Hierarchical Prompt Learning for Scene Graph Generation

Xuhan Zhu [1,2]
zhuxuhan21@mails.ucas.ac.cn
Yifei Xing [1,2]
xingyifei22@mails.ucas.ac.cn
Ruiping Wang [*] [2,3]
wangruiping@ict.ac.cn
Yaowei Wang [1]
wangyw@pcl.ac.cn
Xiangyuan Lan [*] [1]
lanxy@pcl.ac.cn

[1] Pengcheng Laboratory,
Shenzhen, China
[2] University of Chinese
Academy of Sciences
Beijing, China
[3] Institute of Computing Technology,
Chinese Academy of Sciences,
Beijing, China

Abstract

Scene Graph Generation (SGG) delivers structured knowledge representing complex scenes, which is applied in many computer vision fields. However, existing SGG models falter in predicting novel and informative predicates, undermining their applicability for higher-level visual tasks. Drawing inspiration from the success of prompt learning in zero-shot knowledge transfer, we propose a prompt-learning-based method to address novel and informative predicate learning challenges in SGG. Specifically, we perform a comprehensive analysis of three basic prompts in SGG, considering their computational efficiency and learning ability. Subsequently, we build upon these basic prompts to construct a Hierarchical Prompt (HP) learning method to enhance informative predicate learning. HP utilizes the composition of basic prompts constrained to progressively narrowed class groups and encourages the corresponding prompts to focus on the learning of increasingly informative predicates. HP is a plug-and-play solution applicable to various models. Extensive evaluations on SGG benchmarks demonstrate the excellent ability of HP to improve the performance of informative predicates across different baselines. We also introduce a novel predicate generalization task with a new benchmark. Experiments on it demonstrate the superiority of HP in base-to-novel predicate generalization.

1 Introduction

Scene graph generation (SGG) can effectively capture objects and their relationships within visual scenes by predicting triplets $(\texttt{subject}, \texttt{relation}, \texttt{object})$, which has been widely applied in complex visual scene understanding tasks, e.g., visual question answer [13, 37, 50], image caption [1, 10, 55], and image retrieval [17]. However, two challenges hinder the

[*] denotes the corresponding author.

application of SGG to downstream tasks: **1) Inability to predict novel predicates.** Traditional plain SGG models, based on the closed-set assumption, cannot predict novel predicates unseen during training. For example, the plain Motifs model [47] fails to predict the novel predicate `beneath` (c.f. Fig. 1 (a)) as `beneath` class is not encountered during training. **2) Biased prediction towards uninformative predicates.** In typical SGG datasets, annotators tend to annotate less-informative predicates [48], such that informative predicates are sparse and uninformative ones dominate. This poses challenges for learning informative predicates. For example, the plain Motifs model [47] tends to favor uninformative `on` (c.f. Fig. 1 (a)) rather than informative `laying on`. Based on these observations, this work explores methods to enhance the capacity of SGG models for novel and informative predicate learning.

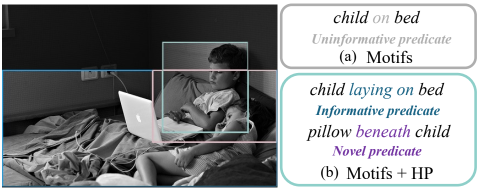

Figure 1: (a) The Motifs model [47] simply predicts **uninformative** predicate `on`. (b) Combined with the proposed HP method, the Motifs model can predict both **informative** (`laying on`) and **novel** (`beneath`) predicates.

We notice prompt learning has outstandingly addressed zero-shot learning challenges [39, 42, 58]. Different from traditional fixed-once-learned architecture like linear classifiers, which limit the ability of models to learn novel classes, prompt learning utilizes text embeddings as learning prototypes to align visual embeddings in a broader semantic space, enabling generalization from base classes to novel classes without requiring training data [7, 9, 56]. Inspired by this, we integrate prompt-learning-based methods into SGG models to solve the novel predicate learning challenge. However, plain prompt learning methods are not designed to target the biased prediction issue in SGG. Therefore, we propose an improved prompt learning method specifically tailored to the unique challenges in SGG.

We focus on the learning and gradient propagation processes of prompts used in SGG and extensively analyze the optimization gradients of a plain SGG model [47] (c.f. Fig. 2 (a)). We make two observations: 1) Negative gradients from the background class exceed the positive gradients of all foreground classes, undermining the learning of meaningful foreground classes. 2) Negative gradients from head classes surpass positive gradients in most tail classes, which further discourages the optimization of informative tail classes. Thus, it is crucial to mitigate the negative impact of the background class in order to shift the learning focus towards informative foreground classes. Similarly, to boost positive gradients for informative tail classes, it is essential to reduce negative gradients from head classes.

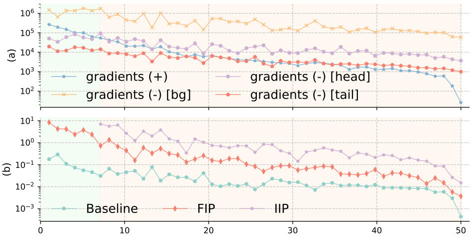

Figure 2: (a) Distribution of gradients, where gradients (+) and gradients (-) denote the positive and negative gradients, respectively. (b) Gradient ratios of positives to negatives for various methods. The x-axis indices are sorted with the sample counts (i.e., from head to tail classes). The light green span indicates the head group, and the light red span indicates the tail group. bg means the background class.

Expanding on the above insights, we propose a **Hierarchical Prompt (HP)** learning method. Through constraining progressively narrowed class groups, the associated prompts are encouraged to focus on the learning of increasingly informative predicates. Specifically, we first include a Base Prompt (BP), which contains training instances from all classes to reduce false positives from the background class. Next, we introduce the Foreground Informative Prompt (FIP), which includes only foreground class instances to ensure that foreground classes do not receive discouraging gradients from the background class. As depicted in Fig. 2 (b), it visibly increases the positive gradient ratio for all foreground classes. Finally, we introduce the Intra-tail Informative Prompt (IIP) to further reduce the class space to only contain tail-class instances. Consequently, negative gradients from head-class are reduced, as shown in Fig. 2 (b), resulting in a higher proportion of positive gradients for tail classes.

