OED: Towards One-stage End-to-End Dynamic Scene Graph Generation

Guan Wang[1] Zhimin Li[2] Qingchao Chen[3] Yang Liu[1*]
[1]Wangxuan Institute of Computer Technology, Peking University
[2]Tencent Inc. [3]National Institute of Health Data Science, Peking University
w.g@stu.pku.edu.cn zhiminli.cn@outlook.com {qingchao.chen, yangliu}@pku.edu.cn

Abstract

Dynamic Scene Graph Generation (DSGG) focuses on identifying visual relationships within the spatial-temporal domain of videos. Conventional approaches often employ multi-stage pipelines, which typically consist of object detection, temporal association, and multi-relation classification. However, these methods exhibit inherent limitations due to the separation of multiple stages, and independent optimization of these sub-problems may yield sub-optimal solutions. To remedy these limitations, we propose a one-stage end-to-end framework, termed OED, which streamlines the DSGG pipeline. This framework reformulates the task as a set prediction problem and leverages pair-wise features to represent each subject-object pair within the scene graph. Moreover, another challenge of DSGG is capturing temporal dependencies, we introduce a Progressively Refined Module (PRM) for aggregating temporal context without the constraints of additional trackers or handcrafted trajectories, enabling end-to-end optimization of the network. Extensive experiments conducted on the Action Genome benchmark demonstrate the effectiveness of our design. The code and models are available at https://github.com/guanw-pku/OED.

1. Introduction

Scene Graph Generation (SGG) has emerged as a crucial component in advancing human-centric scene understanding, garnering significant research attention in recent years. The SGG research has been extensively employed in various high-level tasks, such as Visual Question Answering [3, 12, 18, 20], Visual Commonsense Reasoning [31] and Image Generation [7]. Dynamic scene graph generation (DSGG) further extends SGG with additional temporal dimension and then becomes more challenging, which aims at understanding more informative spatial-temporal cues.

The primary objective of DSGG is to provide a sequence of comprehensive and structural representation of scenes by taking video sequence as input and detecting subject and object as nodes, as well as identifying the multi-relations between them as edges in graphs. Existing studies [8, 14] present promising results in DSGG by decoupling this task into multiple stages: instance detection, temporal association, and multi-relation classification, as illustrated in Fig. 1(a). Specifically, subject and object detection results are obtained using an object detector. Subsequently, a temporal module, such as a tracker, establishes temporal links between instances in adjacent frames and aggregates temporal features on pair-wise combined subject-object proposals within the temporal sequence. The final stage entails performing multi-relation classification utilizing pair-wise features. Notably, prior to multi-relation classification, multi-stage methods requires the enumeration of instance or tracklet pairs. However, enumerative constructing all candidate

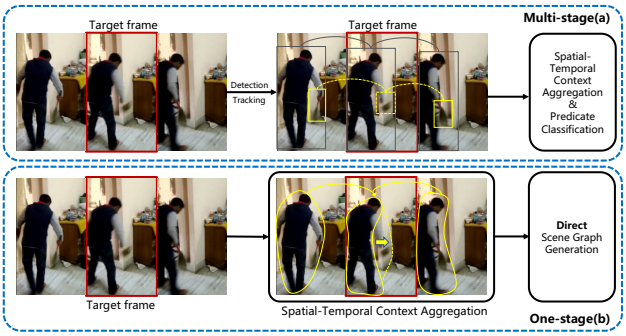

Figure 1. Comparison between existing multi-stage paradigm and proposed one-stage end-to-end framework. (a) Multi-stage methods, typically detect object instances by individual object detector and may associate objects between frames to aggregate temporal context based on detection results, followed by predicate classification for all candidate subject and object pairs, where tracking maybe lost. (b) Our one-stage end-to-end method, directly generates dynamic scene graph for given video sequence, without individual consideration for object instance detection and tracking. The missing spatial context and predicate temporal dependency could be supplemented with spatial context of reference frames.

*Corresponding author

subject-object pairs inevitably introduces not only a considerable of negative samples, significantly outnumbering positive ones, which is harmful in the training, but also substantial redundant computational costs. Furthermore, these methods suffer the problem of sub-optimal solutions due to independent optimization of separated multiple stages.

Recent research [38] has proposed an end-to-end framework that unifies multiple tasks through a transformer-based structure. This approach first obtains instance results for each frame based on a transformer-based tracking model [35]. Subsequently, it enumerates subject-object pairs from tracked objects. Finally, the selected pairs from the previous frame are propagated to the target frame to aggregate temporal context and perform relation classification with pair-wise features. Nonetheless, this method still adopts a two-stage paradigm, constructing candidate subject-object pairs based on the detection results and subsequently executing relation classification. Such pairs are not always valid, rendering the pipeline more intricate and computationally expensive.

Within the DSGG research community, the primary challenge lies in capturing temporal dependencies. Addressing this concern facilitates the detection of occluded or blurred objects and the perception of relation reliant on adjacent frames, such as *looking at*, *holding* and *drinking from*. Existing methods have adopted complex yet sub-optimal strategies, including the utilization of trajectories or 3D convolution operators, to equip models with the capability to capture temporal dependencies. Nevertheless, trajectories generated by additional trackers are difficult for joint training, while 3D convolutions introduce substantial computational overhead, thereby limiting the overall efficiency and effectiveness of these approaches.

