

End-to-End Entity-Predicate Association Reasoning for Dynamic Scene Graph Generation

Liwei Wang¹, Yanduo Zhang^{1,2}✉, Tao Lu¹, Fang Liu¹,
Huiqin Zhang¹, Jiayi Ma³, Huabing Zhou¹✉

¹Hubei Key Laboratory of Intelligent Robot, Wuhan Institute of Technology

²Hubei University of Arts and Science ³Wuhan University

wlw951226@163.com, {ydzhang, lut}@wit.edu.cn,

{fangliu8822, 666zhanghuiqin, jyama2010, Zhouhuabing}@gmail.com

Abstract

Dynamic Scene Graph Generation (DSGG) aims to comprehensively understand videos by abstracting them into visual triplets $\langle \text{subject}, \text{predicate}, \text{object} \rangle$. Most existing methods focus on capturing temporal dependencies, but overlook crucial visual relationship dependencies between entities and predicates, as well as among predicate subclasses. These dependencies are essential for a deeper contextual understanding of scenarios. Additionally, current approaches do not support end-to-end training and instead rely on a two-stage pipeline, which incurs higher computational costs. To address these issues, we propose an end-to-end Association Reasoning Network (ARN) for DSGG. ARN leverages CLIP’s semantic priors to model fine-grained triplet cues to generate scene graph. In addition, we design a Predicate Association Parsing (PAP) module that employs a conditional weight mapping mechanism to structure entity and predicate representations. We further introduce a Hierarchical Attention (HA) mechanism to integrate spatio-temporal context with entity and predicate representations, enabling effective associative reasoning. Extensive experiments on the Action Genome dataset demonstrate significant performance improvements over existing methods. The source code is available in <https://github.com/wlw951226/ARN>.

1. Introduction

Scene graph generation is a promising approach to scene understanding and has emerged as a frontier topic. Dynamic Scene Graph Generation (DSGG) advances this by capturing dynamic visual relationships in videos, contributing to a comprehensive understanding of video content. DSGG has

✉ Corresponding authors.

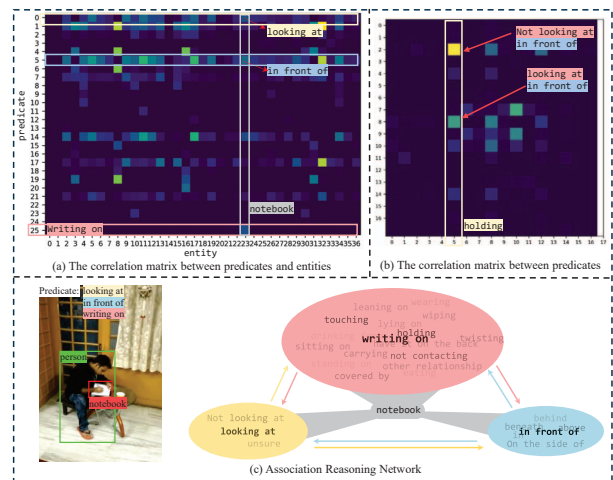


Figure 1. (a) The association frequency matrix between entities and predicates in the dataset [13]; (b) the association frequency matrix between predicate subclasses, where the vertical axis represents the pairwise combinations of two predicate subclasses; (c) an example and the pipeline of our associative reasoning network.

also opened up new possibilities in areas such as video analysis [9, 18, 42], video captioning [12, 38], and video question answering [5, 43], enabling computers to better simulate human perception and cognitive processes in real-world scenarios.

Dynamic scene graphs encode implicitly co-occurrences and affordance relations among entities and predicates. We analyze the frequency co-occurrence of predicates and entities in the Action Genome [13] dataset. As shown in Fig. 1(a), there is a strong association between certain predicates and entities; for instance, the predicate “writing on” often implies that the object is a notebook. Similarly, when the object is a notebook, it is likely in front of a person who is looking at it. Fig. 1(b) highlights the association be-

tween different predicates, and Fig. 1(c) provides an example demonstrating that holding an object strongly suggests its position in front of the person. However, existing methods overlook the dependencies between entities and predicates. Modeling these visual dependencies enables a multi-perspective understanding of scenes, resulting in more accurate scene graphs.

In addition, the currently prevalent DSGG paradigm [4, 6, 15, 29, 31, 36] are built upon a two-stage pipeline, focusing on capturing the temporal dependencies of videos. This two-stage paradigm independently optimizes the sub-tasks of entity localization [33] and predicate prediction, hindering end-to-end learning. Additionally, these methods generate redundant candidate boxes and predict relationships for all detected entity pairs, increasing computational cost.

Recently, the one-stage detector DETR [2] has transformed the pattern of traditional detection frameworks by adopting a set-based prediction approach for one-to-one matching, eliminating the need for computationally expensive hand-designed components such as non-maximum suppression. Meanwhile, DETR-based methods [3, 25, 30, 35, 41] have demonstrated remarkable performance in image understanding.

In this paper, we extend DETR to DSGG and propose an end-to-end Association Reasoning Network (ARN). ARN explores visual relation dependencies within scene graphs and performs association reasoning with fine-grained entity context. ARN leverages Contrastive Language-Image Pre-training (CLIP) [32] to bridge vision and language, modeling triplet cues through textual prompts, and extending predicate prediction to triplet prediction. This approach enriches the semantic representation of predicate prediction while ensuring semantic consistency between paired entities and predicates in DSGG.

We further design a Predicate Association Parsing (PAP) module to link fine-grained representations from paired entities and different predicate subclasses. It employs a conditional weight mapping mechanism to dynamically organize predicate representations across different branches. Additionally, we develop a Hierarchical Attention (HA) mechanism that integrates spatio-temporal context with complementary predicate and entity cues in a structured hierarchy. This method achieves association reasoning between entities and predicates while maintaining stable performance during end-to-end training. Our experiments on the widely used Action Genome dataset demonstrate that ARN achieves state-of-the-art performance in DSGG.

The contributions of this paper can be summarized as follows:

- We propose an end-to-end ARN model for DSGG, which explores visual relation dependencies between entities and predicates and performs association reasoning to gen-

erate accurate scene graphs.

- We introduce a Predicate Association Parsing (PAP) module to associate and organize fine-grained representations of different predicate subclasses and paired entities.
- We designed a Hierarchical Attention (HA) module to hierarchically perform visual relation association reasoning between paired entity representations and predicate relational context.
- We conducted a comprehensive evaluation of the proposed method, demonstrating its superior performance.

