Knowledge Informed Sequential Scene Graph Verification Using VQA

Dao Thauvin, Stéphane Herbin
ONERA - DTIS
Université Paris Saclay, F-91123 Palaiseau - France
firstname.lastname@onera.fr

Abstract

We propose a new task, ***non localized scene graph verification****, whose objective is to provide a justified expression of inconsistencies between the visual content of the image and its non-localized scene graph in order to diagnose errors or anticipate corrections. We introduce a sequential algorithm capable of detecting and proposing plausible corrections, taking into account the information already present in the scene graph and exploiting knowledge priors. Instead of relying on object detection that requires bounding box annotations, we use a simple visual question answering (VQA) as a proxy for visual content analysis. We show on the VG150 dataset that our strategy is efficient compared to a baseline adapted from a caption editing approach. We also show that our algorithm is able to efficiently correct corrupted scene graphs.*

1. Introduction

In this article, we address the task of automatically characterizing a scene graph by providing an evaluation of its quality in an interpretable format. This goal can be useful for filtering an annotation, for detecting inconsistencies between an image and some associated text or caption, for identifying the nature of the discrepancies between the image's visual content and a current annotation, for example in an interactive sequential loop, or for updating a textual medical diagnosis from a new image or helping the physician produce higher quality reports [35].

The quality of an image description can be assessed along two dimensions: its relevance and its accuracy. Relevance is related to how and for what purpose it can be used (medical diagnosis, industrial process monitoring, robotics...) Accuracy is related to truthfulness or veracity and characterizes whether an image description is true or false, given a relevant domain vocabulary for its description, whether it is *consistent* with the image content. In this paper, we focus on inaccuracy detection and explore the possibility of detecting and characterizing inconsistencies.

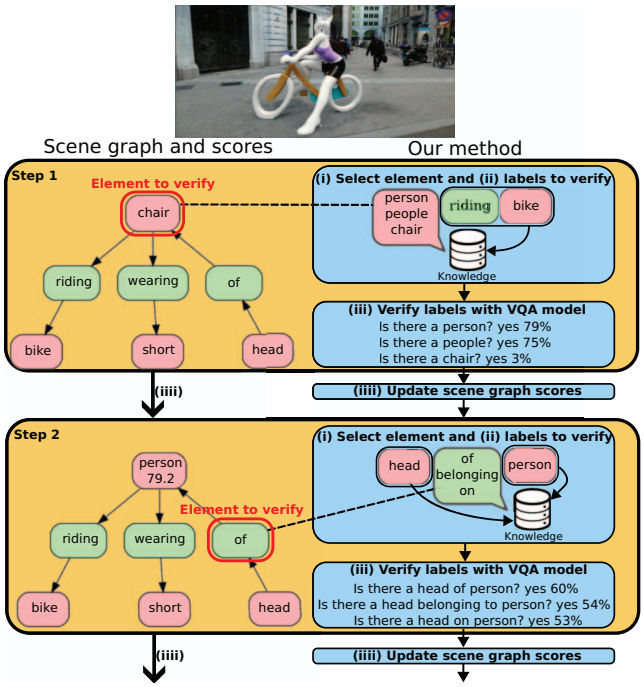

Figure 1. Sketch of the scene graph verification principle. Starting with the scene graph to be verified (on the left with objects in red and relations in green), our method verifies the scene graph components sequentially (here two steps are represented). It first searches an element to verify in the graph (i) and find plausible labels for it (ii). Then it verifies the labels with a VQA model and a yes/no question (iii). All the scores are sequentially gathered to provide consistency scores for a selection of plausible alternate label hypotheses (iiii). The final scores allow to detect inconsistencies (original labels with low scores) and potential corrections (label with high scores).

Representing the visual content of an image can take several forms: simple global labels, free-form captions, or fine-grained attributes. Classic visual content consists of describing the entities (objects, people, animals) present in the image, their attributes, and their relationships. A non-localized Scene Graph (SG) [19] is a formal structure that has been proposed to encode entities as nodes and their re-

lations as edges, both tagged with a label. Compared to a free-form textual description such as a caption, an SG provides a more compact content description, is a convenient computational structure, and relies on a smaller vocabulary for the same or even better level of expressiveness [26]. However, it still has to deal with lexical issues such as polysemy (multiple meanings for the same word), synonyms (multiple words for the same meaning), or hyper/hyponymy (some words refer to more general concepts than others) [4]. In order to keep the idea of a "pure" image content description, we do not consider in an SG the relation to detected bounding boxes, their image grounding, making it a lightweight structure that can be easily obtained from free-form text [38] compared to localized scene graphs that require links to bounding boxes in the image.

We express the main output of the verification process as a distribution of possible alternative labels associated with consistency scores for each node and edge (Fig. 2). The idea of using an intermediate representation with a distribution of scores is to maintain a certain level of uncertainty in the qualification of the scene graph and a flexibility of use: to filter or detect inconsistencies, they can be estimated by detecting a low consistency score relative to other alternatives for the labels of the verified graph; to support the construction of the scene graph, plausible label suggestions for an element can be extracted by looking at labels with high scores.

Our approach to the verification task is to combine two algorithmic components that interact in a sequential process: a knowledge base that represents co-occurrence priors of entities and relations, and a Visual Question Answering (VQA) system that answers simple yes/no questions about the visual content. The computation of scores for the whole graph is realized sequentially by checking each element one by one and updating the score for relevant labels by combining the outputs of these two components (Fig. 1).