In this paper, we present a novel **O**ne-stage **E**nd-to-end architecture to directly predict **D**ynamic scene graphs through set prediction, termed **OED**, and introduce an effective temporal context aggregating strategy. OED reformulates dynamic scene graph generation as a set prediction problem by extending DETR [2] across both spatial and temporal dimensions. Our method comprises a Spatial Context Aggregation module and a Temporal Context Aggregation module, as shown in Fig. 1(b). The architecture first employs cascaded decoders to aggregate spatial context, with the former outputting pair-wise instance feature and the latter generating pair-wise relation feature. The pair-wise instance feature aggregates pair-wise instance related information and acts as the pair-wise relation query in Pair-wise Relation Decoder, providing a strong prior. Subsequently, we concatenate both two features to obtain overall pair-wise feature and feed it into the proposed Progressively Refined Module (PRM) for temporal context aggregating. PRM progressively selects pair-wise feature in reference frames and simultaneously optimizes the pair-wise feature in target frame to mine temporal dependencies via selected reference features, which implicitly links temporal information. This approach eliminates additional trackers and handcrafted trajectories, enabling end-to-end optimization of the network. Following this, classification heads and regression heads are utilized to predict DSGG results given spatial-temporal aggregated pair-wise feature. Finally, due to the challenge of incomplete annotations in video training data, we compute predicate classification loss only over a portion of the predictions that match ground truth, mitigating potential deterioration caused by missing annotations.

In summary, the primary contributions are as follows: (1) We introduce a simple one-stage end-to-end framework for DSGG, termed **OED**, which models dynamic scene graphs as a set prediction problem with pair-wise feature. (2) To effectively mine the temporal dependencies of relation, we propose a Progressively Refined Module (PRM) for aggregating temporal context without the constraints of additional trackers, enabling end-to-end optimization of the network. (3) Experimental results on the Action Genome dataset demonstrate the superiority of the proposed one-stage end-to-end framework and the efficacy of the implemented temporal aggregation module.

2. Related Work

2.1. Scene Graph Generation

Scene Graph Generation for static image has caught broad attention since the benchmark was proposed [11]. Most previous works [16, 32, 33] adopt two-stage paradigms. The first stage is to detect all objects by a off-the-shelf object detector, such as a pretrained Faster R-CNN [22]. Then the relationship between all candidate object pairs is classified using designed modules, such as Message Passing [32, 33], Graph Convolutional Network [16], Casual Inference [26]. Recently, some works [5, 27] adopts one-stage end-to-end paradigms to improve detection and predicate classification jointly without explicit detection.

However, additional temporal axis brings the need for effective perception and usage of complex spatial-temporal context, which means that SGG model hardly effectively handles DSGG task and thus the extension is not trivial.

2.2. Dynamic Scene Graph Generation

2.2.1 Stacking Multi-stage Pipeline

Dynamic scene graph generation task and its benchmark are proposed by [9]. Given that the context dependencies of this task span both spatial and temporal dimensions, the simultaneous modeling of spatial-temporal information and constructing spatial-temporal interaction relationships are essential for enhancing DSGG performance.

Previous approaches employed a multi-stage paradigm, utilizing object detectors to identify instances, and subse-

quently performing grouping and multi-relation classification on the detection results. STTran [4] leverages an off-the-shelf object detector to obtain instance detection results and then enhance and classify pairwise features via a spatial-temporal transformer. HOTR [10] introduces additional human pose features in the second stage to capture more relationship information. Some works [1, 28] capture visual temporal dynamics from 3D CNN backbone, where TRACE [28] designs a hierarchical tree to aggregate spatial context and [1] introduces message passing in a spatial-temporal graph to improve the spatial-temporal feature. TPT [38] unifies object detection and object tracking together, thereby enhancing the pair-wise feature by the results of previous frame and aggregating rich spatial-temporal context. These methods enhance the instance features derived from object detection or achieve instance features via tracker directly, aggregate temporal information, and improve the accuracy of the classification.

Nonetheless, they rely on dedicated modules for multiple tasks, thereby requiring individualized training schemes. This inevitably disrupts the collaborative interactions among these modules for different sub-tasks, ultimately resulting in sub-optimal solutions. On the other hand, the numerous interaction pairs generated by enumeration operation lead to substantial computational redundancy.

2.2.2 Modeling Temporal Dependence

In recent years, an increasing number of studies [8, 14, 29] have delved into the temporal information of features, constructing trajectory information to enhance inter-frame temporal dependencies. APT [14] proposes an anticipatory pre-training scheme to explore the temporal correlations among object pairs across different frames based on object tracking. TR^2[29] tracks object first and utilize the cross-modality feature from CLIP [21] to guide the consistency between the temporal difference of pair-wise visual and semantic features. DSG-DETR [8] constructs inter-frame trajectories of object instances and pairs using bipartite graph matching, aiming to enhance the long-term temporal dependencies of temporal information and subsequently improve the performance of multi-relation classification.

Nevertheless, trajectories generated by additional trackers are difficult for joint training, while handcrafted trajectories exhibit poor robustness, are prone to introducing noise, and consequently impact network performance. In this work, we formulate the DSGG as a one-stage set prediction task, utilizing pair-wise features to represent each subject-object pair within the scene graph. The one-stage framework eliminates the issue of inconsistent optimization objectives introduced by multi-stage approaches. Concurrently, we propose the PRM, which progressively filters reference frame pair-wise features and simultaneously optimizes target frame pair-wise features to mine temporal dependencies of relationships. By discarding additional trackers and handcrafted operators, notable performance enhancement is attained.