Moreover, to address the first challenge, previous works [12, 46] pretrain on datasets containing pseudo-labels of novel predicates derived from caption annotations, followed by prediction on novel predicates. Due to their reliance on supervised data from novel predicates, the experimental settings cannot impartially and objectively evaluate the generalization of base-to-novel predicates. Therefore, we propose a novel predicate generalization task with a new benchmark where models only have access to data from the base predicates during training and then directly evaluate the generalization of SGG models to novel predicates.

Our contributions include: 1) We are the first to conduct a comprehensive analysis of various basic prompts applied in the SGG task. 2) The proposed HP is the first prompt learning method that enables and improves the learning of novel and informative predicates. 3) We establish a predicate generalization task along with a new benchmark. Extensive experiments on it validate the robust transferability of HP to novel predicates. 4) The proposed HP greatly improves the performance of various baselines on informative predicates. When combined with debiasing models, it can outperform SOTA methods on popular SGG benchmarks.

2 Related Work

2.1 Scene Graph Generation

Current SGG research primarily focuses on building two types of models: **Plain models** construct more powerful feature encoders to extract relation features with different structures, including LSTM-based [47], tree-based [35, 40], transformer-based [6, 36], and graph-based [27, 44, 49] structures. **Debiasing models** are designed to address the unbalanced SGG learning issue. There are three common approaches to balanced SGG learning: 1) Rebalancing methods aim to balance the data [23] or loss ratios between different classes in training [18, 25, 26, 30, 43]. 2) Enhancing dataset methods focus on creating higher-quality training data by generating more informative labels for biased datasets [21, 46, 48]. 3) Post-probability processing methods adjust the biased prediction distribution, where some methods suggest removing the harmful bias from the good bias in training [36] or recovering the unbiased probabilities from the biased ones [2]. Though the debiasing models may perform well on informative predicates, there hasn't been any work on solving the biased predicate prediction based on prompt learning, which limits their ability for novel predicate learning.

2.2 Prompt Learning

Prompt learning originates from NLP [4, 31, 32], aiming to alleviate the reliance on extensive supervised data [28] when transferring pre-trained models to downstream tasks. This technique has recently been introduced to the CV field, including image classification [33, 56, 57], visual grounding [45], visual question answering [15], image captioning [15, 59], and zero-shot learning [39, 42, 58]. Recently, there have been several attempts at prompt-learning-based SGG. The naive method uses relation tokens as the prompt [33], but it neglects the object distinction of triplets. [16] build a triplet-specific prompt, but the enormous number of triplets occurring in complex scenarios result in extensive gains in computation time. [53] utilizes pre-trained open-vocabulary grounding models to help SGG models predict novel objects without considering novel predicates. [12, 46] design model-specific prompts, which cannot plug-and-play into other SGG models. Our proposed prompt learning method is model-agnostic, allowing easy application across different SGG models and scalability to other tasks such as long-tail classification and detection (c.f. *appendix*).

3 Method

Preliminary. Given an image, SGG aims to detect pairwise relationships, abbreviated as $\mathcal{G} = \{\mathcal{O}, \mathcal{E}\}$, where $\mathcal{O}$ is the set of object nodes and $\mathcal{E}$ represents the edges of $\mathcal{O}$. Each edge $(s, r, o) \in \mathcal{E}$ includes a subject node s and an object node o along with their relation $r \in R$, where R is the relation class set. Each node s or o consists of a bounding box and an object label obtained from the object detectors (e.g., Faster-RCNN [34] in previous works [36, 47]).
Relation Feature Extractor. The regional feature extractor obtains the regional features (v_s, v_o) of detected objects (s, o) using the ROI-Align [11] function (more details can be found in the *appendix*). Features (v_s, v_o) are then input into the object refined model, which consists of object and relation context encoders as used in [35, 36, 47], to get the refined object features $(\tilde{f}_s, \tilde{f}_o)$. Then, the relation features for final predictions are encoded as:

$$f_r = [\tilde{f}_s \oplus \tilde{f}_o] \circ f_u, \tag{1}$$

where $\oplus$ and $\circ$ denote concatenate and element-wise products, respectively, f_u denotes the union regional features of (s, o) [36], and the suffix r means the relation of the object nodes.

3.1 Prompt Learning for SGG

The proposed method is founded on the paradigm of prompt learning [14, 33], which involves a sequence of three steps: **Step 1:** construct prompt $\mathcal{T}$; **Step 2:** generate text embeddings $t = \mathcal{G}(\mathcal{T})$ using a large-scale language model $\mathcal{G}$ (e.g., language models from CLIP [33] or BERT [4]); **Step 3:** calculate the matched similarities of relation embeddings e_r and text embeddings t as $\mathcal{S}(t, e_r)$, where $\mathcal{S}(\cdot, \cdot)$ is the cosine similarity function, $e_r = L(f_r)$, where L denotes lightweight learnable projection layers designed to project e_r to the same representation space as text embeddings. Cosine similarities serve as the final logits z and are fed into the cross-entropy loss function during training and employed before the softmax function to generate the final posterior probabilities during inference.

3.1.1 Basic Prompts in SGG

Recognizing the significant impact of prompt variations on the performance of prompt learning [33, 57], we comprehensively discuss three types of basic prompts in SGG. Specifically, we compare them in terms of complexity and learning ability. Complexity refers to the time complexity of **Step 2**, which involves the computational time for obtaining text embeddings.
1) Relation Prompt (RP) uses the relation label words as the prompt that is formatted as:

$$\mathcal{T} = \{\mathtt{REL}_i\}_{0 \leq i < |R|}, \tag{2}$$

where REL are the label words of all relation classes. RP is relation-specific, focusing solely on the relationships and disregarding the information about object categories within the triplets. Because all triplets belonging to a relation $\texttt{REL}_i$ share a common prompt template, its computational complexity is relatively low, specifically $O(n)$=$O(|R|)$.

