Location-Free Scene Graph Generation

Ege Özsoy[1,2] Felix Holm[1] Chantal Pellegrini[1,2] Tobias Czempiel[1] Mahdi Saleh[1]
Nassir Navab[1] Benjamin Busam[1]

[1] Technical University of Munich [2] Munich Center for Machine Learning

ege.oezsoy@tum.de

Abstract

Scene Graph Generation (SGG) is a visual understanding task that describes a scene as a graph of entities and their relationships, traditionally relying on spatial labels like bounding boxes or segmentation masks. These requirements increase annotation costs and complicate integration with other modalities where spatial synchronization may be unavailable. In this work, we investigate the feasibility and effectiveness of scene graphs without location information, offering an alternative paradigm for scenarios where spatial data is unavailable. To this end, we propose the first method to generate location-free scene graphs, directly from images, evaluate their correctness and show the usefulness of such location-free scene graphs in several downstream tasks. Our proposed method, Pix2SG, models scene graph generation as an autoregressive sequence modeling task, predicting all instances and their relations as one output sequence. To enable evaluation without location matching, we propose a heuristic tree search algorithm that matches predicted scene graphs with ground truth graphs, bypassing the need for location-based metrics. We demonstrate the effectiveness of location-free scene graphs on three benchmark datasets and two downstream tasks — image retrieval and visual question answering showing they can achieve competitive performance with significantly less annotations. Our findings suggest that location-free scene graphs can still be generated and utilized effectively without location information, thus opening new avenues for scalable, structured and efficient visual representations, such as for multimodal scene understanding by reducing dependency on modality-specific annotations. The code will be made available upon acceptance. Our code is available at https://github.com/egeozsoy/LF-SGG.

1. Introduction

Humans have an innate ability to quickly abstract and interpret visual information, understanding complex scenes almost instantaneously. This involves recognizing individual objects and comprehending the relationships between them. In the field of computer vision, scene graph generation (SGG) has emerged as a powerful structured representation to emulate this capability, representing a scene as a graph where nodes correspond to entities and edges capture their relationships. To our knowledge, all previous methods [6, 21, 29, 33–35, 38–40, 46, 47] localize every node in the image and thus require object locations, such as bounding boxes or segmentation masks, in some part of their pipeline. This location information is indeed crucial for certain tasks, particularly those requiring precise spatial understanding, such as detailed object localization. However, the annotation for such scene graphs is costly, and the reliance on spatial data can hinder integration with other modalities—such as audio, text, or other signals, where synchronized spatial annotations are often unavailable. The annotation for such scene graphs is costly as it consists of two sub-tasks, scene graph annotation and bounding box or mask annotation. For the creation of the bounding box labels, the necessary time investment [30] has been reported as 42 seconds for a single bounding box, which breaks down into drawing (25.5 sec), quality verification (9 sec), and coverage verification (7.8 sec). The estimated workload to create location labels for the Visual Genome dataset [16] is substantial, amounting to approximately 1,993 person-days. One line of work that successfully reduces this annotation cost are weakly supervised methods [28, 45], which infer location from pre-trained detectors. However, they do not inherently solve the problem of dependency on location data. These methods still depend on the availability of a robust object detector that can accurately localize entities within the scene, which can be limiting in many domains.

We believe, even without any object locations, a scene graph can be a valuable and informative structured description of a scene, serving as a lightweight representation that could facilitate multimodal fusion. A location-free scene graph can be used to tackle many downstream tasks such as image retrieval, visual question answering, or image captioning [8, 9, 12, 14, 15, 43, 48]. To this end, in this work,

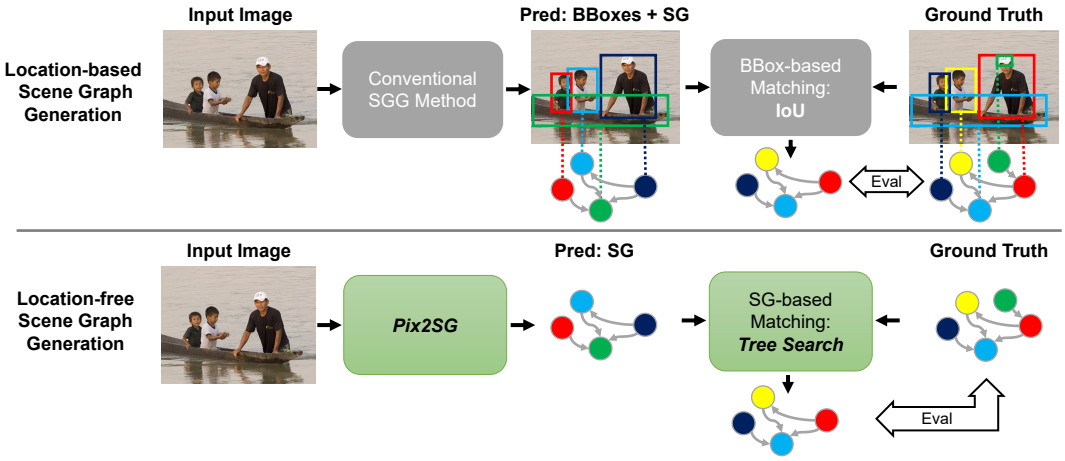

Figure 1. In the task of location-free scene graph generation, no location information such as bounding boxes or segmentation masks are available during training or validation. We design a new method, Pix2SG, leveraging autoregressive language modeling for congruent scene graph predictions, and a heuristic tree search algorithm for scene graph matching necessary for evaluation.