3. Method

3.1. Problem Formulation

Dynamic scene graph generation aims to detect visual relations occurred in the target frame in video sequence. Detected visual relations are represented by a special form of graph, called scene graph. The nodes and edges in the scene graph refer to object instances and relations between them respectively, where an object instance consists of its label and spatial position. Therefore, a scene graph is equivalent to a list of triplets $\langle subject, predicate, object\rangle$, or $\langle s, p, o\rangle$ for short. To generate scene graphs, the target is to model the joint probability $P(\langle s, p, o\rangle|V)$ at each frame, where $\langle s, p, o\rangle$ belongs to a pre-defined triplet set and V denotes the video sequence.

Some of previous works [4] factorize the joint probability as follows:

$$P(\langle s, p, o\rangle|V) = P(p|s, o)P(s, o|D)P(D|V) \quad (1)$$

where D represents object detection results in V. Recent works [8, 29, 38] introduce additional tracking across frames to aggregate temporal information:

$$P(\langle s, p, o\rangle|V) = P(p|s, o)P(s, o|\mathcal{T})P(\mathcal{T}|D)P(D|V) \quad (2)$$

where $\mathcal{T}$ represents object tracking results in V. These two types of solutions inevitably lead to multi-stage pipeline, which is sub-optimal due to separate training and upper bound of each stage.

In this work, we propose a one-stage method to directly model $P(\langle s, p, o\rangle|V)$. To utilize the temporal dependencies of predicate and alleviate the impact of motion and occlusion, we progressively aggregate temporal context information from reference frames. That is to say, we directly model $P(\langle s, p, o\rangle|I_i, \{I_{ref}\})$ at i-th frame, where I_i indicates the i-th frame and $\{I_{ref}\}$ refers to the reference frames of I_i.

3.2. Overview

The pipeline of proposed approach is illustrated as Fig. 2. Given the target frame and reference frames, OED directly generates scene graphs with spatial-temporal context in a way of set prediction.

First, the CNN backbone and Transformer encoder are sequentially utilized to extract visual features of each frame. To extract and aggregate useful spatial context, we adopt DETR-like [2] architecture and associate learnable queries

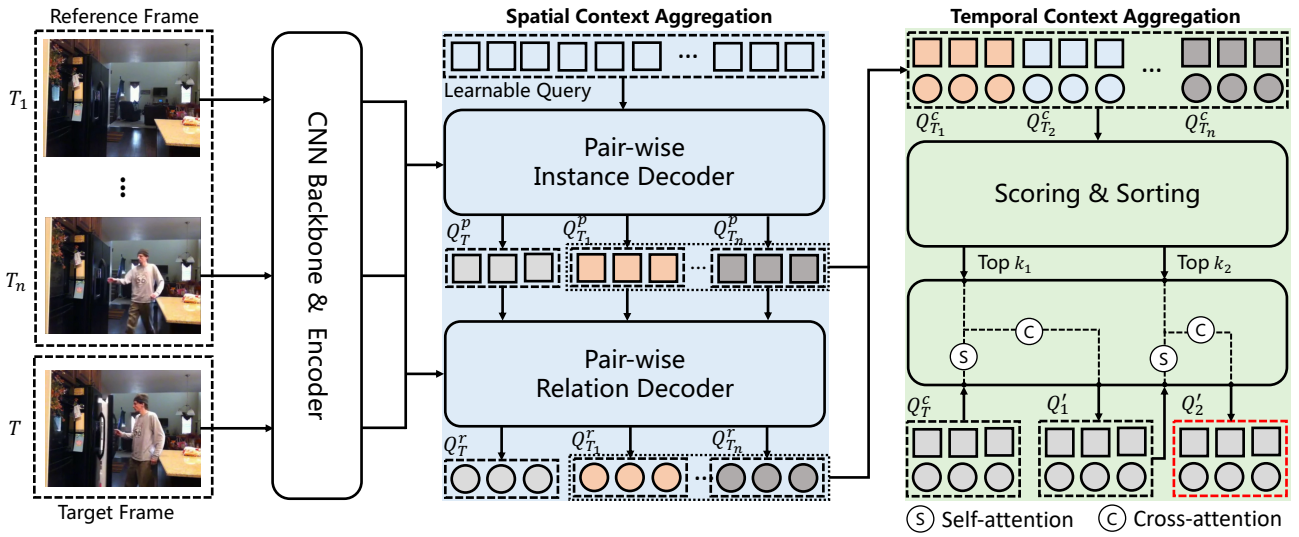

Figure 2. **OED Framework**: Spatial-temporal context aggregation is conducted within a one-stage end-to-end paradigm. Visual features of the target frame and reference frames are extracted using a CNN backbone and a Transformer encoder. Subsequently, two cascaded decoders are employed to aggregate spatial context both within and between pairs. Temporal context is then aggregated in a progressively refined manner, considering pair-wise features of the target frame and reference frames.

with pair-wise feature of candidate object pairs. The pair-wise feature then extracts and aggregates spatial context in Transformer decoder. To improve the detection of blurred object and predicate classification with dependencies on contextual frames at the same time, we introduce a progressive refined pair-wise feature interaction module (PRM) to select and aggregate useful information from reference frames to the pair-wise feature of the target frame in a progressively refined way. PRM fuses additional temporal context with the spatial aggregated pair-wise feature of the target frame, and then we obtain the final pair-wise feature with spatial-temporal context.

The pair-wise detection and predicate classification results will form a list of triplets $\langle s, p, o \rangle$, which corresponds to the scene graph of target frame.

3.3. Spatial Context Aggregation

CNN visual backbone and Transformer encoder yield the visual features of each frame $\boldsymbol{F} = \{f_T, f_{T_1}, ..., f_{T_n}\}$, where $f_i \in \mathbb{R}^{HW \times d_{\text{model}}}, i \in \{T, T_1, ..., T_n\}$ and (H, W) represents the scale of feature map.