2) Discrete Triplet Prompt (DTP) employs the triplet words (e.g., `man standing on sidewalk`) as the prompt that is formatted as:

$$\mathcal{T} = \{[\texttt{SUB}_j, \texttt{REL}_i, \texttt{OBJ}_k]\}_{0\leq i<|R|}^{0\leq j<N_s, 0\leq k<N_o}, \tag{3}$$

where `SUB` and `OBJ` are the subject and object label words, and the number of object pair combinations is $N_p = N_s \times N_o$. DTP is specific to triplets; it not only focuses on relations but also incorporates object category information within the triplet. However, the involved computational complexity is $O(n) = O(N_p \times |R|)$, which is quadratic in nature, resulting in high computational costs and resource consumption. Though it is possible to partially improve the efficiency of DTP through engineering optimization to construct Efficient DTP (denoted as EDTP; more details can be found in *appendix*), it remains inefficient.

3) Continuous Triplet Prompt (CTP) is a type of continuous prompt [29, 56, 57], where all subject or object nodes within triplets sharing the same relation are associated with a collective learnable continuous vector. The formulation of CTP is:

$$\mathcal{T} = \{[V_i^s, \texttt{REL}_i, V_i^o]\}_{0\leq i<|R|}, \tag{4}$$

where V^s and V^o are learnable vectors corresponding subjects and objects in the triplets, and their dimension equals the word embeddings of `REL`. Similar to RP, its computational complexity is $O(n)$=$O(|R|)$. Compared to DTP, it only requires lower computational costs. Moreover, RP and DTP fall under discrete prompts [59] with fixed text embeddings, lacking the learnable ability. On the other hand, CTP is a continuous prompt that enhances the learning capability of prompts at a minimal cost in terms of parameter quantity.

3.2 Hierarchical Prompt Learning for SGG

Building upon the observations presented in Sec. 1, we introduce a hierarchical prompt composed of multiple basic prompts to progressively mitigate negative gradients stemming from various sources. As shown in Fig. 3, the construction steps of HP learning are as follows:

3.2.1 Hierarchical Grouping

We first formulate a hierarchical grouping method to prepare specific relation embeddings for each prompt in HP, which involves progressively constraining the category space from majority to minority. Upon obtaining relation embeddings as described in Sec. 3.1, we divide them into three groups. The first group, denoted as e_r, comprises relation embeddings of entire classes. The second group excludes background relation embeddings and encompasses all foreground relation embeddings, represented by $e_{r\in R_+}$, where R_+ denotes the foreground group. The last group further eliminates head-class relation embeddings, retaining only tail-class relation embeddings, denoted as $e_{r\in R_t}$, where R_t corresponds to the tail group.

3.2.2 Hierarchical Prompt

Base Prompt (BP) Firstly, we maintain a base prompt (denoted as $\mathcal{T}_{\text{bp}}$). This prompt accommodates relation embeddings from all classes, including the background class. This prompt is adopted to suppress false-positive predictions. The training loss for this prompt is:

$$\mathcal{L}_{\text{bp}}(e_{r\in R}) = -\log \frac{\exp(z_r/\tau)}{\sum_{j\neq r}\exp(z_j/\tau) + \exp(z_r/\tau)}, \tag{5}$$

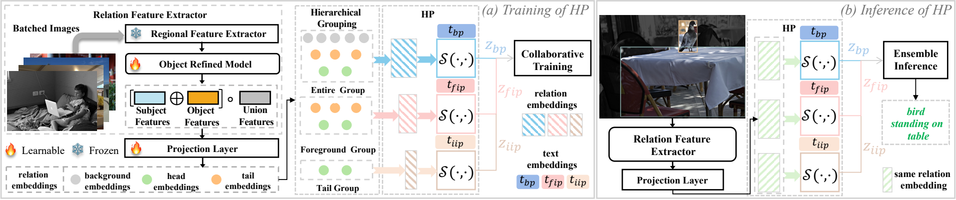

Figure 3: The training and inference pipeline of HP. During training, relation embeddings are extracted through the relation feature extractor and projection layer. Hierarchically grouped relation embeddings are sent to corresponding prompts in HP for collaborative training. During inference, the outputs from prompts in HP are ensembled to generate the final prediction.

where $z_r = \mathcal{S}(e_{r\in R}, t)$, $t = \mathcal{G}(\mathcal{T}_{\text{bp}})$, and τ is a temperature parameter.

Foreground Informative Prompt (FIP) Moving forward from BP, we introduce two hierarchically informative prompts. The first is a foreground informative prompt (denoted as $\mathcal{T}_{\text{fip}}$). It specifically eliminates relation embeddings from the background class, thereby mitigating the impact of background negative gradients on foreground classes. Its training loss is:

$$\mathcal{L}_{\text{fip}}(e_{r\in R_+}) = -\log \frac{\exp(z_r/\tau)}{\sum_{j\in R_+, j\neq r} \exp(z_j/\tau) + \exp(z_r/\tau)}. \tag{6}$$

Intra-tail Informative Prompt (IIP) However, simply addressing the adverse impact of the background on the foreground is insufficient. The suppression of tail class learning by head classes is also not to be overlooked, as discussed in Sec. 1. Therefore, following a similar design philosophy as FIP, we introduce an intra-tail informative prompt (denoted as $\mathcal{T}_{\text{iip}}$). In the IIP training process, only visual embeddings from the tail classes are retained. This ensures that it does not receive negative influence from head classes, reducing the significant inhibition of negative gradients from head classes. The training loss for IIP is:

$$\mathcal{L}_{\text{iip}}(e_{r\in R_t}) = -\log \frac{\exp(z_r/\tau)}{\sum_{j\in R_t, j\neq r} \exp(z_j/\tau) + \exp(z_r/\tau)}, \tag{7}$$

where $z_r = \mathcal{S}(e_{r\in R_t}, t)$, and $t_{r\in R_t} = \mathcal{G}(\mathcal{T}_{\text{iip}})$. As shown in Fig. 4, it can be observed that by incorporating IIP, the posterior probabilities of most tail classes have been further elevated compared to the FIP. More analysis about gradient and probability for HP is in the appendix.

HP is flexible in the selection of composed prompts, allowing BP, FIP, and IIP to be any of the three basic prompts discussed in Sec.3.1.1. Additional design details for informative prompts are presented in *appendix*.

