 2 with traditional OT	19.1	20.1
4	w/o $\mathcal{L}_{\mathrm{ICS}}$ and $\mathcal{L}_{\mathrm{ICC}}$	19.7	20.7
5	w/o $\mathcal{L}_{\mathrm{ICS}}$	20.9	21.9
6	w/o $\mathcal{L}_{\mathrm{ICC}}$	20.7	21.7
7	K=1	19.7	20.6
8	K=2	20.1	21.1
9	K=3	**21.2**	**22.3**
10	K=4	19.9	20.9
11	K=5	19.3	20.2

Table 3: Ablation studies on the proposed MPL. w/o means removing the corresponding part in our model.

the value to supervise prototype-prototype and predicate-prototype distance explicitly.

Effectiveness of Prototype Number K. Exp 7-11 report the performance of our approach regarding the number of prototypes. From Exp 9, we observe that with a higher number K (1 to 3), the performance of our model increases from 19.7 to 21.2 on mR@50. When further considering an even higher number K (3 to 5), the performance gradually significantly degrades (21.2 to 19.3). Although higher prototype numbers allow the model to directly capture more intra-class information, noisy messages start to permeate throughout the OT and hamper the final prediction. Therefore, this suggests that $K = 3$ is the optimal choice.

Qualitative Results

Figure 4 shows six challenging images. For example, considering image (a), we succeeded in correcting the bias in the pre-trained commonsense based on the multi-prototype representation of **riding**, that is, from **lying on** to **riding**. Considering the image (b), the visual posture of the **dog** is deceptive, which makes it easier for the commonsense model to predict **sitting on**. With our novel MPL framework, we correctly identify the predicate **lying on** between **dog** and **bench**. In fact, this is precisely because our **lying on** prototype has differentiated the shape of **dog sitting on bench**. Another interesting example is the image (f), where our model can deal with a situation where the subject is **face**. Our model can correct the coarse predicate of **face on pole** to the fine predicate of **painted on**.

Can our MPL framework really help improve the recall of diverse predicates? Based on the clear definition of prototypes, our scene graph generation process can be easily understood as matching the most similar prototypes to each scene. In Figure 5, we demonstrate the matching of $K = 3$ prototypes for each class in the **riding** category. The matched prototypes are represented by distinct colors,