In order to fully exploit the potential of set prediction design in the DSGG, we associate learnable queries $Q \in \mathbb{R}^{N_q \times d_{\text{model}}}$ with subject-object pairs (s, o). The pair-wise queries are to obtain specific visual features related to corresponding candidate pairs, which means spatial context aggregation. In addition to aggregating the spatial information of each pair individually, the underlying connections between different pairs are significant as well, e.g. (*person*, *dish*) tends to co-occur with (*person*, *table*). We model and aggregate spatial context in these two ways using multi-head attention in transformer decoders.

Multi-head Attention. Given query embedding X_q, key embedding X_k and value embedding X_v, the output of multi-head attention is computed as follows:

$$\begin{aligned} \text{MHead}(X_q, X_k, X_v) &= \text{Concat}(\text{head}_1, ..., \text{head}_h)W \\ \text{head}_i &= \text{Attention}(X_q W_i^q, X_k W_i^k, X_v W_i^v) \\ \text{Attention}(X_q, X_k, X_v) &= \text{softmax}\big(\frac{X_q X_k^T}{\sqrt{d_k}}\big) X_v \end{aligned} \tag{3}$$

where $W_i^q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^k \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^v \in \mathbb{R}^{d_{\text{model}} \times d_v}$ and $W \in \mathbb{R}^{h d_v \times d_{\text{model}}}$.

The spatial context dependencies between pairs are captured by multi-head self-attention $\text{SelfAttn}(Q)$ and Spatial Context Aggregation of each pair is performed by multi-head cross-attention $\text{CrossAttn}(Q, f_i)$ for i-th frame of given video sequence.

$$\begin{aligned} \text{SelfAttn}(Q) &= \text{MHead}(Q, Q, Q) \\ \text{CrossAttn}(Q, f_i) &= \text{MHead}(Q, f_i, f_i) \end{aligned} \tag{4}$$

Considering that a single decoder struggles to handle two different tasks [36], pair detection and predicate classification, we introduce two cascaded decoders. One is tailored to decode features for pair-wise instance related feature, while another one is for pair-wise predicate related feature. Specifically, a set of learnable queries are used to capture pair-wise instance related information in Pair-wise Instance Decoder. Considering that pair-wise instance related feature can act a strong prior to predicate classification, we

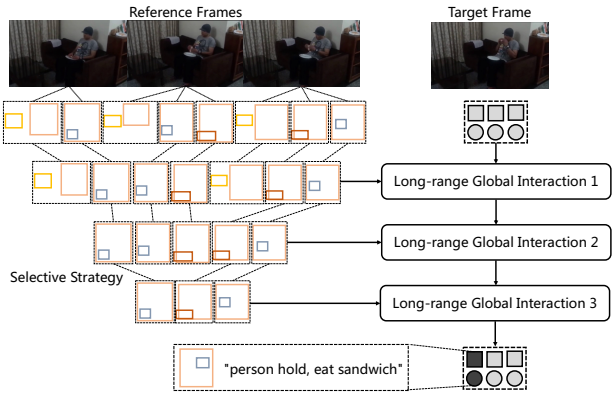

Figure 3. Progressively refined long-range global temporal context aggregation.

take it as pair-wise relation query to the Pair-wise Relation Decoder. In a word, the cascaded decoders aggregate the spatial context by pair-wise instance query and pair-wise relation query.

Pair-wise Instance Decoder. A set of learnable queries Q extract and aggregate pair-wise instance related spatial context, as shown in Eq. 4. Pair-wise instance feature from Pair-wise Instance Decoder $Q^p = \{\boldsymbol{q_1^p}, ..., \boldsymbol{q_{N_q}^p}\} \in \mathbb{R}^{N_q \times d_{\text{model}}}$ then acts as query of Pair-wise Relation Decoder.

Pair-wise Relation Decoder. Apparently pair-wise instance could provide strong priors to classify predicate, especially for spatial and contacting predicates, such as (*person*, *chair*) with large overlapping area leading to *sitting*. Thus, Pair-wise Relation Decoder take pair-wise instance feature Q^p as query to capture and aggregate pair-wise relation specific spatial context $Q^r = \{\boldsymbol{q_1^r}, ..., \boldsymbol{q_{N_q}^r}\} \in \mathbb{R}^{N_q \times d_{\text{model}}}$, similar to the operations in the Pair-wise Instance Decoder.

Therefore, the spatial context information of triplets $\langle s, p, o \rangle$ corresponds to overall pair-wise feature $Q^c = \text{Concat}(Q^p, Q^r) = \{\boldsymbol{q_1^c}, ..., \boldsymbol{q_{N_q}^c}\} \in \mathbb{R}^{N_q \times 2d_{\text{model}}}$, where $\boldsymbol{q_i^c} = \text{Concat}(\boldsymbol{q_i^p}, \boldsymbol{q_i^r}), i \in \{1, ..., N_q\}$.

3.4. Temporal Context Aggregation

Through the cascaded decoders, the pair-wise feature Q^c has aggregated rich spatial context information. Besides spatial dependencies discussed in section 3.3, there are temporal dependencies of predicate across frames, e.g. (*looking at - holding - drinking from*). Reference frames could also improve the detection of blurred and occluded object in the target frame. Therefore, this section further supplements pair-wise feature with additional temporal context, which is orthogonal to the spatial context. To achieve this, we propose a multi-step progressively refined interaction module PRM, motivated by [6].