2. Related Work

Image scene graph generation. Image scene graph generation aims to identify entities and predict the predicates of image in form of triplets, which initially proposed by [14]. Most early works [8, 19, 23, 27, 44, 45], focused on refining entity and visual relationship representations in images using various network architectures. Subsequently, several works have attempted to design end-to-end image SGG models. Cong et al. [7] treated SGG as a set prediction problem and proposed an end-to-end image SGG model that directly predicts sparse scene graphs based solely on visual appearance. Furthermore, Li et al. [20] designed a graph assembling module that infers the connectivity of a bipartite graph based on entities features, generating scene graphs through graph matching. Meanwhile, some researchers [28] have focused on addressing the long-tailed distribution of predicates in benchmark datasets [16] to generate unbiased scene graphs.

Dynamic scene graph generation. The dynamic visual relationships in videos are more complex and have temporal dependencies, making DSGG more challenging compared to image scene graph generation. Previous works have focused on capturing temporal dependencies in videos and addressing data imbalance caused by long-tail distributions. Some works focus on designing more efficient temporal dependencies reasoning networks. Cong et al. [6] employed an encoder-decoder architecture to extract spatio-temporal context from video frames and proposed a two-stage baseline. Building on this, Pu et al. [31] injected prior spatio-temporal knowledge into DSGG via statistical co-occurrence, proposing a spatio-temporal Knowledge-Embedded Transformer. However, this method heavily rely on the inherent distribution of the data, such as the long-tail phenomenon. Differently, ARN does not rely on statistical priors.

To overcome the limitations of the data’s inherent distribution, Nag et al. [29] proposed a memory-guided training method to alleviate the long-tailed distribution. Lin et al. [24] introduced an asymmetric re-weighting loss that relieve the issue of label bias by adjusting the weights of tail predicates. Despite some progress, these methods sacrifice the performance of head predicates to focus on tail

predicates. Additionally, they overlook the visual relational dependencies within the scene graph, which encompass a broader understanding of interaction context.

The most relevant work to ours is OED [37], which introduces a progressive refined module to aggregate temporal context. But this module relies on independently optimized spatial context, resulting in degraded performance during end-to-end training. In contrast to OED, we focus on exploring visual dependencies between entities to enhance holistic scene understanding and enabling fully end-to-end implementation.

3. Method

In this work, we extend the end-to-end detector DETR[2] to DSGG and propose an Association Reasoning Network (ARN), as illustrated in Fig. 2. ARN sequentially extracts visual features and feeds them into two cascaded decoder modules, the Entity Decoder and the Predicate Decoder, to perform two subtasks: entity detection and triplet prediction (Sec 3.1). Subsequently, we introduce triplet cues modeling using CLIP’s semantic knowledge (Sec 3.2). Moreover, the Predicate Association Parsing (PAP) Module in predicate decoder organizes predicate cues from different branches and fine-grained paired entity features (Sec 3.3). The Hierarchical Attention (HA) mechanism then aggregates these with spatio-temporal features to perform association reasoning, enhancing broader contextual understanding of video scenes (Sec 3.4). Finally, we introduce the training loss and inference process of ARN (Sec 3.5).

3.1. Overall Architecture

The overall architecture of our ARN is illustrated in Fig. 2. The ARN is composed of a Backbone, an Encoder, an Entity decoder and a Predicate Decoder.

Backbone. CNN-based backbone maps an input frame I_i to a spatial feature map $\mathbf{f}_i \in \mathbb{R}^{H \times W \times C}$.

Encoder. The transformer-based encoder [2, 3, 30] refines image features and captures spatial contextual information. First, a 1×1 convolution and flatten operation are applied to reduce the dimension and transform it into a serialized feature representation $\mathbf{f}_i \in \mathbb{R}^{HW \times d}$. Next, we feed the \mathbf{f}_i and fixed positional encodings \mathbf{p}_e [1] to the encoder and output the enhanced spaital feature $\mathbf{F}_{vi} \in \mathbb{R}^{HW \times d}$.

Entity Decoder. Entity decoder consists of multiple transformer decoder layers and several FFN heads and introduces an entity query set $\mathbf{Q}^E \in \mathbb{R}^{N_q \times d}$, where N_q is the number of queries. Unlike previous methods, our query set has two subsets $\mathbf{Q}^E = [\mathbf{Q}^s, \mathbf{Q}^o] \in \mathbb{R}^{N_q \times d}$ that aim to separate the subject and the object.

Moreover, a pair of learnable entity position embeddings $\{\mathbf{e}^s, \mathbf{e}^o\} \in \mathbb{R}^{N_q \times d}$ is designed to assign the subject and object at the same position to form pairwise entities. Furthermore, \mathbf{e}^o is employed to align the query sets for the two

decoders and update them during training. Overall, the implementation of the entity decoder layer is as follows:

$$\mathbf{Q}^E = \text{Concat}[\mathbf{Q}^s + \mathbf{e}^s, \mathbf{Q}^o + \mathbf{e}^o], \quad (1)$$

$$\tilde{\mathbf{Q}}_{(l)}^E = \text{Dec}_{(l)}^E(\mathbf{Q}^E, \mathbf{F}_{vi}), \quad (2)$$

Specifically, at each layer $\text{Dec}_{(l)}^E$, the queries first undergoes self-attention to enable interactions among queries. Subsequently, cross-attention aggregates visual features \mathbf{F}_{vi} into the queries to obtain updated queries $\tilde{\mathbf{Q}}_{(l)}^E$. The output of the entity decoder, denoted as \mathbf{F}^E , is derived from the final updated queries $\tilde{\mathbf{Q}}_{(l)}^E$ after l layers.

Finally, the entity representation \mathbf{F}^E passes through the FFN head and object classifier at the top of the entity decoder to obtain the bounding box coordinates $\mathbf{B}^E = \{\mathbf{b}^s, \mathbf{b}^o\} \in \mathbb{R}^{N_q \times 4}$ and object categories $\mathbf{C}^o \in \mathbb{R}^{N_q}$ for the paired entities.

Predicate Decoder The predicate decoder is tailored to capture temporal dependencies and facilitate associative reasoning between entities and multiple predicate subclasses. In detail, the structure of the predicate decoder is a multi-branch decoder architecture, each branch corresponding to the prediction of a predicate subclass.