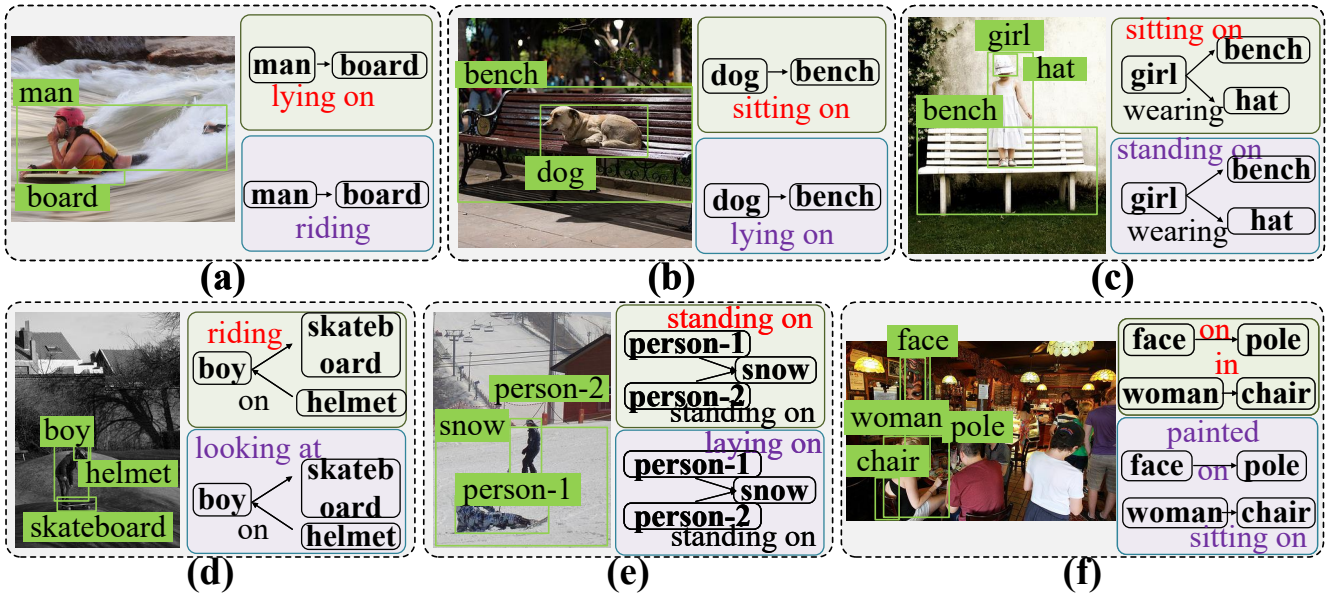

Figure 4: Qualitative examples of the improvement. On the right side of each image, the result of the commonsense-based method (Zareian 2020) is at the top, and our result is at the bottom. The purple are those correctly classified predicates (in ground truth), the red are those incorrectly classified predicates. For better viewing, we only show the failure cases.

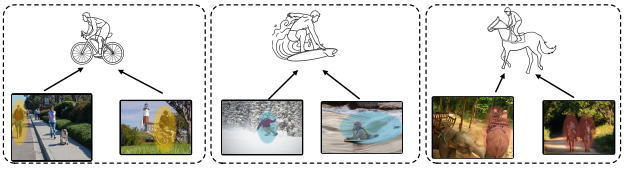

Figure 5: Visualization of predicate-prototype similarity for riding classes.

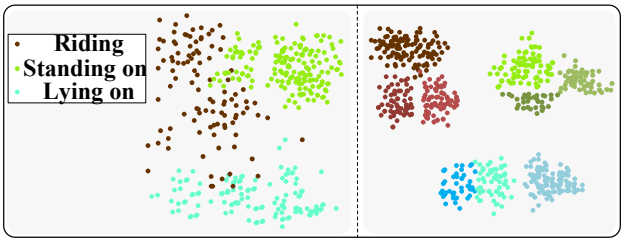

Figure 6: Embedding spaces learned by (left) single prototype model, and (right) our multi-prototype model. For better visualization, we show three classes of VG with three prototypes per class.

namely yellow, blue, and red.

To visualize the matching results, we use the color of the corresponding prototype to indicate the distance of each predicate to the closest prototype. The results show that the prototypes correspond well to meaningful patterns within the classes, which confirms their semantic significance.

In summary, the matching of prototypes based on their similarity to the input scene facilitates the generation of accurate and semantically meaningful scene graphs. Figure 5 provides a clear illustration of the effectiveness of this approach in the **riding** category.

Embedding Spaces Visualization. In Figure 6, on the left is a single prototype model (Zheng et al. 2023a), and on the right is our multi-prototype model. As can be observed, in our algorithm, the predicate embeddings that belong to the same prototypes are well-separated. This is because our model is based on a Wasserstein distance-based optimal transport approach, which reshapes the feature space by encoding the latent data structure into the embedding space. The embedding is directly supervised by both an inter-class separability loss and an intra-class compactness loss. These losses help improve the separability of different classes and the compactness of instances within the same class, respectively. By incorporating these losses, our model can better capture the underlying data structure and produce embeddings that are more informative and discriminative.

Conclusion

In this work, we have demonstrated that the single-prototype used in current C-SGG methods is inadequate for capturing intra-class diversity. To address the problem of misclassifying diverse predicates, we have proposed a novel MPL that enables better segmentation of the predicate space. Our approach included a predicate-prototype matching based on TOT, prototype updating, and prototype space optimization. Experimental results on the VG dataset have confirmed the benefits of maintaining multiple-prototypes.

Acknowledgments

This work was supported in part by Natural Science Foundation of Chongqing (No. CSTB2022NSCQ-MSX0552), National Natural Science Foundation of China (No. 62002121 and 62072183), Shanghai Science and Technology Commission (No. 21511100700 and 22511104600), and the Open Project Program of the State Key Lab of CAD&CG (No. A2203).