The idea of using two rather independent components for the verification process is to make explicit, and therefore more transparent, the combination of two sources of uncertainty: visual and knowledge-based. Indeed, some entities or relations are more difficult to verify visually than others: the confidence level of the VQA output may be low, even for yes/no questions, which are known to have higher performance than what/where types of questions (the performance of yes/no questions in recent VQA challenges [2] is now over 90%). In some situations, visual uncertainty can be compensated by exploiting prior knowledge that encodes the probability of occurrence of entities or relations given their context: for example, given that an entity to be checked is *riding a bike* (Fig. 1), it is more likely that this entity is a person or even an animal than a chair.

The knowledge base can also be used to guide the choice of good yes/no visual questions to ask in the sequential process: keeping the same example from Fig. 1, the role of the knowledge base is to identify the most likely topics that are consistent with *riding a bike*, thus limiting the hypotheses to be visually checked. Here, the knowledge base is used as a generator of hypotheses, not as an uncertainty scoring function.

We make the following contributions in our study:

- We introduce the problem of image description verification and formulate it as the computation of a distribution of possible labels with consistency scores for nodes and edges of a scene graph;
- We compute consistency scores using a sequential process that combines knowledge priors and scored answers to simple questions about the visual content of the scene;
- We exploit a knowledge representation to limit the hypothesis space during the verification process;
- We demonstrate on Visual Genome [23] the efficiency of our approach over a series of competing baselines such as caption editing [44] algorithms and compare our approach to state-of-the-art scene graph generation from scratch [54].

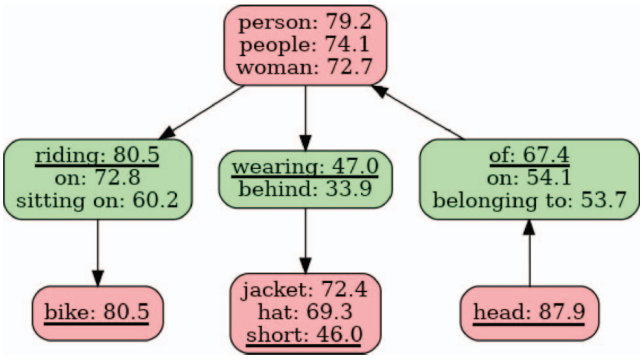

Figure 2. Scores obtained for Fig. 1 erroneous graph. It displays the 3 classes with the highest scores for each element (the scores are multiplied by 100 for better visibility).

2. Related Work

Image Description Assessment. The automated analysis of image description has attracted very few studies. The closer problem to ours is FOIL it! [40] that proposes a task of detection and a task of correction of an erroneous word in a textual image caption. The main motivation of the paper was to understand if Visual Question Answering [3] models were really able to understand text-image interactions. In a subsequent paper [32] showed that error detection was already possible without using image information. It is this observation that motivated us to formalize the verification

step as a consistency score function exploiting different resources: image and knowledge base. Unlike FOIL it!, we focus on image description in the form of a scene graph instead of a free form textual description.

Image captioning editing, i.e. the task of correcting a given caption to produce a more consistent one, is also close to image description but its goal is not to assess the quality of the caption in a declarative way with clear justification, but to produce a new description expected to be more accurate. It has been addressed in [37] who propose a sequential LSTM based scheme to generate the new caption. In a recent work [45] formulates image caption editing as a sequence of actions – which is just a mean of representing a difference between two structures – and uses VilBert [29], a cross-attention transformer. Our solution relies on a simpler and more transparent structure and gives better results for the problem of image description verification.

A related problem is to express in a textual form the difference between two images (Change captioning) [51, 42, 34, 17, 36]: however, although the output of this problem is to assess a difference between two data, it doesn't address inter-modality nor semantic issues as we do.

Scene Graphs and External Knowledge. Scene graphs are used to represent the elements of an image and their relations in a structured way. This representation allows in particular to better represent the relations in the images [26]. They are used in particular in VQA and captioning where it is necessary to carry out complex reasoning on an image that requires to take into account the relations between elements of the image. The structure of the graphs also allows to better control their content, in particular when generating images [20, 48, 46] and outpainting images [48], or for captioning where they allow to isolate independent image subparts to generate different captions [56, 50] or to control the lexical form of the captions [6]. Scene graphs also makes it easier to add external knowledge in the form of knowledge graph [25, 8, 21, 52, 13, 18] or statistics on graph [54, 7, 9, 39]. [14] uses ConceptNet to generate weighted triplet hypotheses in a setting close to ours.

Another way to introduce external knowledge is to exploit priors on scene graphs: [53] uses a transformer based filter to correct globally a given SG to make it more commonsense; [10] defines a conditional auto-regressive generative model [10] able to generate a complete SG from a given sub-graph. However these two approaches do not check if the generated or corrected scene graphs are consistent with the image, only if they are likely.