we want to answer the question if location information is essential for scene graphs and explore the potential of generating and utilizing scene graphs without location information, using our Pix2SG method the first location-free scene graph generation method, which does not rely on any spatial information throughout the entire pipeline. Location-free SGG differs fundamentally from both weakly supervised methods and tasks like image captioning. It predicts relationships between entities without anchoring them to specific pixel coordinates, fully decoupling scene understanding from spatial data. Unlike image captions, which are free-form, ambiguous and can lack granularity, location-free SGG provides a structured graph output, which is more suitable for tasks requiring precise understanding of inter-object relationships. However, location-free scene graph generation presents unique challenges. Traditional SGG methods rely on object locations to identify and extract entities, which becomes infeasible without spatial data. Without per node location information, a new representation is required to distinguish between multiple instances of the same class. Additionally, evaluating location-free scene graphs is complex, as current metrics based on spatial overlaps (e.g., IoU) are not applicable. This comparison becomes computationally expensive, leading to an NP-hard graph-to-graph matching problem when multiple instances are involved.

To address these challenges, we propose a set of innovative solutions that enable effective and accurate location-free SGG. First, we design a novel transformer-based sequence generation method that does not rely on location annotations at any stage, including training or validation. This method is designed to handle the complexities of a scene and produce a corresponding scene graph, abstracting spatial information into relational and semantic descriptors such as "close," "in front," or "over." Even without precise pixel-wise location information, our approach maintains the essential spatial context necessary for understanding object relationships and provides a useful structured description of the scene. To overcome the issue of differentiating between multiple instances of the same object class, our model explicitly predicts unique object identifiers, directly differentiating the instances from each other, without relying on location data. Evaluating location-free scene graph generation requires a novel approach, as traditional metrics based on spatial overlaps are no longer applicable. We design a task-specific efficient heuristic tree search algorithm to the NP-hard [11] problem of matching two graphs, providing a task-specific approximate solution particularly in scenarios involving multiple instances of the same object class. Extensive experiments on the PSG [42], Visual Genome [16] and 4D-OR [23] datasets demonstrate the performance of our proposed model on this new task. Our method outperforms most existing approaches, even those that rely heavily on location data, and demonstrates competitive performance on key downstream tasks such as image retrieval and Visual Question Answering (VQA).

In summary, we make the following major contributions:

- We identify the key challenges of location-free SGG, including the difficulty of entity identification, instance differentiation, and the limitations of traditional evaluation metrics in the absence of spatial data.
- We develop a novel transformer-based sequence generation method, which effectively addresses these complexities without using location information, enabling accurate relationship prediction and multi-instance differentiation.
- We design and implement a heuristic tree search algorithm for evaluation, approximately solving the NP-hard problem of graph matching, enabling the evaluation of

location-free scene graphs.
- We validate our method through extensive experiments on three scene graph generation datasets, PSG, Visual Genome and 4D-OR, and on two downstream tasks (image retrieval and VQA).

2. Related Work

2.1. SGG with Location Supervision

Previous methods for SGG have typically relied on two-stage architectures. The first stage consists of an object detector, often a pretrained Faster-R-CNN model [26]. The detected localized objects are used as proposals for the scene graph. The proposed objects and visual relationships between them are then classified in the second stage of the architecture, which can take the form of Iterative Message Passing [40, 45], Graph Neural Networks [21, 37, 39, 41, 47], and Recurrent Neural Networks [33, 46]. Lately, some works have tried to move away from this paradigm towards end-to-end approaches [6, 29, 35, 42], closely integrating the generation of localized objects and their relationships. These methods are sometimes described as detection-free because they omit an explicit object detector, but they still train to localize the entities in the output scene graph. In contrast, our approach eliminates the need for location data entirely, making it applicable to a broader range of domains, including those where location annotations are sparse or non-existent.

2.2. Weakly-Supervised Scene Graph Generation

While some works have attempted to reduce reliance on location data through weak supervision, these approaches still have significant limitations. Zareian et al. (VSPNet [45]) take object proposals from a pretrained object detector, build them into a semantic bipartite graph as a different formulation of a scene graph and classify and refine the entities through message passing. Shi et al. [28] use a weakly-supervised graph matching method to match object proposals from a pretrained object detector to the location-free scene graph labels. This matching generates a localized scene graph as a pseudo-label to train conventional fully-supervised SGG methods. Other methods utilize natural language from image captions as a weak supervision source, matching linguistic structures to object proposals that again come from pretrained object detectors [20, 44, 48]. We argue that relying on pretrained object detectors assumes the availability of a robust detector for the targeted domain, potentially limiting the applicability of WS-SGG methods.

While this approach does not require location labels in the scene graph datasets itself, a dataset from the same domain that contains location labels for all relevant objects is still required to pretrain the object detector. Shi et al. [28] notice a considerable domain gap for the detector pre-training. This manifests in subpar performance if Faster-R-CNN is pretrained on Open Images [18] and applied to Visual Genome [16] data, although both contain natural images. This issue becomes even more apparent when SGG is attempted in domains where large-scale image datasets are not readily available, such as the medical domain [23]. In contrast, location-free scene graph generation bypasses the need for such detectors altogether, offering a more generalizable solution across different domains.

2.3. Autoregressive Decoding

In response to the challenges posed by generating scene graphs without location data, we turn to autoregressive decoding [1, 3, 4, 7]—an approach that has proven effective in natural language processing and object detection—to sequentially predict relationships and entities, offering a robust alternative to traditional methods. We argue that we can leverage the advantages of autoregressive decoding also in the field of SGG, where there are significant interdependencies in the semantic structure of the scene graph, which can profit from more congruent sequential predictions.