Collaborative Training. We leverage a collaborative learning method to integrate the gradients derived from informative prompts, thereby supplying more positive gradients for the optimization of foreground and tail classes. The formalization is as follows:

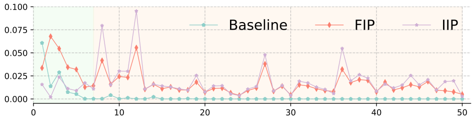

Figure 4: The posterior probabilities of head and tail classes.

$$\mathcal{L}_{\text{hp}}(e_r) = \omega\left(\frac{1}{N_+}\sum_{r\in R_+}\mathcal{L}_{\text{fip}} + \frac{1}{N_t}\sum_{r\in R_t}\mathcal{L}_{\text{iip}}\right) + \frac{1}{N}\sum_{r\in R}\mathcal{L}_{\text{bp}}, \tag{8}$$

where ω is a hyperparameter for uniform loss magnitudes. N_+, N_t, and N are the number of instances in foreground classes, tail classes, and all classes, respectively.

Ensemble Inference. We ensemble logits from FIP, IIP, and BP to integrate information for prediction. Specifically, the background logits from BP serve as the final background logits, while a weighted sum of logits from all prompts in HP is used as the final foreground logits:

$$\begin{cases} z^{+} = \alpha_1 z^{+}_{\text{fip}} + \alpha_2 z^{+}_{\text{iip}} + z^{+}_{\text{bp}} \\ z^{-} = z^{-}_{\text{bp}}. \end{cases} \tag{9}$$

Here, $z_{\text{bp}}{=}\mathcal{S}(e_r, \mathcal{G}(\mathcal{T}_{\text{bp}}))$. z_{fip} and z_{iip} are obtained by replacing $\mathcal{T}_{\text{bp}}$ with $\mathcal{T}_{\text{fip}}$ and $\mathcal{T}_{\text{iip}}$. $z^{+}{\in}\mathbb{R}^{B\times(R-1)}$ and $z^{-}{\in}\mathbb{R}^{B\times 1}$ are foreground and background logits. The final logits are $z{=}z^{-}\oplus z^{+}$. α_1 and α_2 are adjustment parameters. The predicted label $\hat{r}{=}\text{argmax}(\text{softmax}(z))$.

3.2.3 Novel Predicate Generalization

We introduce a task termed Novel Predicate Generalization, denoted as NPG. Specifically, all training samples are only from base classes, aiming to assess base-to-novel predicate generalization during evaluation. We tackle the NPG task using prompt-learning-based methods. The learning objective is to align relation embeddings and text embeddings of the base predicate prompt $\mathcal{T}_{\text{base}}$ in the language semantic space. The training loss for basic prompts is:

$$\mathcal{L}_{npg}(e_r) = \frac{1}{N} \sum_{r\in R_{base}} \mathcal{L}_{\text{CE}}\left(\text{softmax}(z/\tau), r\right), \tag{10}$$

where $z{=}\mathcal{S}(e_r, \mathcal{G}(\mathcal{T}_{\text{base}}))$, and $\mathcal{L}_{\text{CE}}$ is cross-entropy loss. The loss function for HP in the NPG task is the same as in Eq. 8. During inference, we input prompt $\mathcal{T}_{\text{novel}}$ for novel predicate prediction. The relation embeddings are matched with generated text embeddings of $\mathcal{T}_{\text{novel}}$ using cosine similarities. The predicted label with the highest score is calculated as follows:

$$\hat{r} = \text{argmax}(\text{softmax}(z)) = \text{argmax}(\text{softmax}(\mathcal{S}(e_r, \mathcal{G}(\mathcal{T}_{\text{novel}})))). \tag{11}$$

4 Experiments

Tasks, datasets and metrics. We will demonstrate the effectiveness and superiority of the proposed HP [1] in learning informative and novel predicates through evaluation on the following two tasks: **1) Informative predicate learning**. We conduct evaluation on two datasets, Visual Genome (**VG**) [19] and OpenImages (**OI**) [20], to assess the performance of informative predicates. VG dataset [19] has 57723 images for training, 5000 and 26446 images for validation and testing, and contains 150 object and 50 predicates categories [23, 36]. We report Recall@K (R@K) and mean Recall@K (mR@K) metrics similar to [5, 23, 36, 47]. Notably, R@K leans towards uninformative predicates, while mR@K tends to favor informative ones [21]. Thus, we include the metric MR@K [54] to balancely evaluate both types of predicates. OI is a large-scale dataset that contains two versions: V4 and V6. The V4 version has 53953 and 3234 images for the training and testing, with 7 object categories and 9 predicate categories. The V6 version has 126368 images used for training, 1813 and 5322 images for validation and testing, respectively, with 301 object categories and 31 predicate categories. We employ the same data preprocessing method and evaluation protocols as [20, 23, 52]. We report the mR@50, R@50, weighted mean AP of relationships (wmAP$_{rel}$), weighted mean AP of phrases (wmAP$_{phr}$) and weight metric score$_{wtd}$.
2) Novel predicate generalization. We designate VG [8] as the base training dataset, denoted as $\mathcal{D}_{base}$, and create a data split comprising 50 novel predicates not present in $\mathcal{D}_{base}$,

[1] Code of the model is available at https://github.com/ZHUXUHAN/HP.

forming the evaluation dataset. This split is derived from the large-scale SGG dataset VG-1800 [48], which contains 1807 predicate classes. We employ R@K, mR@K, and MR@K metrics to independently present base and novel predicate performance and report their respective Harmonic Mean (HM) [41, 56] to represent the overall performance.

Evaluation tasks. Our proposed methods are assessed on three tasks of SGG as follows: 1) Predicate Classification (PredCls): Given all ground-truth object categories and bounding boxes in an image, predict the relations of object pairs. 2) Scene Graph Classification (SGCls): Given the ground-truth bounding boxes of all objects, predict the categories of objects and the relations of object pairs. 3) Scene Graph Generation (SGDet): Given an image only, detect the bounding boxes and categories of all objects and predict their relationships.

Implementation details. Similar to [35, 36, 47], we adopt Faster R-CNN [34] as the object detector, whose model parameters are frozen during training. The pre-trained text encoder comes from CLIP-RN101 [33], which consists of masked self-attention transformers [38].