Specifically, we extract the spatial pair-wise feature of target frame and reference frames $\{Q_T^c, Q_{T_1}^c, ..., Q_{T_n}^c\}$ and concatenate the pair-wise features of reference frames together $Q_{ref}^c = \{\boldsymbol{q_1^c}, ..., \boldsymbol{q_{n \times N_q}^c}\}$. Then we split the pair-wise feature $\boldsymbol{q_i^c}$ into pair-wise instance feature $\boldsymbol{q_i^p}$ and pair-wise relation feature $\boldsymbol{q_i^r}$ and use classification heads to score subject and object with $\boldsymbol{q_i^p}$ and score predicate with $\boldsymbol{q_i^r}$. We calculate the score of triplet by the multiplication of subject score s_{sub}, object score s_{obj} and predicate score s_{rel}, and then rank them.

$$p(\boldsymbol{q_i^c}) = s_{sub} \times s_{obj} \times s_{rel} \tag{5}$$

The pair-wise feature with higher score tends to have more correlations with corresponding ground truth. Thus, we aggregate the temporal context from more confident reference pair-wise features. Selecting a fixed number of reference pair-wise features is hard to hit a good balance and either bringing much noise or missing some informative reference pair-wise features, so we aggregate temporal context in a multi-step progressively refined way.

In i-th step, the selected Top-K reference pair-wise features $Q_{ref}^{c;k_i}$ interact with the pair-wise features in the target frame Q_i' in Transformer decoder progressively, which is formulated as

$$\begin{aligned} Q_i' &= \text{CrossAttn}(\text{SelfAttn}(Q_{i-1}'), Q_{ref}^{c;k_i}), i > 0 \\ Q_{ref}^{c;k_i} &= \text{Top-K}(Q_{ref}^c, k_i) \\ Q_i' &= Q_T^c \end{aligned} \tag{6}$$

The value of k_i is gradually reduced to obtain more confident refined reference pair-wise features. This progressively refined selection realizes the trade-off between more context information and less background noise.

As shown in Fig. 3, in the progressively refined process, some selected noises from reference frames, such as *towel* denoted as yellow box, are gradually filtered out. Besides, PRM provides a way of long-range global temporal interaction, which means that the temporal interaction is not constrained inside the trajectories of object pair. With the benefit of global perspective, PRM could capture the gradual movement of *sandwich* from the *dish* to *person*.

After m steps' progressively refined temporal aggregation, the pair-wise feature Q_m' is composed of abundant spatial-temporal context information. The spatial-temporal pair-wise feature is then divided into pair-wise instance feature Q_T^p and pair-wise relation feature Q_T^r , which are used to detect object pair and classify predicate respectively.

3.5. Training and Inference

3.5.1 Training

Given video sequence, OED generates a fxied set of predictions for each frame. Then Hungarian algorithm finds out the optimal one-to-one matching $\hat{\sigma} \in \mathfrak{S}$ between prediction

set $P = \{p_i\}_{i=1}^{N_q}$ and ground truth set $G = \{g_i\}_{i=1}^{N_q}$ that is padded with $\varnothing$ to the same length.

$$\hat{\sigma} = \arg\min_{\sigma \in \mathfrak{S}_N} \sum_{i}^{N_q} \mathcal{L}_{\text{match}} \left(g^i, p^{\sigma(i)} \right) \tag{7}$$

where $\mathcal{L}_{\text{match}}(g_i, p_{\sigma(i)})$ is the pair-wise cost between ground truth g_i and the prediction with index $\sigma(i)$. In the dynamic scene graph generation, we define the matching loss as:

$$\mathcal{L}_{\text{match}} \left(g^i, p^{\sigma(i)} \right) = \sum_{j \in \{\text{s,o,p}\}} \alpha_j \mathcal{L}^j_{\text{cls}} + \beta \sum_{j \in \{\text{s,o}\}} \mathcal{L}^j_{\text{box}} \tag{8}$$

where $\mathcal{L}^j_{\text{cls}} = \mathcal{L}^j_{\text{cls}} \left(g^i_j, p^{\sigma(i)}_j \right), j \in \{s, o, p\}$ indicates the classification loss for subject, object and predicate, $\mathcal{L}^j_{\text{box}} = \mathcal{L}^j_{\text{box}} \left(g^i_j, p^{\sigma(i)}_j \right), j \in \{s, o, p\}$ indicates the bounding box regression loss of subject and object. We use cross entropy loss as classification loss of subject $\mathcal{L}^s_{\text{cls}}$ and object $\mathcal{L}^o_{\text{cls}}$, weighted sum of L_1 loss and GIoU loss [23] as bounding box regression loss of subject $\mathcal{L}^s_{\text{box}}$ and object $\mathcal{L}^o_{\text{box}}$ and focal loss [15] as the classification loss of predicate.

We adopt the matching loss as overall objective function but predicate loss. Due to the incomplete annotation issue, we only calculate the predicate loss over the predictions matched with real ground truth, which is not padded background.

$$\mathcal{L}^{p'}_{\text{cls}} \left(g^i, p^{\sigma(i)} \right) = \mathbb{1}_{\{g^i \neq \varnothing\}} \mathcal{L}^p_{\text{cls}} \left(g^i, p^{\sigma(i)} \right) \tag{9}$$

The intuition here is that if the unmatched predictions are directly deemed as negative samples, the false negative samples keep misleading the model, which are positive samples at other frames instead. The experiment in ablation studies shows that the matched predicate classification loss effectively mitigates the impact of incomplete annotation.