3. Method

In this section, we introduce our novel architecture (Pix2SG) to generate location-free scene graphs without any reliance on location data, as well as our heuristic tree search-based matching algorithm that enables objective evaluation of location-free scene graphs.

3.1. Problem Formulation

Location-free scene graph generation is defined as the task of predicting a scene graph from a given image I without utilizing location information. The output is a graph $G = (V, E)$, where V represents to the entity nodes, and E to the pairwise relationships between entities. To objectively evaluate the predicted scene graph G, it must be matched to the ground truth graph G'. This matching is defined as

$$M\left(G, G'\right) = G_m \tag{1}$$

where G_m is the mapped graph prediction after fitting G to G'. The evaluation metric Recall@K $\mathcal{R}_K\left(G_m, G'\right)$ is then computed between G_m and G'. Unlike conventional SGG, this formulation does not rely on location information, making it a more generalized and challenging variant of scene graph generation.

3.2. Proposed Solution

We introduce Pix2SG as the first architecture designed specifically for location-free SGG, drawing inspiration from Pix2Seq [4]. Pix2SG employs an autoregressive approach, a well-established paradigm in natural language processing, to predict entities, their instances, and pairwise relationships directly from an image. The model sequentially

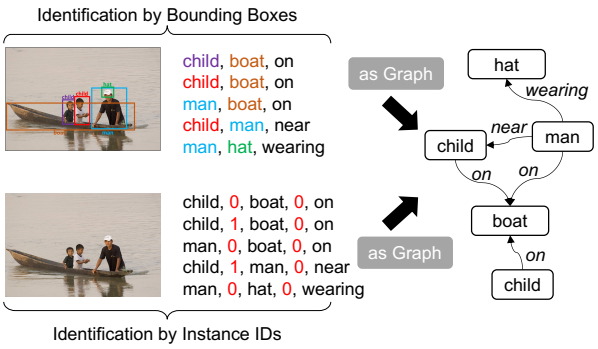

Figure 2. Comparison of existing location-based scene graph annotations to location-free scene graphs with instance identification and mapping to the graph representation.

predicts one token at a time until the entire scene graph is generated, as illustrated in Fig. 3. This autoregressive formulation allows the model to capture dependencies between tokens, improving the accuracy of predictions.

Vocabulary. To enable autoregressive sequence generation for scene graphs, we define a vocabulary of tokens that uniquely identify all entities and predicates. Each entity is represented by two tokens: one for the entity class and another for the entity instance, allowing differentiation of entities of the same class within an image. Predicates are represented by a single token $pred_{cls}$. A relationship between two entities is thus encoded as a quintuple:

$$(sub_{cls}, sub_{idx}, obj_{cls}, obj_{idx}, pred_{cls}) \tag{2}$$

where sub_{cls} and obj_{cls} represent the entity class and sub_{idx} and obj_{idx} the instance ids of the subject and object.

Ground Truth Sequence Generation. To train our proposed sequence generation model, we convert a scene graph label G' into a sequence of tokens from our vocabulary. This SG-sequence will be used in both the training and inference of our method. Scene graphs are first decomposed into individual quintuples as visualized in Fig. 2, and then these quintuples are concatenated in random order to form the SG-sequence. Entity IDs are assigned in ascending order to instances of the same class, ensuring consistent identification across the sequence. Our evaluation method is designed to be invariant to the order of quintuples and instance IDs, thereby eliminating ambiguities during validation.

Network. Our model architecture, depicted in Fig. 3, begins with an image encoding step that produces a flattened representation of the visual features. We extend the flattened image information with a positional encoding to allow the model to discern spatial information within the feature map. The flexibility of our approach allows us to use any feature extraction backbone, and does not restrict us to approaches trained for object detection or segmentation. During the decoding phase, the model uses the flattened image feature and a start token as inputs to generate the first output token. Each subsequent token is generated autoregressively, with the predicted tokens being appended to the sequence and used as input for the next prediction. This process continues until the entire output sequence, representing the scene graph, is complete (see. Fig. 2).

Inference. For inference, tokens are typically selected based on confidence values. However, this can lead to repetition issues in the SG-sequence. To mitigate this, we employ nucleus sampling [13], which introduces controlled randomness into token selection. During experiments, we generate a fixed number of tokens and convert the results into an output of N SG-quintuples (Eq. 2) with N defining the number of total relations. The autoregressive approach already encourages predicting the most prominent relationships first, before predicting more specific ones later in the sequence. Additionally, our model can predict a "stop token", to stop the generation when the graph is sufficiently detailed, a behavior learned from the training data, similar to practices in location-based SGG and image captioning.

3.3. Objective Evaluation Process

Objective evaluation in location-free SGG is challenging due to the complexity of correctly matching predicted scene graphs with ground truth graphs, particularly when multiple instances of the same class are present in a scene. As illustrated in Fig. 4, the task of graph matching is not straightforward; node identification has multiple possible solutions, and accurate matching requires considering the entire graph's structure. Existing approaches, such as exhaustive search, are computationally expensive, while the Hungarian matching algorithm [17] does not adequately capture the relational context needed for accurate graph matching. Therefore, we propose a novel algorithm (Alg. 1) to approximate an exhaustive search with a focus on matching quality, computational efficiency, and flexibility. Our proposed algorithm $M^*\left(G, G', B\right) = G_m^B$, uses a tree search approach with a branching factor B to control the depth of the search. To compute the overlap between graph nodes, we first examine the respective local one-hop neighborhoods of the nodes, represented as a list of (predicate, entity class) tuples. We calculate how many of these tuples are the same for both nodes, requiring both the predicate and the entity classes to match. This overlap score is then normalized by the node degree to ensure fairness across nodes with varying connectivity. For $B = 1$, the algorithm performs greedy matching, mapping each predicted entity to the ground truth entity with the highest local overlap. For $B > 1$, the algorithm explores multiple branches, representing alternative mappings, and recursively searches until all instances are matched. When $B \geq N$, where N is the number of instances of the most common entity class in the ground truth,