Moreover, in HP, the foreground categories are divided into two distinct groups according to the instance count in the training split. Specifically, it includes head classes (more than 10K) and tail classes (less than 10K) in VG; Because of the scarcity of sample categories in OI-V4 and OI-V6, we reduce the range of the head group (more than 100K), and the rare classes belong to the tail group. More implementation details are in the appendix.

4.1 Informative Predicate Learning.

Setting. We conduct two group experiments to verify the proposed method's effectiveness:

i) Comparison of various prompts. We compare several various prompts, including RP, DTP, and CTP, as well as HP-i and HP. Here, HP-i denotes the intermediate prompt format between BP and HP that uses only FIP in conjunction with BP, excluding IIP.

ii) Effectiveness on different baselines. Our proposed method is model-agnostic and functions as a plug-and-play module that integrates seamlessly with various baselines. The first type of baselines are plain models, such as Motifs [47] and VCTree [35]. The second type of baselines are debiasing models like Motifs (Rwt) [3, 47], VCTree (Rwt) [3, 35], and PeNet (Rwt) [3, 54].

	Method	O(n)	t(s)	R@100	mR@100	MR@100
DT	DTP	$O(N_p \times \lvert R \rvert)$	$+\infty$	-	-	-
	EDTP	$O(N_p \times \lvert R \rvert)$	2.24	-	-	-
	RP	$O(1 \times \lvert R \rvert)$	**0.99**	-	-	-
	CTP	$O(1 \times \lvert R \rvert)$	**0.99**	-	-	-
	HP-i	$O((N_+ + 2) \times \lvert R \rvert)$	1.01	-	-	-
	HP	$O((N_+ + N_t + 3) \times \lvert R \rvert)$	1.05	-	-	-
DI	DTP	$O(N_p \times \lvert R \rvert)$	$+\infty$	-	-	-
	EDTP	$O(N_p \times \lvert R \rvert)$	2.20	37.3	8.4	22.9
	RP	$O(1 \times \lvert R \rvert)$	**0.97**	37.1	9.0	23.1
	CTP	$O(1 \times \lvert R \rvert)$	**0.97**	37.5	9.1	23.3
	HP-i	$O(2 \times \lvert R \rvert)$	0.99	37.4	9.3	**23.4**
	HP	$O(3 \times \lvert R \rvert)$	1.03	36.3	**10.5**	**23.4**

Table 1: Prompt comparison in SGDet mode involves assessing computational complexity and running time per image ($t(s)$) during both training (DT) and inference (DI). $+\infty$ means the system runs out of memory.

i) Comparison of various prompts. We assess the computational efficiency of various prompts in SGDet mode, which involves more detected triplets compared to PredCls and SGCls modes. As shown in Tab. 1, the computational complexity of DTP is significantly high, even exceeding memory constraints, as there are a large number of triplets involved in computation in DTP. While EDTP reduces hardware resource consumption, its computational efficiency still falls far behind that of RP and CTP. CTP performs best in mR@K among the three basic prompts. Consequently, in informative predicate learning, we adopt CTP as the BP in HP-i and HP. As shown in Tab. 2, HP-i and HP demonstrate improved mR@K and MR@K than basic prompts. This improvement comes with slight increases in computational cost and time (c.f. Tab. 1). Notably, HP exhibits greater improvements than

Models	PredCls			SGCls			SGDet		
	R@50 / 100	mR@50 / 100	MR@50 / 100	R@50 / 100	mR@50 / 100	MR@50 / 100	R@50 / 100	mR@50 / 100	MR@50 / 100
Motifs [47]	65.2 / 67.0	14.8 / 16.1	40.0 / 41.6	38.9 / 39.8	8.3 / 8.8	23.6 / 24.3	32.8 / 37.2	6.8 / 7.9	19.8 / 22.6
+RP	65.7 / 67.4	16.7 / 18.0	41.2 / 42.7	38.9 / 39.7	9.2 / 9.7	24.1 / 24.7	32.8 / 37.1	7.6 / 9.0	20.2 / 23.1
+EDTP	65.0 / 66.9	16.3 / 17.8	40.7 / 42.4	38.6 / 39.6	9.0 / 9.6	23.8 / 24.6	32.9 / 37.3	7.1 / 8.4	20.0 / 22.9
+CTP	65.7 / 67.4	17.2 / 18.4	41.5 / 42.9	39.4 / 40.2	9.6 / 10.2	24.5 / 25.2	32.9 / 37.5	7.7 / 9.1	20.3 / 23.3
+HP-i	64.8 / 67.4	18.7 / 21.3	41.8 / 44.4	40.0 / 40.7	11.1 / 11.8	25.6 / 26.3	32.9 / 37.4	8.0 / 9.3	**20.5** / **23.4**
+HP	64.2 / 66.0	**24.1** / **25.8**	**44.2** / **45.9**	39.3 / 40.2	**13.5 / 14.3**	**26.4 / 27.3**	31.9 / 36.3	**8.9 / 10.5**	20.4 / **23.4**
VCTree [35]	65.4 / 67.2	16.7 / 18.2	41.1 / 42.7	46.7 / 47.6	11.8 / 12.5	29.3 / 30.1	31.9 / 36.2	7.4 / 8.7	19.7 / 22.5
+HP-i	65.1 / 66.9	20.4 / 22.1	42.8 / 44.5	45.7 / 46.8	13.6 / 14.6	29.7 / 30.7	31.9 / 36.2	7.9 / 9.6	19.9 / 22.9
+HP	63.3 / 65.2	**23.8 / 25.7**	**43.6 / 45.5**	46.2 / 47.2	**14.3** / **15.7**	**30.3** / **31.5**	30.8 / 35.1	**9.2** / **10.8**	**20.0** / **23.0**

Table 2: Performance comparison of different types of prompts on plain baselines on VG. The top-performing methods across all settings are underlined.