3.5.2 Inference

In the inference stage, there are a fixed number of pair predictions for each frame. Because there may be multiple predicates for one pair, we rank the candidate triplets by scoring them as the multiplication of three-part confidences. To reduce duplicate triplet detection, we filter out the predictions with lower scores that have the same triplet label and large overlapping area with others. Specifically, we take the multiplication of IoU of subject and object as the correlation in NMS to filter repeated predictions. The scene graph is generated by those retained triplet predictions.

4. Experiments

4.1. Experimental Setting

Dataset: We evaluate OED on the Action Genome (AG) dataset [9], which annotates 234, 253 frame scene graphs for sampled frames from around 10K videos, based on Charades dataset [24]. The annotations cover 35 object categories and 25 predicates. The overall predicates consist of three types of predicates: attention, spatial and contacting. There may be multiple spatial predicates or contacting predicates between the same pair.

Evaluation Metrics: Following previous works [8, 9, 14], we adopt Recall@k as evaluation metrics to measure the fraction of ground truth hit in the top k predictions under *With Constraint* and *No Constraints* setting, where $k \in \{10, 20, 50\}$. Specifically, We evaluate our method on two protocols: scene graph detection (SGDET) and predicate classification (PredCLS), following TPT [38]. SGDET aims to generate scene graphs for given videos, comprising detection results of subject-object pairs and the associated predicates. The localization of object prediction is considered accurate when the Intersection over Union (IoU) between the prediction and ground truth is greater than 0.5. PredCLS, a simplified task to eliminate object detection errors, requires methods to classify predicates for given oracle detection results of subject-object pairs.

Implementation Details: We employ ResNet-50 as the CNN backbone. The Image Encode, Pair-wise Decoder and Relation Decoder consist of 6 transformer layers, with the number of predefined learnable query $N_q = 100$. Following TPT [38], we initialize Image Encoder and Pair-wise Decoder with the weights pretrained on the MS-COCO dataset and subsequently fine-tune all modules on the Action Genome dataset. PRM includes three instances of progressively refind pair-wise interaction with Top-K as [$80n$, $50n$, $30n$] respectively, where n denotes the number of reference frames. The threshold adopted in inference stage is 0.9. For the PredCLS task, aimed at predicting predicate labels for specified object pairs, we initialize the learnable queries using semantic features derived from the Glove embeddings of the given pair labels. Additionally, we incorporate position embeddings with spatial features obtained from the specified bounding boxes of the pairs. During inference, we derive the outputs by associating the labels and bounding boxes of the ground truth with the predicate classification results for the corresponding pairs.

4.2. Comparison with State of the Arts

We present a comparison of our results with state-of-the-art methods for dynamic scene graph generation in Tab. 1. The performance in the SEDET task exemplifies the effectiveness of our approach. The SGDET task is aligned with the objective of dynamic scene graph generation, which entails generating scene graphs by aggregating spatial-temporal context from video sequences without incorporating additional information. In SGDET, our streamlined one-stage end-to-end pipeline surpasses other methods across all metrics under both *With Constraint* and *No Constraints* settings.

Table 1. Comparison with state-of-the-art dynamic scene graph generation methods on Action Genome. The others methods follow multi-stage paradigms, while ours adopt a one-stage one.

Method	With Constraint						No Constraints					
	SGDET			PredCLS			SGDET			PredCLS		
	R@10	R@20	R@50	R@10	R@20	R@50	R@10	R@20	R@50	R@10	R@20	R@50
VRD [17]	19.2	24.5	26.0	51.7	54.7	54.7	19.1	28.8	40.5	59.6	78.5	99.2
MSDN [13]	24.1	32.4	34.5	65.5	68.5	68.5	23.1	34.7	46.5	74.9	92.7	99.0
M-FREQ [34]	23.7	31.4	33.3	62.4	65.1	65.1	22.8	34.3	46.4	73.4	92.4	99.6
VCTree [25]	24.4	32.6	34.7	66.0	69.3	69.3	23.9	35.3	46.8	75.5	92.9	99.3
RelDN [37]	24.5	32.8	34.9	66.3	69.5	69.5	24.1	35.4	46.8	75.7	93.0	99.0
GPS-Net [16]	24.7	33.1	35.1	66.8	69.9	69.9	24.4	35.7	47.3	76.0	93.6	99.5
TRACE [28]	13.9	14.5	14.5	27.5	27.5	27.5	26.5	35.6	45.3	72.6	91.6	96.4
RelTR [5]	19.7	23.4	25.9	-	-	-	20.9	24.6	28.2	-	-	-
STTran [4]	25.2	34.1	37.0	68.6	71.8	71.8	24.6	36.2	48.8	77.9	94.2	99.1
APT [14]	26.3	36.1	38.3	69.4	73.8	73.8	25.7	37.9	50.1	78.5	95.1	99.2
STTran-TPI [30]	26.2	34.6	37.4	69.7	72.6	72.6	-	-	-	-	-	-
TR^2 [29]	26.8	35.5	38.3	70.9	73.8	73.8	27.8	39.2	50.0	83.1	96.6	**99.9**
TEMPURA [19]	28.1	33.4	34.9	68.8	71.5	71.5	29.8	38.1	46.4	80.4	94.2	99.4
DSG-DETR [8]	30.3	34.8	36.1	-	-	-	32.1	40.9	48.3	-	-	-
TPT [38]	-	-	-	-	-	-	32.0	39.6	51.5	**85.6**	**97.4**	**99.9**
Ours	**33.5**	**40.9**	**48.9**	**73.0**	**76.1**	**76.1**	**35.3**	**44.0**	**51.8**	83.3	95.3	99.2

#	Baseline	SA	TA	With Constraint			No Constraints		
				R@10	R@20	R@50	R@10	R@20	R@50
1	✓			26.3	29.2	32.1	28.4	32.9	37.2
2	✓	✓		31.5	37.7	43.7	33.4	41.6	49.0
3	✓	✓	✓	33.5	40.9	48.9	35.3	44.0	51.8

Table 2. Ablation studies on our framework.