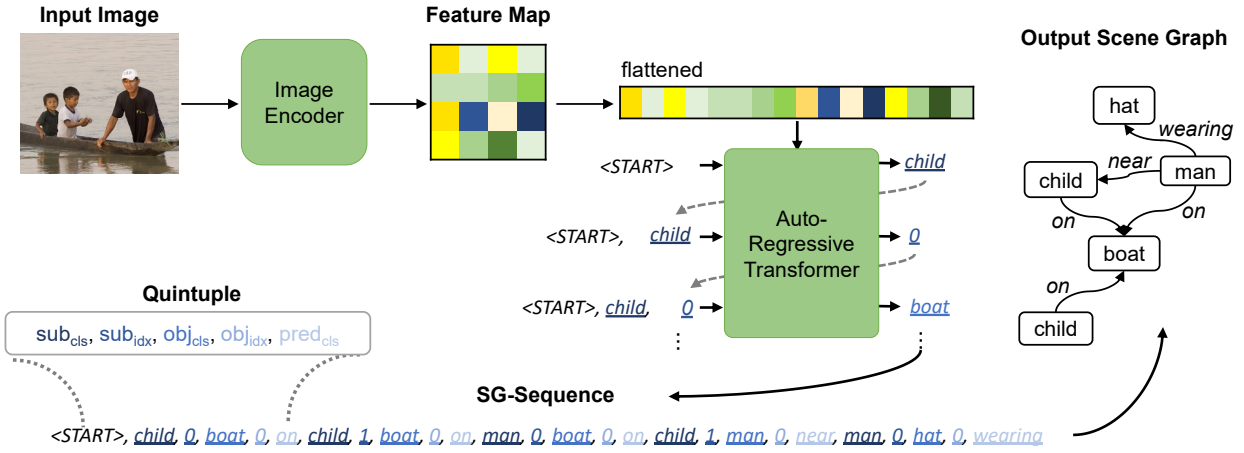

Figure 3. Pix2SG Architecture: An image encoder encodes the image as a feature map that is flattened and used as the input sequence to the autoregressive transformer module. The autoregressive transformer predicts the components of the scene graph, token by token, considering all its previous predictions until the output SG-sequence is completed.

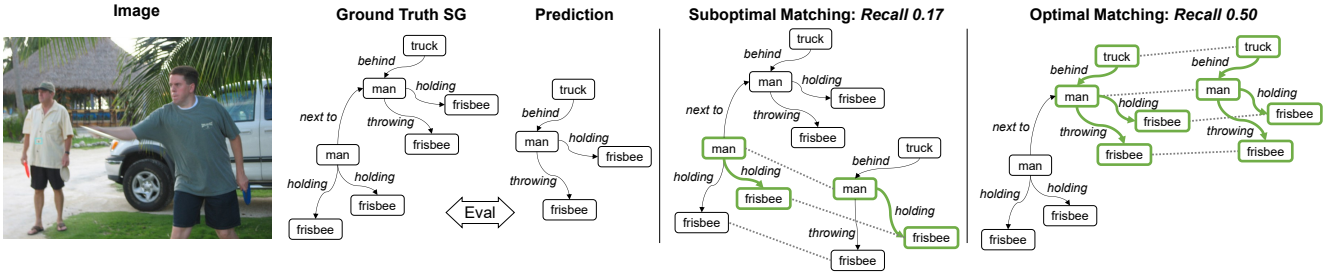

Figure 4. Illustration of the scene graph matching problem. Ground truth scene graph and prediction have to be correctly matched for the evaluation. A suboptimal matching can obscure the model actual performance.

the algorithm performs an exhaustive search, which guarantees finding the optimal solution. The result of our heuristic tree search is a set of graph matches between prediction and ground truth, and we subsequently select the match producing the highest evaluation metric as visualized in Fig. 4. We then use this matching to convert G to G_m.

4. Experiments

4.1. Datasets

Visual Genome (VG). The most frequently used scene graph dataset is VG [16] and it is considered as one of the main benchmarking datasets for SGG. As most previous works [35], we use a split of VG with the 150 most frequent objects and 50 predicates. While conventionally PredCls, SGCls, and SGGen are used as metrics, none are applicable to the case of location-free SGG. Instead, we evaluate all approaches with our proposed metric, where the recall is calculated directly by matching and comparing the predicted scene graph to the ground truth scene graph.

Panoptic Scene Graph Dataset (PSG). PSG [42] is a more recent dataset, designed for panoptic scene graph generation, where location information is not provided as bounding boxes, but as more accurate segmentation masks. Merging aspects of COCO and Visual Genome (VG), PSG consists of 49k images, annotating 133 objects and 56 unique predicates. They address some shortcomings of Visual Genome, such as redundant class and predicate labels, annotation of trivial relationships as well as duplicate localizations. The improvements over Visual Genome make it an interesting benchmark for evaluating scene graph generation.