References

Allen, K.; Shelhamer, E.; Shin, H.; and Tenenbaum, J. 2019. Infinite mixture prototypes for few-shot learning. In *International Conference on Machine Learning*, 232–241. PMLR.

Barron, J. T. 2019. A general and adaptive robust loss function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4331–4339.

Chen, L.; Zhang, Y.; Zhang, R.; Tao, C.; Gan, Z.; Zhang, H.; Li, B.; Shen, D.; Chen, C.; and Carin, L. 2019a. Improving sequence-to-sequence learning via optimal transport. *arXiv preprint arXiv:1901.06283*.

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

Cong, Y.; Yang, M. Y.; and Rosenhahn, B. 2023. Reltr: Relation transformer for scene graph generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Cuturi, M. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26.

Demetci, P.; Santorella, R.; Sandstede, B.; Noble, W. S.; and Singh, R. 2020. Gromov-Wasserstein optimal transport to align single-cell multi-omics data. *BioRxiv*.

Dhingra, N.; Ritter, F.; and Kunz, A. 2021. BGT-Net: Bidirectional GRU transformer network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2150–2159.

Fickinger, A.; Cohen, S.; Russell, S.; and Amos, B. 2021. Cross-Domain Imitation Learning via Optimal Transport. *arXiv preprint arXiv:2110.03684*.

Ge, Z.; Liu, S.; Li, Z.; Yoshie, O.; and Sun, J. 2021. Ota: Optimal transport assignment for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 303–312.

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

Hartigan, J. A.; and Wong, M. A. 1979. Algorithm AS 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, 28(1): 100–108.

Hou, J.; Wu, X.; Qi, Y.; Zhao, W.; Luo, J.; and Jia, Y. 2019. Relational reasoning using prior knowledge for visual captioning. *arXiv preprint arXiv:1906.01290*.

Kang, H.; and Yoo, C. D. 2023. Skew Class-balanced Reweighting for Unbiased Scene Graph Generation. *arXiv preprint arXiv:2301.00351*.

Kang, J. S.; and Ahn, S. C. 2022. Variational Multi-Prototype Encoder for Object Recognition Using Multiple Prototype Images. *IEEE Access*, 10: 19586–19598.

Kim, J.; Lee, S.; Oh, T.-H.; and Kweon, I. S. 2018. Co-domain embedding using deep quadruplet networks for unseen traffic sign recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1): 32–73.

Kuznetsova, A.; Rom, H.; Alldrin, N.; Uijlings, J.; Krasin, I.; Pont-Tuset, J.; Kamali, S.; Popov, S.; Malloci, M.; Kolesnikov, A.; et al. 2020. The open images dataset v4. *International Journal of Computer Vision*, 1–26.

Lakoff, G. 2007. Cognitive models and prototype theory. *The cognitive linguistics reader*, 130–67.

Lei, S. W.; Gao, D.; Wu, J. Z.; Wang, Y.; Liu, W.; Zhang, M.; and Shou, M. Z. 2023. Symbolic replay: Scene graph as prompt for continual learning on vqa task. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 1250–1259.

Li, L.; Chen, L.; Huang, Y.; Zhang, Z.; Zhang, S.; and Xiao, J. 2022. The devil is in the labels: Noisy label correction for robust scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18869–18878.

Li, R.; Lin, G.; and Xie, L. 2021. Self-point-flow: Self-supervised scene flow estimation from point clouds with optimal transport and random walk. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15577–15586.

Li, R.; Sheng, W.; Zhu, F.; and Huang, J. 2018. Adaptive Graph Convolutional Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3546–3553.

Li, R.; Zhang, S.; and He, X. 2022. Sgtr: End-to-end scene graph generation with transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19486–19496.

Li, R.; Zhang, S.; Wan, B.; and He, X. 2021. Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11109–11119.