Sequential Scene Graph Processing. Processing a scene graph sequentially is used in VQA to make complex reasoning using a generated program. Following the words of the question, the model move its attention in the scene graph [41, 16, 55], the final attention of the model allows then to answer the question. This process allows a better understanding of the reasoning process. However, the program is generated using the question, making it difficult to use in other tasks where no text is available. Reinforcement Learning is another sequential scene graph processing proposed in a variety of task such as scene graph generation [27, 30], image captioning [31], VQA [15] and visual curiosity [49]. The work most similar to ours is [49], where a model is learned by reinforcement to ask good questions to an oracle – not an uncertain VQA model – about the inconsistencies of a generated scene graph. Compared to our work, the scene graph is localized and the objective is to optimize the interaction with the oracle that provides the ground truth, a variant of active learning.

3. The task

3.1. Non Localized Scene Graphs

A non localized scene graph G associated to an image I is defined as a directed graph $G = ((E, V), C_E, C_V)$. It contains (E, V) whose nodes $v \in V$ correspond to entities of I and edges $e = (s(e), o(e)) \in E$ correspond to a relation or predicate between a subject entity $s(e) \in V$, the head of e, and an object entity $o(e) \in V$, its tail, an entity being able to be subject and object of several relations in the graph. The elements of a scene graph are characterized, *colored*, by classes. The function $C_E : E \to \mathcal{L}_E$ associates to each edge $e \in E$ a class $C_E(e) \in \mathcal{L}_E$, where $\mathcal{L}_E$ is the set of possible categories of relations, the vocabulary of relations. Similarly for nodes, $C_V : V \to \mathcal{L}_V$ associates each node $v \in V$ to a class $C_V(v) \in \mathcal{L}_V$ where $\mathcal{L}_V$ is the vocabulary characterizing the entities. To simplify,$C(u)$ associates each element to a class. The tuple instantiating the subject-relation-object with class labels, which can be indexed by the edge e: $(C_V(s(e)), C_E(e), C_V(o(e)))$ is customary called a triplet in scene graph literature and represents the encoding of a simple sentence describing one aspect of the image content.

3.2. Non Localized Scene Graph Verification

The objective of Non localized scene graph verification (NL-SGV) is to detect inconsistencies in an image description expressed as a non localized scene graph and to propose plausible corrections. The goal of the verification algorithm `verif` is therefore to generate a consistency score function S that characterizes the possibly erroneous non localized scene graph G that describes the image I: $S = \texttt{verif}(G, I)$. We focus on inaccuracy detection, i.e. on class labels that can be considered untrue because they do not refer meaningfully to any entity (when considering nodes) or relation between entities (when considering edges) in the image. The decision that a label is wrong is based on a distribution of alternate plausible values, each characterized by a consistency score as shown in Fig. 2.

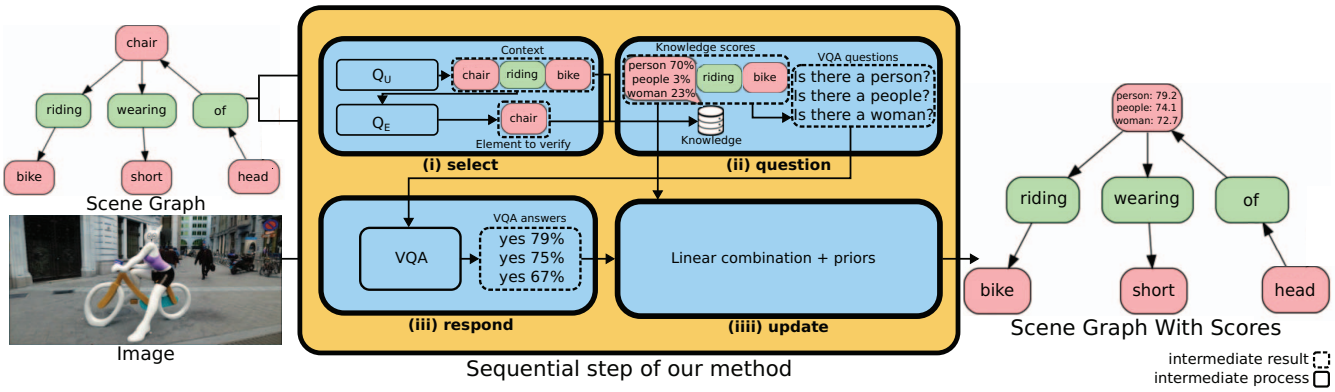

Figure 3. An overview of a step of our model. Dotted squares are intermediate outputs and normal squares are modules and sub modules.

The consistency score function S maps any component of the scene graph and any class label to a score taking a real value in $[0, 1]$ or the default value `NA` when no such a score is available. The higher the value, the more plausible the label.

4. Our approach

4.1. Consistency Score Function

Our approach solves NL-SGV in a sequential way. At each step, the process verifies a limited number K of label hypothesis about a selected element of the scene graph. After the end of the t-th iterative step, the consistency scores are updated by two functions for nodes and edges that assign a new consistency score for a class label hypothesis, $S_V^t : V \times C_V \rightarrow [0, 1] \cup \texttt{NA}$ and $S_E^t : E \times C_E \rightarrow [0, 1] \cup \texttt{NA}$. We also use a simplified notation S^t that makes no distinction between nodes and edges. A `NA` value means that the consistency score is unknown for that class label.

The consistency score when an element of the graph u is characterized by the class c is noted $S^t(u, c)$, whether this element is a node-entity or an edge-relation. This score is estimated from a VQA and priors obtained from an external knowledge base. The scene graph inconsistencies are identified after analyzing the final consistency scores.