4D-OR. 4D-OR [23] is a surgical scene graph dataset. Unlike VG and PSG, which are sparse in their annotations, 4D-OR includes dense annotations, enabling the calculation of precision in addition to recall. As it includes images from multiple views per scene, it allows us to demonstrate the extension of our method for multiple image inputs per scene. Finally, as the dataset size is an order of magnitude smaller than VG and PSG, it allows us to evaluate the performance

Algorithm 1 Heuristic Tree Search (HTS)

Input: gt graph y, predicted graph $\hat{y}$, branching factor B
Output: best mapping m_{best}
function HTS($y, \hat{y}, B, m$)
  $y\_inst \leftarrow$ instance from y with highest node degree
  $\hat{y}\_insts \leftarrow$ instances from $\hat{y}$ with same class as y_inst
  $y\_nbhd \leftarrow$ connected nodes and edges of y_inst
  for $\hat{y}\_inst$ in $\hat{y}\_insts$ **do**
    $\hat{y}\_nbhd \leftarrow$ connected nodes and edges of $\hat{y}\_inst$
    $overlaps \leftarrow$ append $|y\_nbhd \cap \hat{y}\_nbhd|/|y\_nbhd|$
  end for
  $\hat{y}\_insts \leftarrow \hat{y}\_insts$ sorted by $overlaps$
  $M \leftarrow \varnothing$ set for branched mappings m_i
  for $i = 0 : B$ **do** $m_i \leftarrow m \cup (y\_inst \mapsto \hat{y}\_insts[i])$
    if $y \backslash y\_inst == \varnothing$ **then** $M \leftarrow M \cup m_i$
    else $M \leftarrow M \cup$ HTS$(y \backslash y_inst, \hat{y} \backslash \hat{y}_insts[i], B, m_i)$
    end if
  end for
  return M
$m_{best} \leftarrow$ select highest recall from HTS$(y, \hat{y}, B, \varnothing)$

of our method in lower data regimes.

4.2. Downstream Tasks

Image Retrieval. We assess the utility of our location-free scene graphs in the task of image retrieval, specifically using the Sentence-to-Graph Retrieval (S2GR) methodology introduced by [34]. S2GR first converts image captions into scene graphs, and learns to match them with image scene graphs. The goal is to find the correct image, given an image pool of 1000 or 5000 images, measured using R@20 and R@100. This task is deliberately designed to avoid using image features, and instead only focuses on the graphs. We follow their implementation very closely, and only replace their image scene graphs by our own location-free scene graphs generated using Pix2SG VIT-L.

VQA. We further evaluate location-free scene graphs for the task of visual question answering. Concretely, we use the COCOVQA [2] dataset, which consists of multiple question answer pairs for each image. While existing methods rely directly on image features and supervised training, our task is zero-shot, and does not use any labeled data for VQA. We start by using a SGG method to compute scene graphs for the images, and then feed the scene graphs as text (list of triplets) into a Large Language Model (LLM) as a part of the prompt for question answering. We use the following prompt: "Given is the following scene graph for an image: <SG>. Answer the following benchmark question. If you are unsure, make an educated guess. Don't give explanations, your output will be automatically evaluated. You try to maximize your score on the benchmark. Answers mostly consists of one or two words, be concise. Question: <Q>", where we replace <SG> and <Q> with the respective image scene graph and question. This way, an existing LLM can be directly utilized for reasoning about an image. We evaluate the accuracy of the SGG methods on all categories, and the three subcategories, open end, number, and yes/no.

4.3. Implementation details

We use both EfficientNet [31] pretrained on Imagenet [27] as well as Vision Transformer [10] with contrastive pretraining [25] as image encoder backbones. We resize the images to match the input dimensions of the backbones. In 4D-OR, the four multi-view images per scene are processed individually by the feature extraction backbone, then the feature maps are flattened and concatenated to build the input sequence. We use pix2seq [4] as the starting point of the autoregressive sequence modeling implementation. We use a categorical cross-entropy loss, with the entire vocabulary as target classes, and optimize our model with AdamW [22] and a constant learning rate of 4×10^{-5} with weight decay of 1×10^{-4}. The batch size is set to 16 in all our experiments and we train our methods for 200 epochs, employing early stopping. We use a Transformer [36], with a hidden size of 256, eight attention heads, six encoding, and two decoding layers. Unless otherwise specified, we predict 300 relations using nucleus sampling [13] with a p-value of 0.95 and pick the top K unique predictions for Recall@K. We set the branching factor B of our proposed heuristic three-matching algorithm to 3 for all the validation experiments except when indicated otherwise. For the baseline methods on PSG, Visual Genome, and 4D-OR, we use the implementations provided by [42], [32] and [23] respectively. Finally, we provide an efficient implementation of our Heuristic Tree Search based evaluation algorithm in C++, with a sub-second run time for most samples using three as Branching-factor B. We empirically motivate the choice of B in Sec. 4.5 and Fig. 7. We use Vicuna 13B variant[5] as the zero-shot LLM for the VQA task. All of our experiments are done on a single NVIDIA A40 GPU.

4.4. Results

Panoptic Scene Graph Dataset (PSG). As we introduce a new task and a new metric, we first reevaluate existing methods trained on the task of SGG with segmentation masks, with our location-free SGG evaluation method. While their reliance on masks makes them not directly comparable to our method in the task of location-free SGG, we still provide these results as a rough but valuable guideline. We then evaluate our approach, Pix2SG, which is trained and evaluated without any location labels. We present these results in Tab. 1. As we are the first and only method to not require location information at any stage, our method and results serve as the first baseline for the new task of location-free SGG. While using much less annotations, Pix2SG out-