Lin, B.; Zhu, Y.; and Liang, X. 2022. Atom correlation based graph propagation for scene graph generation. *Pattern Recognition*, 122: 108300.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Lin, W.; and Chan, A. B. 2023. Optimal Transport Minimization: Crowd Localization on Density Maps for Semi-Supervised Counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21663–21673.

Liu, H.; and Singh, P. 2004. ConceptNet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4): 211–226.

Liu, Y.; Zhu, L.; Yamada, M.; and Yang, Y. 2020. Semantic correspondence as an optimal transport problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4463–4472.

Lu, C.; Krishna, R.; Bernstein, M.; and Fei-Fei, L. 2016. Visual relationship detection with language priors. In *European conference on computer vision*, 852–869. Springer.

Miller, G. A. 1995. WordNet: a lexical database for English. *Communications of the ACM*, 38(11): 39–41.

Sharifzadeh, S.; Baharlou, S. M.; Schmitt, M.; Schütze, H.; and Tresp, V. 2022. Improving Scene Graph Classification by Exploiting Knowledge from Texts. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 2189–2197.

Sharifzadeh, S.; Baharlou, S. M.; and Tresp, V. 2021. Classification by Attention: Scene Graph Classification with Prior Knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 5025–5033.

Spivak, D. I. 2014. *Category theory for the sciences*. MIT Press.

Sun, S.; Zhi, S.; Liao, Q.; Heikkilä, J.; and Liu, L. 2023. Unbiased Scene Graph Generation via Two-stage Causal Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

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

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

Wu, Y.; Miao, X.; Huang, X.; and Yin, J. 2023. Jointly imputing multi-view data with optimal transport. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 4747–4755.

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

Yang, X.; Liu, Y.; and Wang, X. 2022. Reformer: The relational transformer for image captioning. In *Proceedings of the 30th ACM International Conference on Multimedia*, 5398–5406.

Yao, Y.; Zhang, A.; Han, X.; Li, M.; Weber, C.; Liu, Z.; Wermter, S.; and Sun, M. 2021. Visual distant supervision for scene graph generation. In *ICCV*, 15816–15826.

Zareian, A.; Wang, Z.; You, H.; and Chang, S.-F. 2020. Learning visual commonsense for robust scene graph generation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*, 642–657. Springer.

Zareian, K. S. C. S.-F., Alireza. 2020. Bridging knowledge graphs to generate scene graphs. In *European Conference on Computer Vision*, 606–623. Springer.

Zhang, L.; Jin, L.; Sun, X.; Xu, G.; Zhang, Z.; Li, X.; Liu, N.; Liu, Q.; and Yan, S. 2023a. TOT: Topology-Aware Optimal Transport for Multimodal Hate Detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 4884–4892.

Zhang, Y.; Pan, Y.; Yao, T.; Huang, R.; Mei, T.; and Chen, C.-W. 2023b. Boosting scene graph generation with visual relation saliency. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(1): 1–17.

Zhang, Y.; Pan, Y.; Yao, T.; Huang, R.; Mei, T.; and Chen, C.-W. 2023c. Learning To Generate Language-Supervised and Open-Vocabulary Scene Graph Using Pre-Trained Visual-Semantic Space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2915–2924.

Zhao, N.; Chua, T.-S.; and Lee, G. H. 2021. Few-shot 3d point cloud semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8873–8882.

Zheng, C.; Lyu, X.; Gao, L.; Dai, B.; and Song, J. 2023a. Prototype-based Embedding Network for Scene Graph Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 22783–22792.

Zheng, W.; Yan, L.; Zhang, W.; and Wang, F.-Y. 2023b. Webly Supervised Knowledge-Embedded Model for Visual Reasoning. *IEEE Transactions on Neural Networks and Learning Systems*.

Zhong, Y.; Shi, J.; Yang, J.; Xu, C.; and Li, Y. 2021. Learning to generate scene graph from natural language supervision. In *ICCV*, 1823–1834.

Zhou, B.; Chen, Y.; Liu, K.; and Zhao, J. 2023. Event Process Typing via Hierarchical Optimal Transport. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 14038–14046.