To observe intermediate stages, we also define two functions $C_V^t : V \rightarrow \mathcal{L}_V$ and $C_E^t : E \rightarrow \mathcal{L}_E$ used to predict the most probable class of a node or an edge according to the consistency scores at time t: $C_E^t(e) = \arg\max_{c \in C_E} S_E^{t-1}(e, c)$ and $C_V^t(v) = \arg\max_{c \in C_V} S_V^{t-1}(v, c)$ simplified as $C^t(u)$ where u can be a node or an edge. In the case where no score is given for the element, the class of the initial graph are used as the current prediction.

4.2. Sequential Step

The goal of a sequential step is to update the scoring function S^t by using Visual Question Answering to extract image information and external knowledge. Initially, no consistency scores are available, meaning that the initial score function S^0 only takes value `NA`. It is iteratively updated by applying a series of 4 modules as shown in Fig. 3:

`select` which chooses the element to verify.

`question` which identifies the plausible class label hypothesis that must be checked in the form of a yes/no question using the knowledge base and the current scores.

`respond` which computes the response to the yes/no question and scores it using a VQA model.

`update` which updates the consistency scores by using the responses obtained by `respond` and the knowledge priors.

All modules except `update` require learning. We present in more details each module in the following.

`select` The goal of this module is to identify the most interesting parts needing verification in the scene graph, given the verification history, in order to assess the truthfulness of their value. The main difficulty that faces this step is the propagation of errors in the graph.

The verified part is selected in two steps. We first choose a target triplet in the graph, and then the element to verify in the triplet (subject, relation or object). We note $q_E^t \in E$ and $q_U^t \in \{\texttt{s}, \texttt{r}, \texttt{o}\}$ the chosen triplet and element to verify, where `s` means that we are targeting the subject $s(q_E^t)$ for verification, `o` the object $o(q_E^t)$ and `r` the relation q_E^t.

We propose 2 ways of computing the selection functions q_E^t and q_U^t. A Rule-Based version selecting elements with a simple ranking rule. For the selection of the target triplet q_E^t, the goal is to look first at edges that have not been verified then verify edges with low score. To select an element q_U^t, we first look at questions that have not been asked previously (primarily edges as it allows to verify the full triplet afterwards), and if the questions are all asked, we take the

question that has received the lowest score (more details in the supplementary).

The second version is a RL-Based version selecting q_E^t and q_U^t from two separate Q-functions:

$$q_E^t = \arg\max_{E \cup \{\texttt{done}\}} Q_E(e|S^{t-1}; \theta_E) \quad (1)$$

$$q_U^t = \arg\max_{u \in \{\texttt{s,o,r}\}} Q_U(u|S^{t-1}, q_E^t; \theta_U) \quad (2)$$

where a `done` output means that no interesting part needs to be verified and that the verification process can be halted. Note that an element can be verified more than one time, allowing to test more class hypotheses.

The two Q-functions are instantiated as Deep Q-Neural Networks (DQN) [33] parameterized by θ_E and θ_U and learned by reinforcement. A more detailed description of the network architecture, stopping condition and rewards, involving graph neural network and multi-layer perceptron is in the supplementary material.

`question` The role of this module is to identify the class labels for which we want to find the consistency score once we have found the graph component q_U^t to be queried. The type of questions that are allowed in our protocol are simple "yes/no" questions like "Is there an umbrella ?" or "Is there a man wearing a hat?".

Not all triplets are relevant: for instance, it would seem strange to ask a question about a person "wearing an elephant". Knowledge priors are rather strong, and can be used to limit the scope of hypotheses that must be verified and to prevent the querying of irrelevant hypotheses that may introduce some noise. We propose in our approach to exploit a simple knowledge base relying on linguistic correlations learned from a scene graph dataset. Formally, the knowledge base is able to receive queries that can be expressed in an SQL like format: `SELECT r,prior WHERE s= "man" AND o= "hat" ORDER BY prior DESC LIMIT 10` which returns the 10 best potential labels for a relation with highest prior values that are consistent with triplets where the subject is `man` and the object is `hat`.

The goal of this database is 1/ to select the potential class label hypotheses that are likely to be true for the component q_U^t according to the context edge q_E^t identified in the previous step and 2/ to provide a priori consistency scores, i.e. not conditioned by the image.

Given the context q_E^t and the verified element q_U^t, we estimate the most consistent class labels for q_U^t according to context q_E^t by looking at the most frequent labels for q_U^t according to current labels C^t of q_E^t triplet. Then we query the knowledge database to retrieve the top-k ranked class label hypotheses and their prior consistency scores for the verified component. The prior consistency scores are estimated in a previous learning phase by computing their frequency on a dattaset. Those scores allow the update of a knowledge-based consistency score function $S_K^t(q_U^t, p)$ by filling the scores for the class labels p that have been retrieved. Those knowledge based values will be used to update the final consistency score in the `update` step.

The textual questions asked to the VQA model can be created in a simple way. If $p \in \mathcal{L}_E$ is a label characterizing a relation (e.g. "`wearing`"), textual questions will be created as "Is there a $C_V^t(s(q_E^t))$ p $C_V^t(o(q_E^t))$?" where the subject and object of the triplet come from current estimates of the class labels (e.g. "`man`" for the subject and "`hat`" for the object resulting in a "`Is there a man wearing a hat?`" question). If $p \in \mathcal{L}_V$, textual questions will be created as "Is there a p?" (e.g. "`Is there a man?`").