Models	PredCls			SGCls			SGDet		
	R@50 / 100	mR@50 / 100	MR@50 / 100	R@50 / 100	mR@50 / 100	MR@50 / 100	R@50 / 100	mR@50 / 100	MR@50 / 100
Motifs (IETrans) [illegible] *ECCV '22*	54.7 / 56.7	30.9 / 33.6	42.8 / 45.2	32.5 / 33.4	16.8 / 17.9	24.7 / 25.6	26.4 / 30.6	12.4 / 14.9	19.4 / 22.8
Motifs (GCL) [illegible] *CVPR '22*	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 (PKO) [22] *BMVC '22*	56.0 / 58.2	31.4 / 34.0	43.7 / 46.1	34.0 / 35.1	17.6 / 19.1	25.8 / 27.1	27.0 / 31.1	13.4 / 16.1	20.2 / 23.6
Motifs (CFA) [illegible] [illegible] *ICCV '23*	54.1 / 56.6	35.7 / 38.2	44.9 / **47.4**	34.9 / 36.1	17.0 / 18.4	26.0 / 27.3	27.4 / 31.8	13.2 / 15.5	20.3 / 23.7
Motifs (Rwt) [3, 54]	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
Motifs (Rwt) + HP-i	54.2 / 56.4	34.8 / 37.3	44.5 / 46.9	33.6 / 34.6	20.3 / 21.9	27.0 / 28.3	25.8 / 29.7	15.2 / 17.9	20.5 / 23.8
Motifs (Rwt) + HP	53.3 / 55.4	**37.3 / 39.3**	**45.3 / 47.4**	33.4 / 34.4	**21.6 / 22.7**	**27.5 / 28.6**	25.9 / 30.0	15.4 / 18.2	**20.7** / **24.1**
VCTree (IETrans) [illegible] *ECCV '22*	53.0 / 55.0	30.3 / 33.9	41.7 / 44.5	32.9 / 33.8	16.5 / 18.1	24.7 / 30.0	25.4 / 29.3	11.5 / 14.0	18.5 / 21.7
VCTree (GCL) [illegible] *CVPR '22*	40.7 / 42.7	37.1 / 39.1	38.9 / 40.9	27.7 / 28.7	22.5 / 23.5	25.1 / 26.1	17.4 / 20.7	15.2 / 17.5	16.3 / 19.1
VCTree (PKO) [22] *BMVC '22*	56.1 / 58.2	32.2 / 34.6	44.2 / 46.4	39.1 / 40.4	22.3 / 23.7	30.7 / 32.1	26.5 / 30.7	13.2 / 15.9	19.9 / 23.3
VCTree (CFA) [illegible] *ICCV '23*	54.7 / 57.5	34.5 / 37.2	44.6 / 47.4	42.4 / 43.5	19.1 / 20.8	30.8 / 32.2	27.1 / 31.2	13.1 / 15.5	**20.1** / 23.4
VCTree (Rwt) + HP	55.1 / 57.1	36.3 / 38.5	**45.7 / 47.8**	39.8 / 41.0	**26.1** / **27.6**	**33.6** / **34.3**	25.4 / 29.8	14.5 / 17.4	20.0 / **23.6**
PeNet (Rwt) [54]* *CVPR '22*	53.8 / 56.3	42.8 / 45.4	48.3 / 50.9	31.6 / 32.9	25.2 / 26.7	28.4 / 29.4	24.1 / 28.0	16.3 / 19.4	20.2 / 23.7
PeNet (Rwt) + HP	56.1 / 58.6	**43.2** / **46.0**	**49.7** / **52.3**	33.5 / 34.8	**25.4** / **27.2**	**29.5 / 31.0**	24.0 / 27.9	**16.7** / **19.8**	**20.4 / 23.9**

Table 3: Performance comparison of different types of debiasing models on VG. $*$ denotes we integrate the CTP module into the method based on our code framework.

HP-i in mR@K and MR@K, underscoring that a higher proportion of positive gradients for tail groups leads to more significant performance enhancements for informative predicates.

ii) Effectiveness on different baselines. Firstly, HP improves mR@K with minimal to no decrease in R@K when applied to plain baseline models, as seen in Tab. 2. For example, with the Motifs baseline, HP achieves enhancements ranging from 32.9% to 60.2% in mR@100 and 3.5% to 12.3% in MR@100. Secondly, as seen in Tab. 3, when applied to debiasing baseline models, both HP-i and HP outperform the baseline on mR@K and MR@K. Moreover, HP achieves comparable or better mR@K compared to other debiasing models, with its MR@K showing the most significant improvement among all models. Furthermore, we demonstrate the superiority of our methods on the OI dataset (c.f. Tab. 4). Both HP-i and HP outperform SOTA debiasing models only with plain baseline models. Upon incorporating the simple Rwt [3] debiasing method, HP shows a substantial boost in mR@K, surpassing the previous best model BGNN [23] across all metrics. Finally, we further validate additional baselines and present qualitative results (c.f. *appendix*), all confirming the effectiveness of HP in informative predicate learning.

D	**Models**	**mR@50**	**R@50**	**wmAP**		**score**wtd
				rel	**phr**	
V4	RelDN[51] *CVPR'19*	-	74.94	35.54	38.52	44.61
	GPS-Net[26] *CVPR'20*	-	77.27	38.78	40.15	47.03
	BGNN[23] ‡ *CVPR'21*	72.11	75.46	37.76	41.70	46.87
	Motifs + HP-i	73.43	**78.40**	**38.82**	**42.96**	**47.79**
	Motifs + HP	**73.86**	78.38	38.63	42.87	47.58
V6	Motifs[47] *CVPR'18*	32.68	71.63	29.91	31.59	38.93
	RelDN[51] *CVPR'19*	33.98	73.08	32.16	33.39	40.84
	VCTree[35] *CVPR'19*	33.91	74.08	34.16	33.11	40.21
	TDE[36] ‡ *CVPR'20*	35.47	69.30	30.74	32.80	39.27
	GPS-Net[26] ‡ *CVPR'20*	35.26	74.81	32.85	33.98	41.69
	BGNN[23] ‡ *CVPR'21*	40.45	74.98	33.51	34.15	42.06
	PeNet[54] *CVPR'23*	-	76.50	36.60	37.40	44.90
	Motifs + HP-i	37.97	76.50	**38.44**	**39.29**	**46.28**
	Motifs + HP	38.65	**76.51**	37.78	38.49	45.59
	Motifs (Rwt) + HP ‡	**41.46**	76.34	34.81	35.42	42.31

Table 4: Performance comparison on OI. Debiasing models are marked by ‡.