#	Baseline	CD	ML	With Constraint			No Constraints		
				R@10	R@20	R@50	R@10	R@20	R@50
1	✓			26.3	29.2	32.1	28.4	32.9	37.2
2	✓	✓		27.2	30.3	33.7	29.2	34.0	38.4
3	✓	✓	✓	31.5	37.7	43.7	33.4	41.6	49.0

Table 3. Ablation studies on Spatial Context Aggregation.

#	SA	NPR	PR	With Constraint			No Constraints		
				R@10	R@20	R@50	R@10	R@20	R@50
1	✓			31.5	37.7	43.7	33.4	41.6	49.0
2	✓	✓		32.3	39.7	47.8	34.0	42.7	50.6
3	✓	✓	✓	33.5	40.9	48.9	35.3	44.0	51.8

Table 4. Ablation studies on Temporal Context Aggregation.

OED outperforms the second-best methods by an average of 6.2% (3.2%, 4.8% and 10.6% respectively) under the *With Constraint* setting and an average of 2.2% (3.3%, 3.1% and 0.3% respectively) under *No Constraints* setting. This outcome underscores the importance of addressing dynamic scene graph generation as a comprehensive task rather than partitioning it into multiple sub-tasks. More performance comparison in SGDET task with long-tail issue related metircs can be found in supp.M section 1.

#	With Constraint			No Constraints		
	R@10	R@20	R@50	R@10	R@20	R@50
1	37.5	44.8	51.9	40.1	48.6	56.5
2	33.5	40.9	48.9	35.3	44.0	51.8

Table 5. Comparison between oracle query selection and progressively refined query selection.

In PredCLS, OED improves the performance of the second-best methods by an average of 2.1% (2.0%, 2.2% and 2.3% respectively) under the *With Constraint* setting. However, our method's performance in PredCLS is marginally lower than that of TPT and TR^2 under the *No Constraints* setting. We conjecture the reason is as follows: in the PredCLS task, oracle object tracks from ground truth are provided. Both TPT and TR^2 are tracking-based methods, and they utilize the oracle trajectories in their respective track-based temporal aggregation modules. Due to the nature of the one-stage paradigm, our method cannot explicitly use this information, which consequently reduces the efficiency of leveraging oracle information. Furthermore, TPT employs additional multi-scale features and TR^2

incorporates the CLIP [21] model, which is pre-trained using 4M image-text pairs, providing them with an advantage over our approach. Despite the fact that multi-stage methods benefit from oracle tracks and can directly aggregate entirely accurate spatial-temporal context, our approach still outperforms others by a significant margin under the *With Constraint* setting and attains comparable performance under the *No Constraints* setting.

4.3. Ablation Study

In this part, we evaluate the effectiveness of different modules and designs of our OED with SGDET task on the Action Genome test set.

Spatial-Temporal Context Aggregation: In Tab. 2, We evaluate the effectiveness of the proposed Spatial Context Aggregation (SA) and Temporal Context Aggregation (TA) modules individually. We first adapt DETR [2] for dynamic scene graph generation, establishing it as our baseline (#1), where the object pair predictions and predicate classification are derived from the same decoded query representation. By incorporating the Spatial Context Aggregation module into the baseline (#2), we observe a significant improvement in performance. This indicates that the performance of spatial scene graph generation plays a crucial role in the effectiveness of dynamic scene graph generation. Furthermore, when the Temporal Context Aggregation module is integrated alongside the Spatial Context Aggregation module (#3), a further gain is achieved. This suggests that effectively exploiting temporal dependency information can further enhance the performance of DSGG.

Designs in Spatial Context Aggregation: In Tab. 3, we evaluate the efficacy of the proposed Cascaded Decoders (CD) and Matched Predicate Loss (ML). Building upon the baseline, we introduce an additional Pair-wise Relation Decoder and combine the two decoders in a cascaded manner (#2), addressing the optimization challenges of unified representation in multi-task settings and the dependence of predicate classification on pair detection results. The performance improvement achieved by the cascaded decoders validates our aforementioned considerations. More qualitative results can be found in supp.M section 2. Furthermore, to mitigate the misleading effects of incomplete annotations in the Action Genome dataset, we compute the predicate loss only over the predictions from queries that match the ground truth (#3). The resulting performance gain underscores the effectiveness of the matched predicate classification loss in addressing the issue of incomplete annotations.

Designs in Temporal Context Aggregation: In Tab. 4, we evaluate the importance of temporal context and our proposed PRM. We select the spatial aggregation model as our baseline (#1). Taking into account the potential loss of information due to motion and the temporal dependency of predicates, we hypothesize that context clues can be obtained from adjacent reference frames to enhance pair detection and predicate classification. To incorporate the temporal context, we interact the pair-wise feature of target frame with all reference pair-wise features (#2) without progressively refined (NPR). The experimental results substantiate the effectiveness of temporal context in dynamic scene graph generation. Moreover, considering that not all pair-wise features are valid, as they may attend to duplicate areas or background noise, we implement a progressively refined interaction (PR) between the pair-wise features of target frame and reference frames (#3). The demonstrated effectiveness of the progressive refinement of pair-wise interactions indicates that filtering out background noise is crucial for improving semantic context aggregation.