`respond` The role of this module is essentially to answer a question about a piece of visual content in the image and to provide a confidence score. It uses the capacity of a VQA algorithm [22] to answer yes/no questions expressed in a free form. The question is fed to the VQA algorithm to compute a confidence score $S_I^t(q_U^t, p)$ about the image content p referred to by the question. It is worth noting that this way of querying the visual content is generic, and can potentially make use of modern foundational models, but is specialized since the asked question is limited to a yes/no type: it can be expected that the response uncertainty will be lower than those resulting from more complex questions

`update` Once the question has been answered, the global consistency score function at time t $S^t(u, c)$ can be updated as a linear combination of the visual content and knowledge based scores: $\alpha S_I^{t'}(u, c) + (1 - \alpha) S_K^{t'}(u, c)$. We also take into account the history of the verification process by exploiting time dependent updating rules. More details are given in the supplementary material.

5. Experiments

5.1. Resources and metrics

Dataset We evaluate and train our models on VG150 [47], a variant of Visual Genome [24] that focuses on the 150 and 50 most frequent object and predicate classes. To create the test set, we remove the VG150 data set from MS COCO [28] as our VQA model is also pretrained on MS COCO. The MS COCO data are moved to the train set. This makes 51498 images for training and 56575 for test. The validation set is obtained by randomly selecting 10000 train images. We test the different methods by randomly corrupting one and three elements on a graph. We also evaluate our approach on generated graphs obtained using [54]. We use ground truth bounding boxes to generate graphs so that they have the same structure as the ground truth graphs.

VQA model: We use a pre-trained Vilt [22, 1] on VQAv2 [12] which does not require region supervision (e.g., object detection). It is fine-tuned on our dataset to improve the prediction of relation (more details in supplementary material).

Graph Type	Algorithm	Top@3 Element Accuracy			Mean Top@3 Element Accuracy		
		Correct	Corrupted	F1-score	Correct	Corrupted	F1-score
1 error	RL-Based	99.43/96.60	32.26/**83.95**	48.71/**89.83**	99.16/83.89	26.25/**46.09**	41.51/**59.49**
		99.20	58.17	***73.33***	95.34	31.21	*47.02*
1 error	Rule-Based	97.32/94.34	**35.92**/81.33	**52.47**/87.35	96.42/77.63	**32.09**/44.70	**48.15**/56.73
		96.05	**58.69**	*72.85*	91.73	**35.24**	***50.91***
1 error	ECE [44]	**100/100**	16.68/48.11	28.59/64.96	**100/100**	8.63/11.39	15.88/20.45
		100	32.59	*49.15*	**100**	9.3	*17.01*
3 errors	RL-Based	98.86/94.38	20.32/63.50	33.71/75.92	98.51/83.41	15.97/33.59	27.48/**47.89**
		96.90	41.64	*58.24*	94.73	20.37	*33.52*
3 errors	Rule-Based	97.64/91.84	**26.79/65.76**	**42.04/76.64**	96.80/75.89	**22.62/34.66**	**36.67**/47.58
		95.10	**46.03**	***62.03***	91.57	**25.63**	***40.05***
3 errors	ECE [44]	**100/100**	13.95/40.93	24.48/58.08	**100/100**	6.63/9.13	12.43/16.73
		100	27.33	*42.92*	**100**	7.25	*13.51*

Table 1. Baselines: Mean and Top@3 Element Accuracy as percentage for nodes/edges and all graph elements on synthesized data with 1 inconsistency (1 error), 3 inconsistencies (3 errors) on VG150 test set. Numbers in bold show the best result for each type of accuracy.

Algorithm	Top@3 Element Accuracy		Mean Top@3 Element Accuracy	
	Accuracy	F1-score	Accuracy	F1-score
None	50.00/43.25		50.00/20.65	
	46.58		*42.66*	
RL-Based	**79.50**/81.22	**27.45**/69.29	**72.31/44.54**	19.25/**53.58**
	80.27	***54.19***	***65.37***	*29.28*
Rule-based	77.52/**81.25**	22.04/**69.64**	71.34/43.85	**20.61**/52.87
	79.20	*52.75*	*64.47*	***29.92***
ECE [44]	76.04/64.26	5.59/12.54	69.60/19.18	3.90/7.76
	64.26	*9.43*	*56.99*	*4.87*

Table 2. Mean and Top@3 Element Accuracy as percentage for nodes/edges and all graph elements on generated graphs [54] (and referred as None in the table) on VG150 test set. Numbers in bold show the best result for each metric.

Metrics Traditional scene graph generation metrics such as Recall@K [11] are not suitable for our problem as they are independent of objects and relations in the graph. To evaluate the predicted labels, we define the Top@K Element Accuracy metric. For each element, we consider that proposals are correct if their ground truth belong to the set of class labels with the K highest score for the element. As most scene graph generation datasets are biased, we also define the Mean Top@K Element Accuracy metric that averages the Top@K Element Accuracy by class. We look separately at elements that are inconsistent before the algorithm (Corrupted) and those that are not (Correct). We also compute F1-score (or F1) as one example of trade-off between Corrupted and Correct accuracies. To evaluate the correction of generated graphs we use the Top@K Element Accuracy (Accuracy), as it allows global evaluation of scene graphs.