4.2 Novel Predicate Generalization.

Setting. We evaluate three SGG tasks: PredCls, SGCls, and SGDet. Different from the informative predicate learning task, the prior frequency bias [47] and the ensemble infer-

Split	Methods	PredCls			SGCls			SGDet		
		R@ 50 / 100	mR@ 50 / 100	MR@ 50 / 100	R@ 50 / 100	mR@ 50 / 100	MR@ 50 / 100	R@ 50 / 100	mR@ 50 / 100	MR@ 50 / 100
Base	RP	64.45 / 66.36	14.64 / 15.77	39.55 / 41.07	38.59 / 39.40	8.63 / 9.12	23.61 / 24.26	31.92 / 36.17	6.41 / 7.50	19.17 / 21.84
	HP-i	64.15 / 66.11	17.09 / 18.44	40.62 / 42.28	38.23 / 39.04	9.92 / 10.48	24.08 / 24.76	31.78 / 35.97	7.43 / 8.76	19.61 / 22.37
	HP	63.86 / 65.88	**19.43** / **21.07**	**41.65** / **43.48**	38.15 / 39.01	**10.64** / **11.39**	**24.40** / **25.20**	31.59 / 35.86	**7.76** / **9.17**	**19.68** / **22.52**
Novel	RP	11.10 / 11.17	5.88 / 5.90	8.49 / 8.54	6.86 / 6.86	3.97 / 3.97	5.42 / 5.42	5.41 / 5.83	3.14 / 3.48	4.28 / 4.66
	HP-i	12.65 / 12.68	7.12 / 7.13	9.89 / 9.91	6.92 / 6.99	4.40 / 4.42	5.66 / 5.71	6.28 / 7.23	5.24 / 5.86	5.76 / 6.55
	HP	13.35 / 13.39	**7.85** / **7.87**	**10.60** / **10.63**	7.21 / 7.28	**6.00** / **6.18**	**6.61** / **6.73**	6.89 / 7.73	**5.52** / **6.29**	**6.21** / **7.01**
HM	RP	18.94 / 19.12	8.39 / 8.59	13.98 / 14.13	11.65 / 11.69	5.44 / 5.53	8.81 / 8.85	9.25 / 10.04	4.22 / 4.75	6.99 / 7.67
	HP-i	21.13 / 21.28	10.05 / 10.28	15.90 / 16.05	11.72 / 11.86	6.10 / 6.22	9.17 / 9.27	10.49 / 12.04	6.15 / 7.02	8.90 / 10.13
	HP	22.08 / 22.26	**11.18** / **11.46**	**16.90** / **17.08**	12.13 / 12.27	**7.67** / **8.01**	**10.40** / **10.62**	11.31 / 12.72	**6.45** / **7.46**	**9.43** / **10.69**

Table 5: Performance comparison of various prompt-learning-based methods on the NPG task. HM means the harmonic mean metric [56]. The baseline model is Motifs [47].

ence method are not applicable in the NPG task. This is because specific prior knowledge associated with base predicates may negatively impact the performance of novel predicates.

As seen in Tab. 6, RP achieves the best mR@K and MR@K performance on the novel split among all three basic prompts, highlighting that EDTP and CTP exhibit weaker transferability to novel predicates, as their class-specific trait reduces the transferability. Therefore, we choose RP as the BP in HP-i and HP. Additionally, as shown in Tab. 5, HP-i and HP outperform RP on mR@K and MR@K in the novel split, indicating that better performance on informative predicates helps improve novel predicate prediction. Furthermore, HP outperforms HP-i on mR@K, suggesting that enhanced performance on more diverse and informative predicates benefits transferability to novel predicates. Finally, HP achieves the best performance on the HM metric across three tasks, demonstrating its superior performance on both base predicate learning and novel predicate generalization.

Split	Methods	PredCls		
		R@ 50 / 100	mR@ 50 / 100	MR@ 50 / 100
Base	RP	64.45 / 66.36	14.64 / 15.77	39.55 / 41.07
	EDTP	57.58 / 59.07	12.67 / 13.59	35.13 / 36.33
	CTP	64.51 / 66.37	**16.09 / 17.31**	**40.30 / 41.84**
Novel	RP	11.10 / 11.17	**5.88 / 5.90**	**8.49 / 8.54**
	EDTP	1.72 / 1.95	3.63 / 3.87	2.68 / 2.91
	CTP	1.17 / 1.17	1.37 / 1.37	1.27 / 1.27
HM	RP	18.94 / 19.12	**8.39 / 8.59**	**13.98 / 14.13**
	EDTP	3.34 / 3.78	5.64 / 6.02	4.97 / 5.39
	CTP	2.30 / 2.30	2.53 / 2.54	2.46 / 2.47

Table 6: Performance comparison of different basic prompts on the NPG task.

5 Conclusion

In this work, we propose a hierarchical prompt (HP) learning method to improve informative and novel predicate performance in SGG. HP leverages two forms of informative prompts to alleviate the detrimental effects of negative gradients, thereby enhancing foreground and tail predicate learning. Combined with various baselines, HP shows significant performance gains in informative predicates and achieves new SOTA results on mR@K and MR@K metrics. Furthermore, we establish a novel predicate generalization task with a new benchmark to assess the novel predicate performance. HP demonstrates optimal performance compared to other prompt-learning-based methods. Additionally, HP can be easily extended to other tasks, offering new inspiration for their advancement in balanced and zero-shot learning.

Acknowlegdgement

This work is supported by Pengcheng Laboratory Research Project No. PCL2023A08 and partially supported by the National Natural Science Foundation of China under contracts Nos. U21B2025 and 62402252.

References

[1] Shizhe Chen, Qin Jin, Peng Wang, and Qi Wu. Say as you wish: Fine-grained control of image caption generation with abstract scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9962–9971, 2020.

[2] Meng-Jiun Chiou, Henghui Ding, Hanshu Yan, Changhu Wang, Roger Zimmermann, and Jiashi Feng. Recovering the unbiased scene graphs from the biased ones. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1581—-1590, 2021.

[3] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[5] 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.

[6] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19427–19436, 2022.

[7] Yu Du, Fangyun Wei, Zihe Zhang, Miaojing Shi, Yue Gao, and Guoqi Li. Learning to prompt for open-vocabulary object detection with vision-language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14084–14093, 2022.