Each branch consists of multiple decoder layers and initializes predicate queries $\mathbf{Q}^{predi} \in \mathbb{R}^{N_q \times d}$ in i -th branch. Unlike the entity decoder, we capture temporal dependencies in visual features \mathbf{F}_{vi} . Specifically, we initialized a learnable frame encoding $\mathbf{p}_t \in \mathbb{R}^{N_q \times t \times d}$ to identify different frames. Subsequently, we transpose the \mathbf{F}_{vi} in the temporal dimension and update it by self-attention (SA) operation to obtain the spatio-temporal context feature \mathbf{F}_{vit} .

$$\mathbf{F}_{vit} = SA(\text{transpose}(\mathbf{F}_{vi})), \quad (3)$$

Ultimately, \mathbf{Q}^{predi} aggregates the \mathbf{F}_{vit} through cross-attention to generate fine-grained triplet representations \mathbf{F}^{predi} . Furthermore, to encourage associative reasoning across predicates, we design a predicate association embedding $\mathbf{e}^{predi} \in \mathbb{R}^{N_q \times d}$, which is dynamically updated along with the \mathbf{Q}^{predi} in the predicate decoder. The decoding process of the predicate representation in the l -th layer of predicate decoder Dec_l^{predi} is as follows:

$$\tilde{\mathbf{e}}^{predi} = \mathbf{e}^o + \mathbf{e}^{predi}, \quad (4)$$

$$\mathbf{F}_{(l)}^{predi} = \text{Dec}_{(l)}^{predi}(\mathbf{Q}^{predi}, \tilde{\mathbf{e}}^{predi}, \mathbf{F}_{vit}, \mathbf{p}_t), \quad (5)$$

3.2. Semantic Triplet Cue Modeling

We use CLIP [32] to extending traditional predicate prediction to triplet prediction and inject domain-agnostic prior knowledge into our ARN.

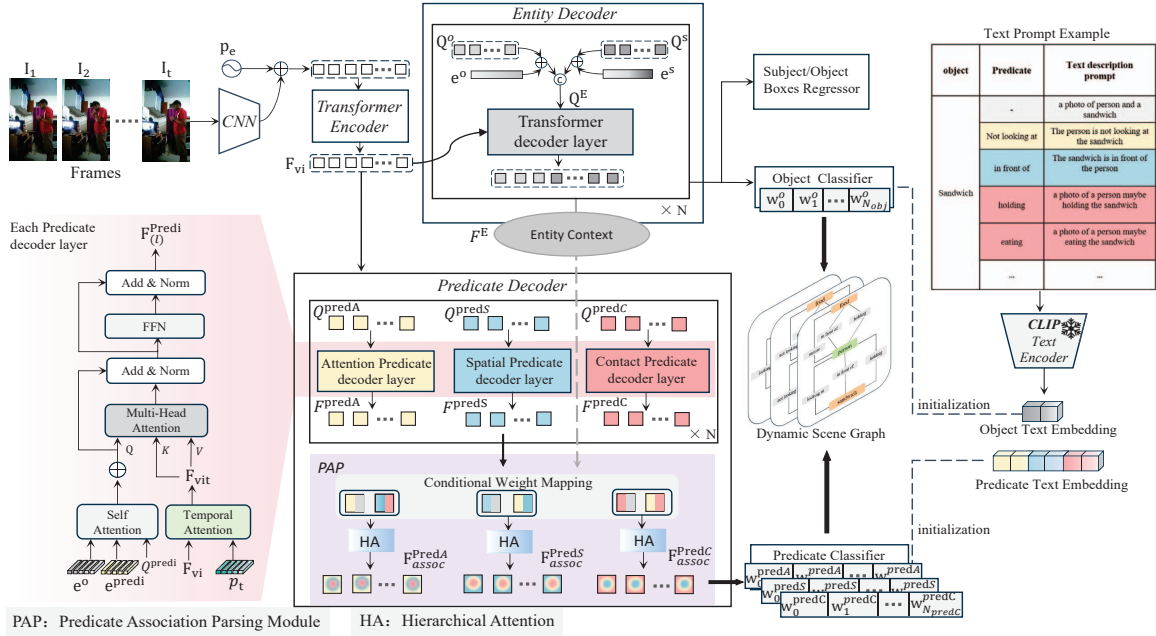


Figure 2. Illustration of the proposed Association Reasoning Network (ARN), which first extracts visual features from the video, and the entity decoder is used to locate entities. Subsequently, the multi-branch predicate decoder captures the temporal dependencies in visual features and infers representations for each predicate subclasses. Finally, the proposed Predicate Association Parsing (PAP) Module and Hierarchical Attention (HA) associates different predicate subclasses cues and aggregates fine-grained entity features to refine the prediction results.

Specifically, we follow the ActionGenome [13] setting and divide predicates into three subclasses: Attention, Spatial, and Contact. Then we design distinct textual prompts for each predicate subclass to build a bridge for interaction between CLIP and visual triplets: (1) ‘‘The person is [attention predicate] at the [object]’’; (2) ‘‘The [object] is [spatial predicate] the person’’; (3) ‘‘a photo of a person maybe [contacting predicate] the [object]’’, an example is illustrated on the right side of Fig. 2. Next, we use the CLIP text encoder to extract text embeddings $\mathbf{E}^{predi} \in \mathbb{R}^{N_{predi} \times d}$ for each predicate subclass triplets, where \mathbf{E}^{predi} serves as semantic cues for the visual triplets, N_{predi} denotes the number of triplet categories in i -th predicate decoder branch.

Based on this, we use the \mathbf{E}^{predi} to initialize the classifier weights w^{predi} . For the n -th query in i -th branch, it computes the similarity score s_n^i between the query representation and the triplet embeddings to align the visual and semantic information, then apply softmax activation to compute the focal loss [22], yielding the predicate prediction loss \mathcal{L}_{pred} .