Baselines The verification of scene graphs is a new problem. As a baseline, we compare our method to a state-of-the-art caption editing method, ECE [44], for which we transformed the original scene graph in a caption to be plugged as an input to the algorithm that produces the suggested editions. This baseline and its implementation is fully described in the supplementary material.

5.2. Verification by consistency score computation

The experimental results for the main verification task relying on the computation of consistency scores are presented in Tab. 1. As a general comment, one can say that the verification of non localized scene graphs is a difficult task: we are far from reaching 100% performance. The evaluated algorithms show different behaviors, though.

The two sequential algorithms (Rule-Based or RL-Based), show much better detection and correction of corruptions than ECE (58.17% and 58.69% vs. 32.59%), which seems to be more conservative and keeps a higher level of identification of correct classes.

The Rule-Based approach provides overall comparable performance (F1 score) to the RL-Based question selection strategy for corrupted graph with 1 error, and better performance with 3 errors, at the expense of adding noise to the prediction when the initial class is correctly assigned.

We also observe for all models a poor rate of corrections

Graph Type	Algorithm	Top@3 Element Accuracy			Mean Top@3 Element Accuracy		
		Correct	Corrupted	F1-score	Correct	Corrupted	F1-score
1 error	RL-Based	99.43/96.60	**32.26/83.95**	**48.71/89.83**	99.16/83.89	**26.25**/46.09	**41.51/59.49**
		99.20	**58.17**	***73.33***	95.34	**31.21**	***47.02***
1 error	RL-Based Rand-Props	99.33/**99.66**	2.45/9.51	4.78/17.36	99.24/**99.56**	2.30/8.16	4.49/15.08
		98.35	9.09	*16.64*	**99.32**	3.77	*7.26*
1 error	RL-Based All-Props	**99.50**/95	5.00/83.86	9.52/89.08	**99.47**/81.45	0.60/**49.44**	1.19/61.53
		97.49	42.27	*58.97*	94.96	12.81	*22.57*

Table 3. Ablation Studies: Mean and Top@3 Element Accuracy as percentage for nodes/edges and all graph elements on synthesized data with 1 inconsistency (1 error) on VG150 test set. Numbers in bold show the best result for each metric.

on nodes compared to edges. For the sequential models, this can be explained by the fact that the knowledge base imposes strong priors that prevent the discovery of the true class label for certain nodes. Another explanation is that scene graphs are incomplete [5]: several entities may be observable in the image but not encoded in the graph, potentially causing the VQA to assign a wrong class to the verified component. This behavior is confirmed by the poor results of the All-Props strategy on nodes (Tab. 3), which checks all possible classes.

The number of inconsistencies is also influential. Even if our algorithms and ECE improve results with 3 inconsistencies, we observe a large gap compared to 1 inconsistency correction. This is mostly due to triplets containing 2 or more errors. Finding the correct edge label in this case is difficult because it requires finding the correct class hypothesis to check without the help of neighboring nodes.

5.3. Correction of scene graph generation

One possible application of the verification algorithm is to make a correction of a given scene graph generated by another source. Tab. 2 shows the improvement of the correction starting from a graph generated directly from the image and using [54].

We see that trying to verify a graph by comparing it to the output of a scene graph generator may not be a good idea: the initial generated graph has a very low quality (None line) and can not therefore be considered as a good reference. We have computed the average number of inconsistencies to be 6.24.

The three tested algorithms show a clear capacity of improving the graph. This is especially true for our Rule-Based and RL-Based algorithms, which improve by more than 33% the Top@3 accuracy, i.e. the percentage of correct labels for all elements, nodes and edges.

5.4. Ablation Study

Tab. 3 compares the performance on two ablations. Their role is to replace the selection of the `question` component by a random selection of labels with the same number of propositions at each step (Rand-Props) and an exhaustive verification of all possible labels of an element at each step (All-Props).

We observe that modifying `question` is detrimental to corruption detection: when selecting random classes, the model is not able to find the correct class most of the time. When selecting all classes, we observe better results on edges but very bad results on nodes: it struggles to figure out where in the graph a node class is supposed to go. We also observe a loss of correct graph elements due to VQA inaccuracy.

5.5. Qualitative Analysis

Fig. 4 presents several output examples. a), c) and b) are examples of correct corrections for scene graphs with 1 inconsistency, with 3 inconsistencies and from scene graph generation respectively. For generated graphs, a lot of corrected inconsistencies are synonyms similar to c). Our model is able to find those synonyms compared to classic approaches that use a large classification layer to predict object/relation hypotheses and lacks flexibility since it forces the prediction to output a unique hypothesis. We also analyze errors of our method: a difficulty are triplets completely incorrect such as "skier growing on railing" in d). In our method, `question` module is not able to find the correct class with the erroneous context. However, as we see in the example, the score given to such triplet is very low, allowing to detect them. e) is another frequent error: the triplet "sign on pole" which is incorrect at the start is replaced by "letter on pole" that can be considered correct but not present in the graph. f) is an error caused by the VQA model (`respond` module), it detects a short in the image.

6. Limitations

Impact of knowledge base Our method shows a large bias toward label edges compared to ECE following biases observed on Visual Genome [54, 43]. This bias is strengthened by the use of our training set as a knowledge base. Finding another source of external knowledge could reduce the bias of the method but may also add other biases.