[8] D.Xu, Y. Zhu, C. Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *CVPR*, 2017.

[9] J Gu, Z Han, S Chen, et al. A systematic survey of prompt engineering on vision-language foundation models. arxiv. *arXiv preprint arXiv:2307.12980*, 2023.

[10] Jiuxiang Gu, Shafiq Joty, Jianfei Cai, Handong Zhao, Xu Yang, and Gang Wang. Unpaired image captioning via scene graph alignments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10323–10332, 2019.

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[12] Tao He, Lianli Gao, Jingkuan Song, and Yuan-Fang Li. Towards open-vocabulary scene graph generation with prompt-based finetuning. In *European Conference on Computer Vision*, pages 56–73. Springer, 2022.

[13] Marcel Hildebrandt, Hang Li, Rajat Koner, Volker Tresp, and Stephan Günnemann. Scene graph reasoning for visual question answering. *arXiv preprint arXiv:2007.01072*, 2020.

[14] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021.

[15] Woojeong Jin, Yu Cheng, Yelong Shen, Weizhu Chen, and Xiang Ren. A good prompt is worth millions of parameters? low-resource prompt-based learning for vision-language models. *arXiv preprint arXiv:2110.08484*, 2021.

[16] Ying Jin, Yinpeng Chen, Lijuan Wang, Jianfeng Wang, Pei Yu, Zicheng Liu, and Jenq-Neng Hwang. Is object detection necessary for human-object interaction recognition? *arXiv preprint arXiv:2107.13083*, 2021.

[17] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678, 2015.

[18] Boris Knyazev, Harm de Vries, Cătălina Cangea, Graham W Taylor, Aaron Courville, and Eugene Belilovsky. Graph density-aware losses for novel compositions in scene graph generation. In *BMVC*, 2020.

[19] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017.

[20] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision(IJCV)*, 2020.

[21] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18869–18878, 2022.

[22] Lin Li, Guikun Chen, Jun Xiao, Yi Yang, Chunping Wang, and Long Chen. Compositional feature augmentation for unbiased scene graph generation. *arXiv preprint arXiv:2308.06712*, 2023.

[23] R. Li, S. Zhang, B. Wan, and X. He. Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *CVPR*, 2021.

[24] Xingchen Li, Long Chen, Jian Shao, Shaoning Xiao, Songyang Zhang, and Jun Xiao. Rethinking the evaluation of unbiased scene graph generation. *arXiv preprint arXiv:2208.01909*, 2022.

[25] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. Oct 2017.

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

[27] Xin Lin, Changxing Ding, Yibing Zhan, Zijian Li, and Dacheng Tao. Hl-net: Heterophily learning network for scene graph generation. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19476–19485, 2022.

[28] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021.

[29] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. *arXiv preprint arXiv:2103.10385*, 2021.

[30] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19467–19475, 2022.

[31] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[32] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021.

[34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[35] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *CVPR*, 2019.

[36] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *CVPR*, 2020.

[37] Damien Teney, Lingqiao Liu, and Anton van Den Hengel. Graph-structured representations for visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2017.

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[39] Hualiang Wang, Yi Li, Huifeng Yao, and Xiaomeng Li. Clipn for zero-shot ood detection: Teaching clip to say no. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1802–1812, 2023.

[40] Wenbin Wang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Sketching image gist: Human-mimetic hierarchical scene graph generation. In *European conference on computer vision*, pages 222–239. Springer, 2020.

[41] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4582–4591, 2017.

[42] Guangyue Xu, Joyce Chai, and Parisa Kordjamshidi. Gipcol: Graph-injected soft prompting for compositional zero-shot learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5774–5783, 2024.

[43] 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.

[44] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *ECCV*, pages 670–685, 2018.

[45] Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Cpt: Colorful prompt tuning for pre-trained vision-language models. *arXiv preprint arXiv:2109.11797*, 2021.

[46] Qifan Yu, Juncheng Li, Yu Wu, Siliang Tang, Wei Ji, and Yueting Zhuang. Visually-prompted language model for fine-grained scene graph generation in an open world. *arXiv preprint arXiv:2303.13233*, 2023.

[47] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, 2018.

[48] Ao Zhang, Yuan Yao, Qianyu Chen, Wei Ji, Zhiyuan Liu, Maosong Sun, and Tat-Seng Chua. Fine-grained scene graph generation with data transfer. In *ECCV*, 2022.

[49] Ce Zhang, Simon Stepputtis, Joseph Campbell, Katia Sycara, and Yaqi Xie. Hiker-sgg: Hierarchical knowledge enhanced robust scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28233–28243, 2024.

[50] Cheng Zhang, Wei-Lun Chao, and Dong Xuan. An empirical study on leveraging scene graphs for visual question answering. *arXiv preprint arXiv:1907.12133*, 2019.

[51] J. Zhang, K. Shih, A. Elgammal, A. Tao, and B. Catanzaro. Graphical contrastive losses for scene graph parsing. In *CVPR*, 2019.

[52] Ji Zhang, Kevin J. Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. Graphical Contrastive Losses for Scene Graph Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2019.

[53] Yong Zhang, Yingwei Pan, Ting Yao, Rui Huang, Tao Mei, and Chang-Wen Chen. 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*, pages 2915–2924, 2023.

[54] Chaofan Zheng, Xinyu Lyu, Lianli Gao, Bo Dai, and Jingkuan Song. Prototype-based embedding network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22783–22792, 2023.

[55] Yiwu Zhong, Liwei Wang, Jianshu Chen, Dong Yu, and Yin Li. Comprehensive image captioning via scene graph decomposition. In *Proceedings of the European Conference on Computer Vision*, pages 211–229, 2020.

[56] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, pages 16816–16825, 2022.

[57] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

[58] Ziqin Zhou, Yinjie Lei, Bowen Zhang, Lingqiao Liu, and Yifan Liu. Zegclip: Towards adapting clip for zero-shot semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11175–11185, 2023.

[59] Peipei Zhu, Xiao Wang, Lin Zhu, Zhenglong Sun, Weishi Zheng, Yaowei Wang, and Changwen Chen. Prompt-based learning for unpaired image captioning. *arXiv preprint arXiv:2205.13125*, 2022.