Moreover, for the entity decoder, we design textual prompts for object categories ‘‘a photo of person and a [object]’’. We extract text embeddings $\mathbf{E}^o \in$

$\mathbb{R}^{N_{obj} \times d}$ in the same manner and use them to initialize the classifier weights w^o of the entity decoder. The object classifier is optimized using cross-entropy loss.

$$s_n^i = [\text{sim}(\mathbf{F}_n^{predi}, \mathbf{E}_1^{predi}), \dots, \text{sim}(\mathbf{F}_n^{predi}, \mathbf{E}_{N_{predi}}^{predi})] \quad (6)$$

$$\mathcal{L}_{pred} = \frac{1}{N} \sum_{n \in N_q} \mathcal{L}_{focal}(\sigma(s_n^i), y_r^{predi}), \quad (7)$$

3.3. Predicate Association Parsing Module

As mentioned previously, the pairwise entities can be associated with multiple predicates, and these entity-predicate associative cues are crucial for DSGG. Therefore, we propose a Predicate Association Parsing (PAP) Module that first leverages a conditional weight mapping mechanism to differentiate and organize multiple predicate subclass cues, then performs associative reasoning by aggregating global spatio-temporal context and fine-grained entity representations.

We denote the outputs of different branches in the predicate decoder as \mathbf{F}^{predA} , \mathbf{F}^{predS} , and \mathbf{F}^{predC} , using the initials of predicate subclasses for simplicity. For \mathbf{F}^{predA} , we treat \mathbf{F}^{predS} and \mathbf{F}^{predC} as complementary predicate

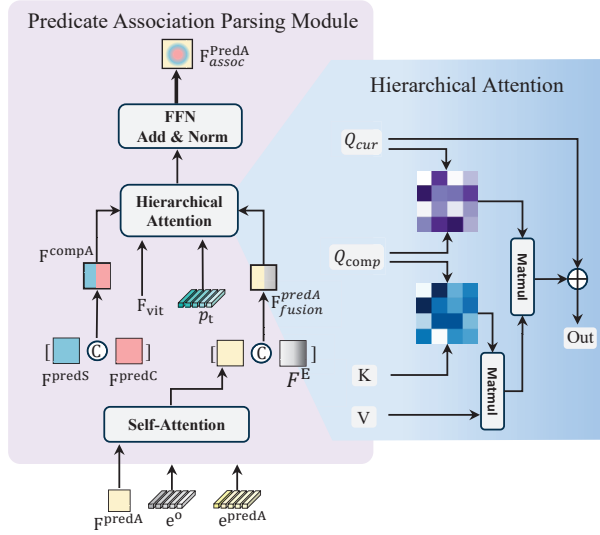


Figure 3. The architecture of the proposed Predicate Association Parsing (PAP) Module and Hierarchical Attention (HA) Mechanism. The PAP organizes complementary predicate cues and embeds entity context into the current predicate representation. Building on this, the HA aggregates spatio-temporal context, explores visual relation dependencies, and progressively refines the predicate representation.

representations and use a linear layer to reduce the dimensionality of them. Then we concatenate them as complementary predicate cues for the current branch, denoted as $\mathbf{F}^{compA} = \text{Concat}[\mathbf{F}^{predS}, \mathbf{F}^{predC}]$. During the dot product operation, the cross-attention weights consist of two components, balancing the contributions of different predicate subclasses and preventing the model from being biased toward any single predicate representation.

$$\mathbf{Q} = (\mathbf{Q}_S, \mathbf{Q}_C) = (\mathbf{F}^{predS}, \mathbf{F}^{predC}), \mathbf{K} = \mathbf{F}^{vit} + \mathbf{P}_t, \quad (8)$$

$$\mathbf{Q}^T \mathbf{K} = \mathbf{Q}_S^T \mathbf{K} + \mathbf{Q}_C^T \mathbf{K}, \quad (9)$$

Next, we integrate the fine-grained paired entity features \mathbf{F}^E into \mathbf{F}^{predA} .

$$\mathbf{F}_{fusion}^{predA} = \text{MLP}(\text{concat}[\mathbf{F}^{predA} + \mathbf{e}^{predA}, \mathbf{F}^E]), \quad (10)$$

where \mathbf{e}^{predA} denotes the predicate association embedding for attention predicates. Finally, we perform associative reasoning between the entity representation and various predicate representations through the proposed hierarchical attention module.

3.4. Hierarchical Attention

Visual relational dependencies are crucial for understanding scene context, yet traditional cross attention mechanisms

struggle to associate multiple semantic representations. To this end, we designed a Hierarchical Attention (HA) Mechanism, building upon the predicate association parsing module. The HA mechanism enables fine-grained integration of entity features with diverse predicate representations, facilitating associative reasoning and improving the prediction of tail predicates. The concept is illustrated in Fig. 3.

Specifically, HA is a quadruplet attention paradigm $(\mathbf{Q}_{cur}, \mathbf{Q}_{comp}, \mathbf{K}, \mathbf{V})$, where \mathbf{Q}_{cur} is the predicate subclass representation of the current branch, and \mathbf{Q}_{comp} represents a complementary predicate cues, which is the cues representation of the other predicate subclasses.

Taking the branch of the attention predicate subclass as an example, we perform the first-level cross-attention operation to associate complementary predicate cues \mathbf{F}^{compA} , where \mathbf{F}^{compA} serves as the query \mathbf{Q}_{comp} , and \mathbf{F}^{vit} is used as the key and value, \mathbf{e}^o as the query pos. This process highlights regions of interest related to other predicate subclasses and updates \mathbf{Q}_{comp} to obtain \mathbf{F}_{comp} .

$$\mathbf{F}_{comp} = \text{Att}_1^{HA}(\mathbf{Q}_{comp}, \mathbf{K} = \mathbf{F}^{vit} + \mathbf{p}_t, \mathbf{V} = \mathbf{F}^{vit}), \quad (11)$$

Second, we perform the second-level cross-attention operation, where \mathbf{F}^{predA} serves as the query \mathbf{Q}_{cur} , \mathbf{Q}_{comp} as the key and \mathbf{F}_{comp} as the value. This operation first establishes the association between the current predicate subclass and complementary predicate cues, then propagates the spatio-temporal visual features to the query of the current subclass, highlighting the most relevant categories and updating the predicate representation \mathbf{F}^{predA} . Finally, the \mathbf{F}^{predA} passes through the linear mapping layer and is added to the original input \mathbf{Q}_{cur} , yielding the final output $\mathbf{F}_{assoc}^{predA}$.