Future work We have discussed the problem of the incompleteness of scene graphs. However, finding the relevant

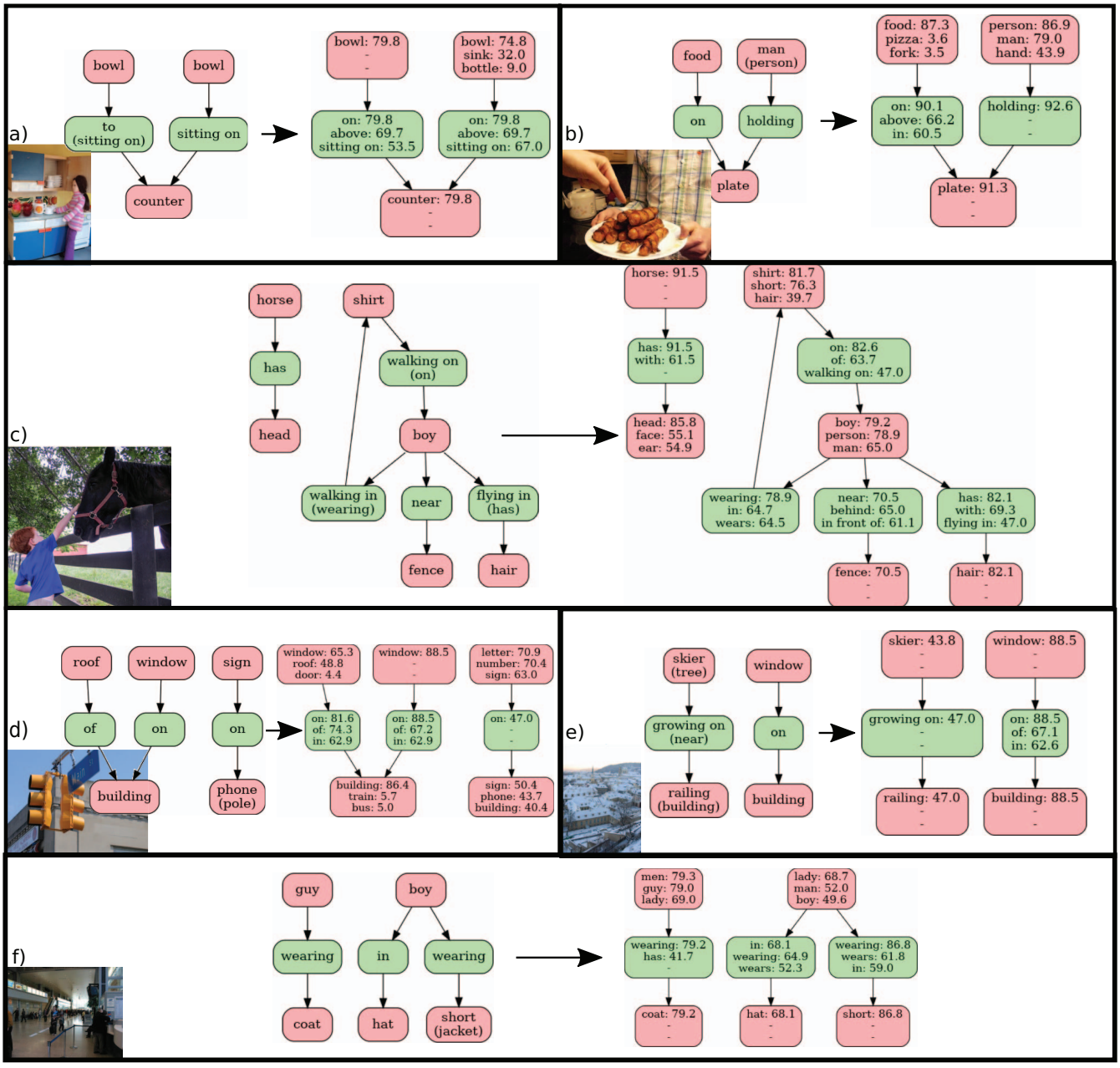

Figure 4. Examples of outputs of a sequential step of our method (one by rectangle). Scene graphs before arrows correspond to the graphs given to our model with the ground truth labels in parenthesis for corrupted elements. Scene graphs after arrows give for each graph element the top 3 highest scores classes. The corresponding images are in the left corner bottom of each rectangle.

level of description of an image depends on its use. In our approach, we have used semantic priors as a way to focus on the good label hypotheses to be verified given a graph structure, not as a way to modify the structure itself, typically by adding new nodes. An interesting question is, therefore, how to control the completeness of the description – an issue related to increasing the relevance of the description – for example, by exploiting the generative capabilities of semantic priors [53]. We leave this question for future work.

7. Conclusion

We have introduced a new task: image description verification from non-localized scene graphs, and proposed a new method that combines VQA and image description priors in a sequential decision process. Our experiments show that it is possible to achieve convincing results with our method on the VG150 dataset. We also show on a few examples that our decision process is transparent, making it possible to identify the potential causes of verification errors.