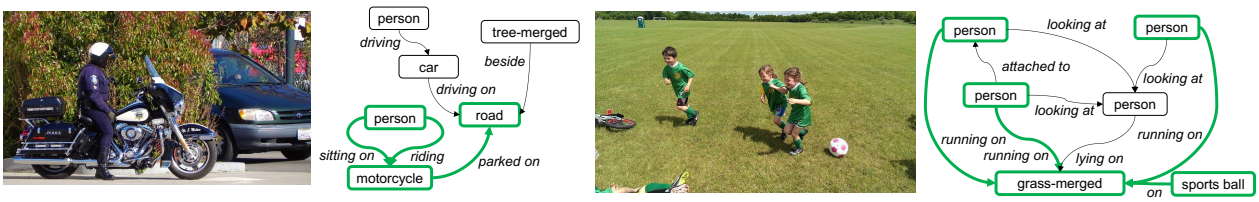

Figure 5. Qualitative Results of Pix2SG on the Panoptic Scene Graph Dataset. Images and corresponding Ground Truth Scene Graphs are shown. Nodes and edges correctly predicted by our model are highlighted in green. Additional triplets are predicted which are not in the ground truth but meaningful.

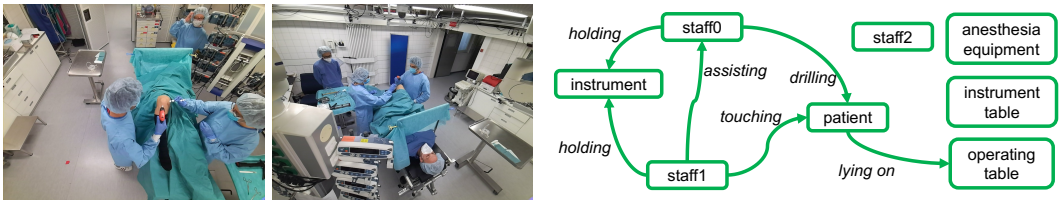

Figure 6. Qualitative Result of Pix2SG on 4D-OR dataset. Images from two of the six viewing angles and the corresponding ground truth scene graphs are displayed. Nodes and edges correctly predicted by our model are highlighted in green.

Table 1. Location-free SGG results of different SG models at R@k on PSG dataset. Checkmarks indicate no mask supervision was used during model training. B5, B7, VIT-B, VIT-L represent the corresponding EfficientNet and Vision Transformer backbones we used in our model.

Model	Location-free	R@20	R@50	R@100
IMP [40]		25.38	29.46	31.06
GPSNet [21]		25.96	30.03	31.91
PSGFormer [42]		26.10	31.47	34.75
MOTIFS [46]		30.50	34.68	36.30
VCTree [33]		31.72	36.28	37.99
PSGTR [42]		**39.25**	**43.95**	**44.11**
Pix2SG B5(Ours)	✓	29.07	34.63	37.00
Pix2SG B7(Ours)	✓	30.66	34.28	35.92
Pix2SG VIT-B(Ours)	✓	33.35	38.10	39.93
Pix2SG VIT-L(Ours)	✓	**35.54**	**40.40**	**41.72**

Table 2. Location-free SGG results of different SG models at R@k on Visual Genome dataset. B5, B7, VIT-B, VIT-L represent the corresponding EfficientNet and Vision Transformer backbones we used in our model.

Model	Location-free	R@20	R@50	R@100
IMP [40]		21.66	30.78	37.07
SS-R-CNN [35]		22.09	26.43	28.57
SGTR [19]		23.62	30.38	34.85
RelTR [6]		25.86	30.99	33.31
VCTree [33]		27.06	35.59	41.21
Transformer [34]		28.79	37.81	43.69
MOTIFS [46]		**29.02**	**38.08**	**43.64**
Pix2SG B5(Ours)	✓	19.32	23.59	25.47
Pix2SG B7(Ours)	✓	21.51	24.81	26.66
Pix2SG VIT-B(Ours)	✓	22.10	25.65	28.64
Pix2SG VIT-L(Ours)	✓	**22.98**	**26.92**	**30.05**

Table 3. Location-free SGG results on 4D-OR dataset. B5 and B7 represent the corresponding EfficientNet backbones we used in our model.

Model	Location-free	Temporality	Prec.	Rec	F1
4D-OR baseline [23]			0.68	0.87	0.75
LABRAD-OR [24]		✓	0.87	0.90	0.88
Pix2SG B5(Ours)	✓		0.88	0.92	0.90
Pix2SG B7(Ours)	✓		**0.89**	**0.94**	**0.91**

Table 4. Image retrieval results on Visual Genome. Gallery size refers to the number of images in the image pool from which one image is retrieved.

Gallery Size		1000		5000	
Model	Location-free	R@20	R@100	R@20	R@100
MOTIFS [46]		20.8	59.2	05.2	21.3
VCTree [33]		19.1	55.5	05.1	20.3
Pix2SG VIT-L(Ours)	✓	**38.3**	**73.9**	**12.7**	**39.8**

Table 5. Visual Question Answering results on COCOVQA [2].

Model	Location-free	Open.	Num.	Yes/No	Overall
IMP [40]		26.65	32.17	66.44	42.32
PSGTR [42]		**29.03**	**32.58**	67.34	**43.89**
Pix2SG VIT-L(Ours)	✓	28.27	32.23	**67.62**	43.57

Table 6. Effect of nucleus sampling with a p-value of 0.95 on VG compared to conventional maximum likelihood selection for location-free SGG.