$$\mathbf{F}^{predA} = \text{Att}_2^{HA}(\mathbf{Q}_{cur} = \mathbf{F}^{predA}, \mathbf{Q}_{comp}, \mathbf{F}_{sup}), \quad (12)$$

$$\mathbf{F}_{assoc}^{predA} = \text{Linear}(\mathbf{F}^{predA}) + \mathbf{F}^{predA}. \quad (13)$$

3.5. Training and Inference

Training. We compute the matching cost using the Hungarian matching algorithm [17], thereby obtaining the query indices that best match the ground truth within the query set. Then we compute the loss between the corresponding queries and the ground truth across three aspects: object category loss \mathcal{L}_{obj} , paired entity bounding box loss \mathcal{L}_{box} and predicate category loss \mathcal{L}_{pred} . The first two aspects follow the DETR [2] framework. Besides, we follow [11] to apply an \mathcal{L}_1 loss between the predicate and image embeddings to align the visual and textual features. The overall loss function is formulated as:

$$\mathcal{L}_{total} = \lambda_{obj} \mathcal{L}_{obj} + \lambda_{box} \mathcal{L}_{box} + \lambda_{pred} \mathcal{L}_{pred} + \lambda_{l_1} \mathcal{L}_1, \quad (14)$$

where λ_{obj} , λ_{box} , λ_{pred} and λ_{l_1} are hyper-parameters.

Inference. During inference, the score of object category s^o is simply the top-1 score from the softmax distribution over objects, and the triplet score $(s^{triA}, s^{triS}, s^{triC})$ of each predicate subclass is obtained using sigmoid function. Subsequently, we calculate the predicate score. For the i -th query and the j -th predicate class, we take the maximum score among the triplet scores belonging to the predicate subclasses associated with that predicate class to obtain the score for the j -th predicate class.

$$s_{ij}^{predA} = \text{Max}\{s_{i1}^{triA}, s_{i2}^{triA}, \dots, s_{ik}^{triA}\}, \quad (15)$$

where $1, 2, \dots, k$ are the score indices in the triplet scores of the j -th query. Finally, we compute $s_i^o * s_{ij}^{predA}$ as the final score for the j -th predicate of the i -th query to generate the scene graph.

4. Experiments

4.1. Experimental Setting

Dataset. We evaluated our method on the Action Genome dataset [13], which is a large-scale video scene graph dataset built on the Charades dataset [34]. This dataset covers 234,253 frames from 9,848 videos, with a total of 476,229 object bounding boxes and 1,715,568 relationship entities. There are 36 entity categories and 26 predicate categories in these annotations. Predicates are further divided into three categories: attention, spatial, and contacting.

Evaluation Metrics. We evaluated the Scene Graph Detection (SGDet) and Predicate Classification (PredCls) task on the AG dataset. SGDet aims to detect entities and predict predicates without labels. PredCls is a relatively simple task, focused on predicting predicates based on given oracle detection results. We primarily evaluate the performance of SGDet task. Following previous works, we adopt Recall@k and mean Recall@K as evaluation metric to measure the ratio of correct entities among the top K predicted entities with the highest confidence (K=10, 20, 50). To make a sufficient comparison with the baseline, the experiment was conducted under two settings: With Constraint and No Constraints. With Constraint only selects one predicate for each entity pair. No Constraints allows multiple predictions of the relationship between each entity pair.

Training Details. We employ ResNet-50 as the CNN backbone and The number of layers for the Transformer encoder and decoder are set to 6 and 3, respectively. For training our model, we initialize the network with the parameters of DETR trained with the COCO dataset. ARN train for 20 epochs using the AdamW optimizer [26] with the batch size 8, initial learning rate of $1e^{-4}$ of the backbone. For each decoder, we set the number of queries $N_q = 100$ and use fine-tune the CLIP text embeddings initialized classifier with a small learning rate of $1e^{-5}$. Moreover, the last layer

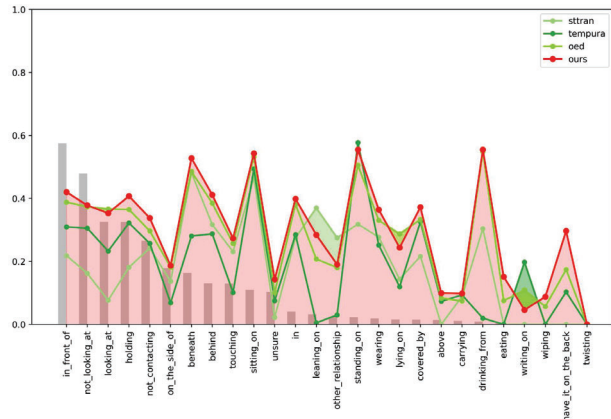


Figure 4. Comparative per predicate class performance for SGDET task. Results are in terms of mR@10 under “No constraint” setup. Predicates are arranged from left to right in order from head to tail.

of the predicate decoder is our proposed predicate association parsing module. The hyper-parameters for the Hungarian costs λ_{obj} , λ_{pred} , λ_{box} , λ_{l_1} , and the loss weights of them are set to 1, 2, 2.5, 20, respectively. We conduct our experiments on one NVIDIA 4090 GPU.

4.2. Comparison to state-of-the-art

To evaluate the performance of our model, we compare it with several state-of-the-art DSGG approaches [6, 10, 21, 23, 27, 29, 36, 37, 39, 40]. Tab. 1 shows the comparative results for SGDET task in terms of R@K under both **No constraints** and **With Constraint** settings.

In SGDet task, our method outperforms all existing two stage and one stage methods, with ARN improves the performance of the second-best method by 2.3%, 2.4%, and 2.3% under R@K metric of **No constraints** setting, respectively. Under the **With Constraint** setting, ARN improves performance by 1.6% on R@10 and 0.9% on R@20 compared to the current best method, however, it is slightly lower than the current best method on R@50. We conjecture the reason is as follows: In triplet prediction, ARN may generate more semantically similar result, and under the **With Constraints** setting, redundant results may occupy the top-ranking positions, thereby affecting the results in this setting. However, the results become reliable with fewer predictions when $K = [10, 20]$.