4.4. Discussion

To further assess the effectiveness of temporal pair-wise interaction and estimate the upper bound of our PRM, we assume that the selected reference queries are oracle queries, meaning that these queries are matched with ground truth via bipartite matching. As illustrated in Tab. 5, the oracle selection (#1) achieves a significantly higher performance compared to our PRM (#2). This result indicates that there is still room for exploration and improvement in our approach. In the future, we plan to delve deeper into the effective selection of true positive samples from pair-wise features of reference frames. It is worth noting that our objective is not to obtain precise trajectories. We regard trajectories as a form of long-term yet local information, subject to instance-level constraints. It impedes the perception of the relationships among different instances across frames. We consider that long-term global information extracted by PRM plays a crucial role, and our approach focuses on filtering more accurate pair-wise instances across frames to facilitate the relation classification for the target frame.

5. Conclusion

In this paper, we present a one-stage end-to-end framework, named OED, for dynamic scene graph generation. Our approach reformulates the task as a set prediction problem and employs pair-wise features to represent each subject-object pair within the scene graph. Furthermore, we introduce a Progressively Refined Module (PRM) for temporal context aggregating. The PRM progressively filters pair-wise features of reference frames while simultaneously optimizing the pair-wise features of the target frame to extract temporal dependencies through filtered features. Consequently, OED achieves significant improvement over the baseline, establishing sota performance across all metrics in SGDET task.

Acknowledgements. This work was supported by the grants from the National Natural Science Foundation of China 62372014.

References

[1] Anurag Arnab, Chen Sun, and Cordelia Schmid. Unified graph structured models for video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8117–8126, 2021. 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*, pages 213–229. Springer, 2020. 2, 3, 8

[3] 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*, pages 444–453, 2022. 1

[4] 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*, pages 16372–16382, 2021. 3, 7

[5] Yuren Cong, Michael Ying Yang, and Bodo Rosenhahn. Reltr: Relation transformer for scene graph generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2, 7

[6] Jiajun Deng, Yingwei Pan, Ting Yao, Wengang Zhou, Houqiang Li, and Tao Mei. Relation distillation networks for video object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7023–7032, 2019. 5

[7] Azade Farshad, Yousef Yeganeh, Yu Chi, Chengzhi Shen, Böjrn Ommer, and Nassir Navab. Scenegenie: Scene graph guided diffusion models for image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 88–98, 2023. 1

[8] 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*, pages 5130–5139, 2023. 1, 3, 6, 7

[9] 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*, pages 10236–10247, 2020. 2, 6

[10] Jingwei Ji, Rishi Desai, and Juan Carlos Niebles. Detecting human-object relationships in videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8106–8116, 2021. 3

[11] 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*, 123:32–73, 2017. 2

[12] 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*, pages 1250–1259, 2023. 1

[13] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE international conference on computer vision*, pages 1261–1270, 2017. 7

[14] 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*, pages 13874–13883, 2022. 1, 3, 6, 7

[15] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 6

[16] 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*, pages 3746–3753, 2020. 2, 7

[17] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 852–869. Springer, 2016. 7

[18] Jianguo Mao, Wenbin Jiang, Xiangdong Wang, Zhifan Feng, Yajuan Lyu, Hong Liu, and Yong Zhu. Dynamic multistep reasoning based on video scene graph for video question answering. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3894–3904, Seattle, United States, 2022. Association for Computational Linguistics. 1

[19] 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*, pages 22803–22813, 2023. 7

[20] Tianwen Qian, Jingjing Chen, Shaoxiang Chen, Bo Wu, and Yu-Gang Jiang. Scene graph refinement network for visual question answering. *IEEE Transactions on Multimedia*, 2022. 1

[21] 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*, pages 8748–8763. PMLR, 2021. 3, 8

[22] 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. 2

[23] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference*

on computer vision and pattern recognition, pages 658–666, 2019. 6

[24] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 510–526. Springer, 2016. 6

[25] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6619–6628, 2019. 7

[26] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3716–3725, 2020. 2

[27] Yao Teng and Limin Wang. Structured sparse r-cnn for direct scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19437–19446, 2022. 2

[28] 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*, pages 13688–13697, 2021. 3, 7

[29] Jingyi Wang, Jinfa Huang, Can Zhang, and Zhidong Deng. Cross-modality time-variant relation learning for generating dynamic scene graphs. *arXiv preprint arXiv:2305.08522*, 2023. 3, 7

[30] 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*, pages 5793–5801, 2022. 7

[31] Zhecan Wang, Haoxuan You, Liunian Harold Li, Alireza Zareian, Suji Park, Yiqing Liang, Kai-Wei Chang, and Shih-Fu Chang. Sgeitl: Scene graph enhanced image-text learning for visual commonsense reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5914–5922, 2022. 1

[32] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5410–5419, 2017. 2

[33] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 670–685, 2018. 2

[34] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5831–5840, 2018. 7

[35] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *European Conference on Computer Vision*, pages 659–675. Springer, 2022. 2

[36] Aixi Zhang, Yue Liao, Si Liu, Miao Lu, Yongliang Wang, Chen Gao, and Xiaobo Li. Mining the benefits of two-stage and one-stage hoi detection. *Advances in Neural Information Processing Systems*, 34:17209–17220, 2021. 4

[37] Ji Zhang, Kevin J Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. Graphical contrastive losses for scene graph parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11535–11543, 2019. 7

[38] Yong Zhang, Yingwei Pan, Ting Yao, Rui Huang, Tao Mei, and Chang-Wen Chen. End-to-end video scene graph generation with temporal propagation transformer. *IEEE Transactions on Multimedia*, 2023. 2, 3, 6, 7