Sampling	R@20	R@50	R@100
Maximum Likelihood	19.05	21.18	23.19
Nucleus [13]	21.51	24.81	26.66

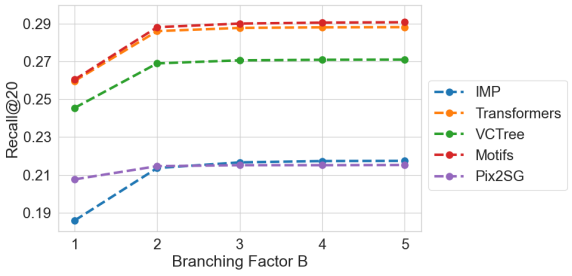

Figure 7. Ablation of Branching factor with $B = 3$ providing a good trade-off between speed and matching performance.

performs every method except PSGTR [42], validating our location-free scene graph generation architecture. In addition to the quantitative results, we also provide qualitative results in Fig. 5, and in the supplementary material visualize the attention maps for three quintuple predictions, illustrating that our model attends to relevant parts of the image for predicting relationships, acting as a sanity check.

Visual Genome (VG). Similar to PSG, we also reevaluate existing methods trained on the task of SGG with bounding boxes with our location-free evaluation method. While the results are again not directly comparable, they provide an additional data point. We present these results in Tab. 2. We see that without location supervision, Pix2SG performs comparatively worse on VG than on PSG. We think the discrepancy is mainly caused by lower quality and decreased consistency of the scene graph annotations on Visual Genome, and argue that, as our autoregressive formulation can exploit interrelation dependencies between entities, it thrives where label consistency is maintained.

4D-OR. As the evaluation proposed in 4D-OR [23] is applicable to our method, we directly compare our method to the existing results in Tab. 3. We not only significantly improve upon the existing single frame baseline, from 75% F1 to 91% F1, we even outperform the SOTA, which utilizes both visual and temporal information. These results not only support our theory regarding the importance of label consistency, but also validate our approach in a different and unique domain, which signifies the transferability and adaptability of Pix2SG. Importantly, we achieve this without using bounding boxes, depth, or 3D point clouds, which are all used by the existing methods. We provide qualitative results for 4D-OR in Fig. 6.

Image Retrieval. In the task of image retrieval, Pix2SG outperforms existing methods by a large margin, even though it uses less labels during training, as can be see in Tab. 4. This supports our motivation of introducing location-free SGG, as they can be just as useful as location based scene graphs for some downstream tasks.

VQA. We present the results on the task of zero shot VQA in Tab. 5. We again observe that our method performs very comparable to both location based scene graph generation methods. This again supports the wider use of location-free SGG for downstream tasks not directly requiring accurate pixel location information.

4.5. Ablation Studies

We perform ablation studies to validate the performance of our evaluation method, as well as to demonstrate the importance of nucleus sampling. In Fig. 7, we evaluate five different branching factors for our tree search-based matching on the four baseline methods, IMP, VCTree, Motifs, Transformers and Pix2SG on Visual Genome. The results show that our algorithm converges at $B = 3$, providing a good balance between speed and matching performance. We, therefore, set B to 3 in all other experiments. In Tab. 6, we show the importance of nucleus sampling by comparing it against always choosing the token with the highest probability. By reducing repetition and increasing variance, it leads to significantly higher recall in all thresholds.

4.6. Limitations

Our method, Pix2SG, excels in generating scene graphs without spatial data, offering a scalable and annotation-efficient alternative to traditional approaches. While this capability broadens its applicability, it may not fully address tasks requiring precise localization, such as object detection or fine-grained spatial reasoning. This reflects our design choice to prioritize reduced annotation costs, tailoring it to scenarios where spatial precision is less critical.

5. Conclusion

In this work, we introduced the first location-free scene graph generation method, Pix2SG, to generate scene graphs without relying on spatial data at any stage. By leveraging autoregressive sequence modeling, Pix2SG achieves competitive results on standard benchmarks, even in the absence of location information. To facilitate evaluation, we propose an efficient heuristic tree search algorithm that enables the robust assessment of location-free scene graphs. Our findings in this work demonstrate that location information is not a requirement for generating useful and accurate scene graphs. This opens up new opportunities for applying SGG in contexts where location data is either unavailable, incomplete, or too costly to annotate. By decoupling scene graph generation from the need for location information, our approach paves the way for integrating visual scene understanding with other modalities, such as text, audio and other signals, in multimodal learning frameworks, broadening the applicability of SGG to a wider range of domains, and make it more accessible and scalable, particularly in resource-constrained environments.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a Visual Language Model for Few-Shot Learning, 2022. arXiv:2204.14198 [cs]. 3

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015. 6, 7

[3] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners, 2020. arXiv:2005.14165 [cs]. 3

[4] Ting Chen, Saurabh Saxena, Lala Li, David J. Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. In *ICLR*, 2022. 3, 6, 1

[5] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 6

[6] Yuren Cong, Michael Ying Yang, and Bodo Rosenhahn. RelTR: Relation Transformer for Scene Graph Generation. *arXiv:2201.11460 [cs]*, 2022. arXiv: 2201.11460. 1, 3, 7

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019. arXiv:1810.04805 [cs]. 3

[8] Helisa Dhamo, Azade Farshad, Iro Laina, Nassir Navab, Gregory D. Hager, Federico Tombari, and Christian Rupprecht. Semantic Image Manipulation Using Scene Graphs. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5212–5221, Seattle, WA, USA, 2020. IEEE. 1

[9] Helisa Dhamo, Fabian Manhardt, Nassir Navab, and Federico Tombari. Graph-to-3D: End-to-End Generation and Manipulation of 3D Scenes Using Scene Graphs. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16332–16341, Montreal, QC, Canada, 2021. IEEE. 1

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 6

[11] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):377–388, 1996. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence. 2

[12] Jiuxiang Gu, Shafiq Joty, Jianfei Cai, Handong Zhao, Xu Yang, and Gang Wang. Unpaired image captioning via scene graph alignments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10323–10332, 2019. 1