We evaluated the mR@K metric for each predicate category under **No constraints** setting, selecting several representative methods for comparison, as shown in Fig. 4. Our method achieves the best performance across most predicate categories, with varying shades of green indicate instances where other methods slightly outperform ours in a few categories. This demonstrates that our proposed ARN

Method	With Constraint					No Constraint						
	R@10	R@20	R@50	mR@10	mR@20	mR@50	R@10	R@20	R@50	mR@10	mR@20	mR@50
Two Stage Method												
VRD[27]	19.2	24.5	26.0	-	-	-	19.1	28.8	40.5	-	-	-
GPS-Net[23]	24.7	33.1	35.1	-	-	-	24.4	35.7	47.3	-	-	-
TRACE[36]	13.9	14.5	14.5	8.2	8.2	8.2	26.5	35.6	45.3	22.8	31.3	41.8
STTran[6]	25.2	34.1	37.0	16.6	20.8	22.2	24.6	36.2	48.8	20.9	29.7	39.2
APT[21]	26.3	36.1	38.3	-	-	-	25.7	37.9	50.1	-	-	-
STTran-TPI[40]	26.2	34.6	37.4	15.6	20.2	21.8	-	-	-	-	-	-
TR ² [39]	26.8	35.5	38.3	-	-	-	27.8	39.2	50.0	-	-	-
TEMPURA[29]	28.1	33.4	34.9	18.5	22.6	23.7	29.8	38.1	46.4	24.7	33.9	43.7
TD ² [24]	28.7	-	37.1	20.4	-	26.1	30.5	-	49.3	<u>27.9</u>	-	46.3
DSG-DETR[10]	30.3	34.8	36.1	16.4	19.3	20.1	32.1	40.9	48.3	21.3	28.1	-
OED[37]	<u>33.5</u>	<u>40.9</u>	48.9	<u>21.5</u>	<u>27.4</u>	<u>33.4</u>	<u>35.3</u>	<u>44.4</u>	<u>51.8</u>	<u>27.2</u>	<u>40.5</u>	<u>50.4</u>
One Stage Method												
RelTR[7]	19.7	23.4	25.9	-	-	-	20.9	24.6	28.2	-	-	-
OED*[37]	31.5	37.7	43.7	20.6	26.1	30.2	33.4	41.3	49.0	26.1	38.5	47.7
Ours	35.1	41.8	<u>47.2</u>	23.2	29.0	33.6	37.6	46.8	54.1	29.7	41.9	51.9

Table 1. Comparative results for SGDet task, on Action Genome dataset [13], in terms of Recall@K and mean Recall@K metric, best and second best results under each setting are respectively marked in bold and underline. where * indicates end-to-end training results.

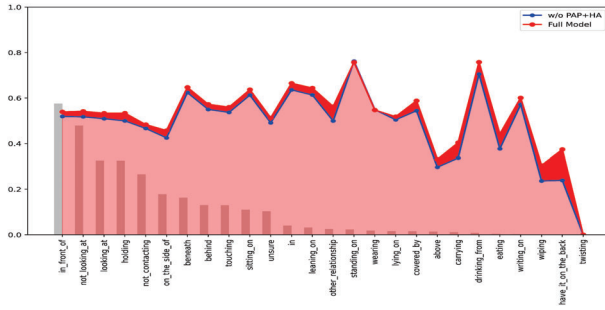


Figure 5. Ablation results for per class performance in SGDET task. Results are in terms of mR@50 under “No constraint”.

Method	With Constraint			No Constraint			FPS	#Params(M)
	R@10	R@20	R@50	R@10	R@20	R@50		
TRACE[36]	27.5	27.5	27.5	72.6	91.6	96.4	-	-
STTran[6]	68.6	71.8	71.8	77.9	94.2	99.1	-	151.03
TEMPURA[29]	68.8	71.5	71.5	80.4	94.2	<u>99.4</u>	-	248.8
TD ² [24]	70.1	-	73.1	81.7	-	98.8	-	-
OED[37]	73.0	76.1	76.1	<u>83.3</u>	<u>95.3</u>	99.2	121	54.54
Ours	<u>74.6</u>	<u>74.6</u>	<u>74.6</u>	83.7	96.7	99.9	162	54.06

Table 2. Comparison results for PredCls task, on Action Genome dataset [13], in terms of Recall@K metric. Best and second best results under each setting are respectively marked in bold and underline.

framework effectively mitigates the long-tail problem, improving accuracy on tail predicates while maintaining performance on head predicates.

In PredCls task, as shown in Tab. 2, ARN improves the performance of the second-best methods by 0.4%, 1.4%, 0.5% on the R@K metric under the **No Constraints** setting. It also achieves comparable performance under the **With Constraints** setting. Although ARN is designed for SGDet task, its performance on the PredCls task also demonstrates

its versatility across multiple tasks.

Additionally, as shown in Tab. 2, our model exhibits a significant advantage in parameter efficiency compared to two-stage methods [6, 29]. In terms of inference speed, it surpasses the end-to-end algorithm OED. Notably, OED relies on adjacent frames during inference, requiring the processing of more images. Our experiments show that our method achieves an inference time of 8 minutes per round, whereas OED requires 22.7 minutes per round.

4.3. Ablative Study

We proposed two modules in our ARN, Predicate Association Parsing Module (PAP) and Hierarchical Attention (HA), and performed ablation studies on different components to clarify their contributions to performance.

Association Parsing Network. We evaluate the effectiveness of several components in the proposed ARN framework. We extend DETR for DSGG and establish it as our baseline. We choose focal loss as the loss function for predicate prediction and compare it with cross-entropy loss for fairness, as shown in the first and second rows of Tab. 3. The results indicate that focal loss is better suited for DSGG. We leverage CLIP to extend predicate prediction to triplet prediction, using CLIP’s semantic knowledge to guide the model in learning more robust triplet semantic context. As demonstrated in the third row of Tab. 3, this is an effective approach for enhancing scene understanding.

Predicate Association Parsing Module. We examine the impact of the proposed Predicate Association Parsing Module on model performance, as shown in the fourth row of Tab. 3. Without incorporating HA, associating fine-grained paired entity features enhances the model’s performance, demonstrating the positive role of visual relation

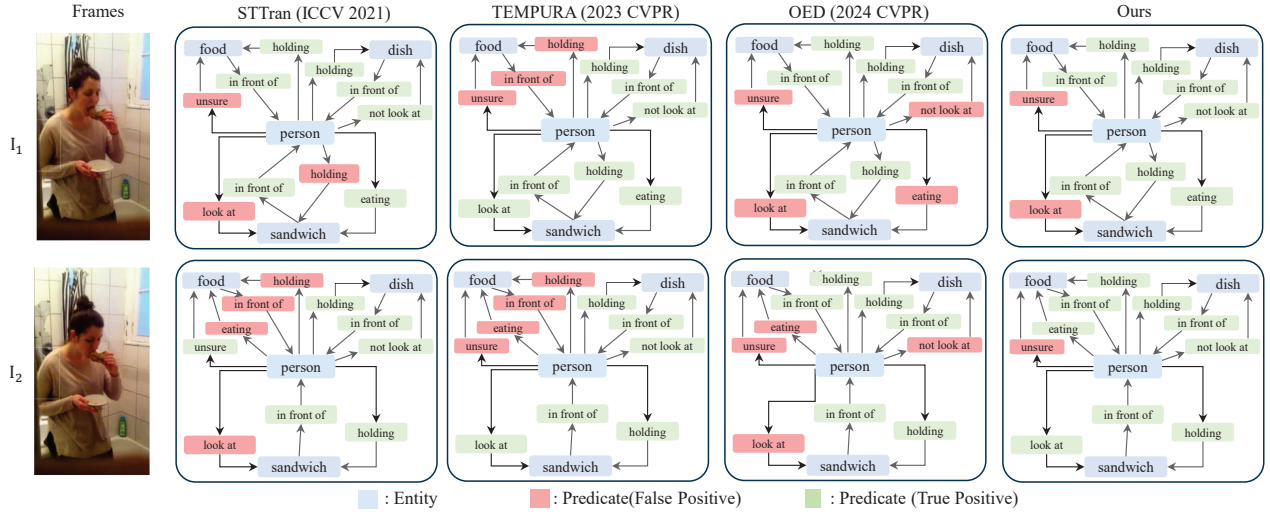


Figure 6. **Comparative qualitative results.** From left to right: input video frames, dynamic scene graphs generated by STTran [6], TEMPURA [29], OED [37] and ours. Incorrect and correct predicate predictions are shown in green and red, respectively.

Baseline	\mathcal{L}_{focal}	CLIP	PAP	HA	No Constraint			
					R@10	R@50	mR@10	mR@50
✓					28.4	44.3	19.6	39.3
✓	✓				31.7	45.4	25.6	44.8
✓	✓	✓			34.4	51.7	27.1	48.5
✓	✓	✓	✓		35.3	52.2	27.9	49.8
✓	✓	✓	✓	✓	37.6	54.1	29.7	51.9

Table 3. Ablative studies on our framework for SGDet task.

Method	No Constraint			
	R@10	R@50	mR@10	mR@50
Model I	35.0	51.5	27.3	48.9
Model II	36.9	53.0	29.3	51.4
Ours	37.6	54.1	29.7	51.9

Table 4. Ablative studies on Conditional weight mapping.

dependencies between entities and predicates in predicate prediction. With **Hierarchical Attention** included, the results further improve (fifth row of Tab. 3), indicating that visual relation dependencies between predicate subclasses also facilitate a broader understanding of interactive context, validating our motivation.

Conditional weight mapping. We evaluate the effectiveness of the proposed conditional weight mapping mechanism for association reasoning. Following the cross-attention structure, we directly add paired entity representations to the predicate query, denoted as **Model I**. Additionally, we experiment with handling features from different sources through nonlinear mapping, denoted as **Model II**. As observed in the first and second rows of Tab. 4, introducing entity information without proper organization negatively impacts the original performance of ARN. Nonlinear

mapping yields some improvement, but conditional weight mapping provides a more elegant approach to achieving effective association reasoning.

More analyze. To further analyze the impact of the proposed modules, we present the average recall for each predicate class in the SGDet task, as shown in Fig. 5. It can be observed that after removing the PAP and HA modules, the model’s performance drops in 92% of predicate categories, with only slight improvements in the “standing on” and “wearing” categories (0.48% and 0.17%, respectively).

4.4. Qualitative Results

Fig. 6 presents the qualitative results in SGDet task. The results show that existing two-stage methods (i.e., STTran, TEMPURA) are clearly at a disadvantage in entity detection (erroneous detections result in incorrect triplets). In contrast, one-stage methods excel in entity detection. Additionally, ARN accurately detects more tail predicates, demonstrating a broader capability for scene understanding.

5. Conclusion

In this paper, we propose an end-to-end Association Reasoning Network (ARN) for DSGG. ARN is the first to model reasoning over visual relation dependencies. We extend predicate prediction to triplet prediction and introduce a Predicate Association Parsing Module (PAP) to organize and interpret multiple predicate subclasses and fine-grained entity representations. Additionally, we design a Hierarchical Attention (HA) mechanism to effectively aggregate multi-source information for association reasoning. Extensive experiments on the Action Genome dataset demonstrate the effectiveness of our method.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China (62171328,62171327,62072350); in part by the Key Program of Hubei Provincial Natural Science Foundation (Xiangyang Innovation and Development Joint Fund) under Grand (2025AFD050); by the Central Government’s Special Project for Guiding Local Development in Hubei Province (2022BGE242); by the Hubei Science and technology innovation team (T2023009); by the National Natural Science Foundation of Hubei (2023AFB158).