[13] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The Curious Case of Neural Text Degeneration, 2020. arXiv:1904.09751 [cs]. 4, 6, 7, 1

[14] Drew A. Hudson and Christopher D. Manning. Learning by Abstraction: The Neural State Machine, 2019. arXiv:1907.03950 [cs]. 1

[15] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3668–3678, Boston, MA, USA, 2015. IEEE. 1

[16] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations, 2016. arXiv:1602.07332 [cs]. 1, 2, 3, 5

[17] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955. 4

[18] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. arXiv:1811.00982 [cs]. 3

[19] Rongjie Li, Songyang Zhang, and Xuming He. SGTR: End-to-end Scene Graph Generation with Transformer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19464–19474, New Orleans, LA, USA, 2022. IEEE. 7, 1

[20] Xingchen Li, Long Chen, Wenbo Ma, Yi Yang, and Jun Xiao. Integrating Object-aware and Interaction-aware Knowledge for Weakly Supervised Scene Graph Generation. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4204–4213, New York, NY, USA, 2022. Association for Computing Machinery. 3

[21] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. GPS-Net: Graph Property Sensing Network for Scene Graph Generation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3743–3752, Seattle, WA, USA, 2020. IEEE. 1, 3, 7

[22] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, 2019. arXiv:1711.05101 [cs, math]. 6

[23] Ege Özsoy, Evin Pınar Örnek, Ulrich Eck, Tobias Czempiel, Federico Tombari, and Nassir Navab. 4d-or: Semantic scene graphs for or domain modeling. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*, pages 475–485, Cham, 2022. Springer Nature Switzerland. 2, 3, 5, 6, 7, 8

[24] Ege Özsoy, Tobias Czempiel, Felix Holm, Chantal Pellegrini, and Nassir Navab. Labrad-or: Lightweight memory scene graphs for accurate bimodal reasoning in dynamic operating rooms. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2023. 7

[25] 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. 6

[26] 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*, 2015. 3

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015. 6

[28] Jing Shi, Yiwu Zhong, Ning Xu, Yin Li, and Chenliang Xu. A Simple Baseline for Weakly-Supervised Scene Graph Generation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16373–16382, Montreal, QC, Canada, 2021. IEEE. 1, 3

[29] Suprosanna Shit, Rajat Koner, Bastian Wittmann, Johannes Paetzold, Ivan Ezhov, Hongwei Li, Jiazhen Pan, Sahand Sharifzadeh, Georgios Kaissis, Volker Tresp, and Bjoern Menze. Relationformer: A unified framework for image-to-graph generation. In *ECCV*, 2022. 1, 3

[30] Hao Su, Jia Deng, and Li Fei-Fei. Crowdsourcing Annotations for Visual Object Detection. In *HCOMP@AAAI*, 2012. 1

[31] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019. 6

[32] Kaihua Tang. A Scene Graph Generation Codebase in PyTorch, 2020. 6

[33] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to Compose Dynamic Tree Structures for Visual Contexts. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6612–6621, Long Beach, CA, USA, 2019. IEEE. 1, 3, 7

[34] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased Scene Graph Generation From Biased Training. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3713–3722, Seattle, WA, USA, 2020. IEEE. 6, 7, 1

[35] Yao Teng and Limin Wang. Structured Sparse R-CNN for Direct Scene Graph Generation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19415–19424, New Orleans, LA, USA, 2022. IEEE. 1, 3, 5, 7

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need, 2017. arXiv:1706.03762 [cs]. 6

[37] Johanna Wald, Helisa Dhamo, Nassir Navab, and Federico Tombari. Learning 3d semantic scene graphs from 3d indoor reconstructions. In *CVPR*, 2020. 3

[38] Johanna Wald, Nassir Navab, and Federico Tombari. Learning 3D Semantic Scene Graphs with Instance Embeddings. *International Journal of Computer Vision*, 2022. 1

[39] Shun-Cheng Wu, Johanna Wald, Keisuke Tateno, Nassir Navab, and Federico Tombari. SceneGraphFusion: Incremental 3D Scene Graph Prediction from RGB-D Sequences. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7511–7521, Nashville, TN, USA, 2021. IEEE. 3

[40] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. Scene Graph Generation by Iterative Message Passing. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3097–3106, 2017. ISSN: 1063-6919. 1, 3, 7

[41] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *ECCV*, 2018. 3

[42] Jingkang Yang, Yi Zhe Ang, Zujin Guo, Kaiyang Zhou, Wayne Zhang, and Ziwei Liu. Panoptic scene graph generation. In *ECCV*, 2022. 2, 3, 5, 6, 7, 8

[43] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10685–10694, 2019. 1

[44] Keren Ye and Adriana Kovashka. Linguistic Structures as Weak Supervision for Visual Scene Graph Generation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8285–8295, Nashville, TN, USA, 2021. IEEE. 3

[45] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Weakly Supervised Visual Semantic Parsing. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3733–3742, Seattle, WA, USA, 2020. IEEE. 1, 3

[46] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural Motifs: Scene Graph Parsing with Global Context. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5831–5840, Salt Lake City, UT, 2018. IEEE. 1, 3, 7

[47] Ji Zhang, Kevin J. Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. Graphical Contrastive Losses for Scene Graph Parsing. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11527–11535, 2019. ISSN: 2575-7075. 1, 3

[48] Yiwu Zhong, Jing Shi, Jianwei Yang, Chenliang Xu, and Yin Li. Learning to Generate Scene Graph from Natural Language Supervision. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1803–1814, Montreal, QC, Canada, 2021. IEEE. 1, 3