References

- [1] Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V Le. Attention augmented convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3286–3295, 2019. 3
- [2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, pages 213–229, 2020. 2, 3, 5
- [3] Mingfei Chen, Yue Liao, Si Liu, Zhiyuan Chen, Fei Wang, and Chen Qian. Reformulating hoi detection as adaptive set prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9004–9013, 2021. 2, 3
- [4] Siqi Chen, Jun Xiao, and Long Chen. Video scene graph generation from single-frame weak supervision. In *Eleventh International Conference on Learning Representations (ICLR)*, 2022. 2
- [5] Anoop Cherian, Chiori Hori, Tim K Marks, and Jonathan Le Roux. (2.5+ 1) d spatio-temporal scene graphs for video question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 444–453, 2022. 1
- [6] Yuren Cong, Wentong Liao, Hanno Ackermann, Bodo Rosenhahn, and Michael Ying Yang. Spatial-temporal transformer for dynamic scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16372–16382, 2021. 2, 6, 7, 8
- [7] Yuren Cong, Michael Ying Yang, and Bodo Rosenhahn. Reltr: Relation transformer for scene graph generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2023. 2, 7, 1
- [8] Naina Dhingra, Florian Ritter, and Andreas Kunz. Bgt-net: Bidirectional gru transformer network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2150–2159, 2021. 2
- [9] Azade Farshad, Yousef Yeganeh, Yu Chi, Chengzhi Shen, Björn Ommer, and Nassir Navab. Scenegenie: Scene graph guided diffusion models for image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 88–98, 2023. 1
- [10] Shengyu Feng, Hesham Mostafa, Marcel Nassar, Somdeb Majumdar, and Subarna Tripathi. Exploiting long-term dependencies for generating dynamic scene graphs. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5130–5139, 2023. 6, 7
- [11] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *Eleventh International Conference on Learning Representations (ICLR)*, 2021. 5
- [12] Xia Hua, Xinqing Wang, Ting Rui, Faming Shao, and Dong Wang. Adversarial reinforcement learning with object-scene relational graph for video captioning. *IEEE Transactions on Image Processing (TIP)*, 31:2004–2016, 2022. 1
- [13] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10236–10247, 2020. 1, 4, 6, 7
- [14] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3668–3678, 2015. 2
- [15] Anant Khandelwal. Flocode: Unbiased dynamic scene graph generation with temporal consistency and correlation debiasing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2516–2526, 2024. 2
- [16] 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 (IJCV)*, 123:32–73, 2017. 2
- [17] Harold W Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics quarterly (NRL)*, 2(1-2): 83–97, 1955. 5
- [18] Stan Weixian Lei, Difei Gao, Jay Zhangjie Wu, Yuxuan Wang, Wei Liu, Mengmi Zhang, and Mike Zheng Shou. Symbolic replay: Scene graph as prompt for continual learning on vqa task. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 1250–1259, 2023. 1
- [19] Jiankai Li, Yunhong Wang, Xiefan Guo, Ruijie Yang, and Weixin Li. Leveraging predicate and triplet learning for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28369–28379, 2024. 2
- [20] Rongjie Li, Songyang Zhang, and Xuming He. Sgtr: End-to-end scene graph generation with transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19486–19496, 2022. 2
- [21] Yiming Li, Xiaoshan Yang, and Changsheng Xu. Dynamic scene graph generation via anticipatory pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13874–13883, 2022. 6, 7
- [22] T Lin. Focal loss for dense object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2980–2998, 2017. 4

- [23] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3746–3753, 2020. 2, 6, 7
- [24] Xin Lin, Chong Shi, Yibing Zhan, Zuopeng Yang, Yaqi Wu, and Dacheng Tao. Td²-net: Toward denoising and debiasing for video scene graph generation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 3495–3503, 2024. 2, 7, 1
- [25] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. Dab-detr: Dynamic anchor boxes are better queries for detr. In *Eleventh International Conference on Learning Representations (ICLR)*, 2022. 2
- [26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Eleventh International Conference on Learning Representations (ICLR)*, 2017. 6
- [27] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *European Conference on Computer Vision (ECCV)*, pages 852–869, 2016. 2, 6, 7
- [28] Yukuan Min, Aming Wu, and Cheng Deng. Environment-invariant curriculum relation learning for fine-grained scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13296–13307, 2023. 2
- [29] Sayak Nag, Kyle Min, Subarna Tripathi, and Amit K Roy-Chowdhury. Unbiased scene graph generation in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22803–22813, 2023. 2, 6, 7, 8, 1
- [30] Shan Ning, Longtian Qiu, Yongfei Liu, and Xuming He. Hoiclip: Efficient knowledge transfer for hoi detection with vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23507–23517, 2023. 2, 3
- [31] Tao Pu, Tianshui Chen, Hefeng Wu, Yongyi Lu, and Liang Lin. Spatial-temporal knowledge-embedded transformer for video scene graph generation. *IEEE Transactions on Image Processing (TIP)*, 33:556–568, 2023. 2, 1
- [32] 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 *International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021. 2, 3
- [33] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015. 2
- [34] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision (ECCV)*, pages 510–526, 2016. 6
- [35] Masato Tamura, Hiroki Ohashi, and Tomoaki Yoshinaga. Qpic: Query-based pairwise human-object interaction detection with image-wide contextual information. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10410–10419, 2021. 2
- [36] Yao Teng, Limin Wang, Zhifeng Li, and Gangshan Wu. Target adaptive context aggregation for video scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13688–13697, 2021. 2, 6, 7
- [37] Guan Wang, Zhimin Li, Qingchao Chen, and Yang Liu. Oed: Towards one-stage end-to-end dynamic scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27938–27947, 2024. 3, 6, 7, 8, 1
- [38] Hao Wang, Guosheng Lin, Steven CH Hoi, and Chunyan Miao. Cross-modal graph with meta concepts for video captioning. *IEEE Transactions on Image Processing (TIP)*, 31: 5150–5162, 2022. 1
- [39] Jingyi Wang, Jinfa Huang, Can Zhang, and Zhidong Deng. Cross-modality time-variant relation learning for generating dynamic scene graphs. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8231–8238. IEEE, 2023. 6, 7
- [40] Shuang Wang, Lianli Gao, Xinyu Lyu, Yuyu Guo, Pengpeng Zeng, and Jingkuan Song. Dynamic scene graph generation via temporal prior inference. In *Proceedings of the 30th ACM International Conference on Multimedia (ACMMM)*, pages 5793–5801, 2022. 6, 7
- [41] Xiaoshi Wu, Feng Zhu, Rui Zhao, and Hongsheng Li. Cora: Adapting clip for open-vocabulary detection with region prompting and anchor pre-matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7031–7040, 2023. 2
- [42] Xu Yang, Chongyang Gao, Hanwang Zhang, and Jianfei Cai. Hierarchical scene graph encoder-decoder for image paragraph captioning. In *Proceedings of the 30th ACM International Conference on Multimedia (ACMMM)*, pages 4181–4189, 2020. 1
- [43] Zekun Yang, Noa Garcia, Chenhui Chu, Mayu Otani, Yuta Nakashima, and Haruo Takemura. Bert representations for video question answering. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1556–1565, 2020. 1
- [44] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5831–5840, 2018. 2
- [45] Chaofan Zheng, Xinyu Lyu, Yuyu Guo, Pengpeng Zeng, Jingkuan Song, and Lianli Gao. Learning to generate scene graph from head to tail. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2022. 2