References

[1] Vision-and-language transformer (vilt), fine-tuned on vqav2. https://huggingface.co/dandelin/vilt-b32-finetuned-vqa. 5

[2] VQA challenge 2020. https://visualqa.org/roe.html. Accessed: 2023-08-15. 2

[3] 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. 2

[4] David Abou Chacra and John Zelek. The Topology and Language of Relationships in the Visual Genome Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4860–4868, 2022. 2

[5] David Abou Chacra and John Zelek. The topology and language of relationships in the visual genome dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 4860–4868, June 2022. 7

[6] Shizhe Chen, Qin Jin, Peng Wang, and Qi Wu. Say As You Wish: Fine-grained Control of Image Caption Generation with Abstract Scene Graphs. *arXiv:2003.00387 [cs]*, Feb. 2020. arXiv: 2003.00387. 3

[7] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-Embedded Routing Network for Scene Graph Generation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6156–6164, June 2019. ISSN: 2575-7075. 3

[8] Vincent S. Chen, Paroma Varma, Ranjay Krishna, Michael Bernstein, Christopher Re, and Li Fei-Fei. Scene Graph Prediction With Limited Labels. pages 2580–2590, 2019. 3

[9] Bo Dai, Yuqi Zhang, and Dahua Lin. Detecting Visual Relationships with Deep Relational Networks. Technical Report arXiv:1704.03114, arXiv, Apr. 2017. arXiv:1704.03114 [cs] type: article. 3

[10] Sarthak Garg, Helisa Dhamo, Azade Farshad, Sabrina Musatian, Nassir Navab, and Federico Tombari. Unconditional Scene Graph Generation. *arXiv:2108.05884 [cs]*, Aug. 2021. 3

[11] Nikolaos Gkanatsios, Vassilis Pitsikalis, Petros Koutras, and Petros Maragos. Attention-Translation-Relation Network for Scalable Scene Graph Generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 6

[12] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering, May 2017. arXiv:1612.00837 [cs]. 5

[13] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene Graph Generation With External Knowledge and Image Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1969–1978, 2019. 3

[14] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene Graph Generation With External Knowledge and Image Reconstruction. pages 1969–1978, 2019. 3

[15] Marcel Hildebrandt, Hang Li, Rajat Koner, Volker Tresp, and Stephan Günnemann. Scene Graph Reasoning for Visual Question Answering, July 2020. arXiv:2007.01072 [cs, stat]. 3

[16] Ronghang Hu, Jacob Andreas, Trevor Darrell, and Kate Saenko. Explainable Neural Computation via Stack Neural Module Networks, Mar. 2019. arXiv:1807.08556 [cs]. 3

[17] Qingbao Huang, Yu Liang, Jielong Wei, Yi Cai, Hanyu Liang, Ho-fung Leung, and Qing Li. Image Difference Captioning With Instance-Level Fine-Grained Feature Representation. *IEEE Transactions on Multimedia*, 24:2004–2017, 2022. 3

[18] Yufeng Huang, Jiji Tang, Zhuo Chen, Rongsheng Zhang, Xinfeng Zhang, Weijie Chen, Zeng Zhao, Tangjie Lv, Zhipeng Hu, and Wen Zhang. Structure-CLIP: Enhance Multi-modal Language Representations with Structure Knowledge, May 2023. arXiv:2305.06152 [cs]. 3

[19] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image Generation from Scene Graphs. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1219–1228, June 2018. 1

[20] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image Generation from Scene Graphs, Apr. 2018. arXiv:1804.01622 [cs]. 3

[21] Xuan Kan, Hejie Cui, and Carl Yang. Zero-Shot Scene Graph Relation Prediction through Commonsense Knowledge Integration. Technical Report arXiv:2107.05080, arXiv, July 2021. arXiv:2107.05080 [cs] type: article. 3

[22] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision, 2021. 5

[23] 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 Li Fei-Fei. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision*, 123(1):32–73, May 2017. 2

[24] 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. *arXiv:1602.07332 [cs]*, Feb. 2016. arXiv: 1602.07332. 5

[25] Chung-Wei Lee, Wei Fang, Chih-Kuan Yeh, and Yu-Chiang Frank Wang. Multi-label zero-shot learning with structured knowledge graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1576–1585, 2018. 3

[26] Xiangyang Li and Shuqiang Jiang. Know More Say Less: Image Captioning Based on Scene Graphs. *IEEE Transactions on Multimedia*, 21(8):2117–2130, Aug. 2019. 2, 3

[27] Xiaodan Liang, Lisa Lee, and Eric P. Xing. Deep Variation-Structured Reinforcement Learning for Visual Relationship and Attribute Detection. In *Proceedings of the IEEE Con-*

ference on Computer Vision and Pattern Recognition, pages 848–857, 2017. 3

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, Lecture Notes in Computer Science, pages 740–755, Cham, 2014. Springer International Publishing. 5

[29] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019. 3

[30] Yichao Lu, Himanshu Rai, Jason Chang, Boris Knyazev, Guangwei Yu, Shashank Shekhar, Graham W. Taylor, and Maksims Volkovs. Context-aware Scene Graph Generation with Seq2Seq Transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15911–15921, Montreal, QC, Canada, Oct. 2021. IEEE. 3

[31] Yichao Lu, Himanshu Rai, Jason Chang, Boris Knyazev, Guangwei Yu, Shashank Shekhar, Graham W. Taylor, and Maksims Volkovs. Context-Aware Scene Graph Generation With Seq2Seq Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15931–15941, 2021. 3

[32] Pranava Madhyastha, Josiah Wang, and Lucia Specia. Defoiling Foiled Image Captions. Technical Report arXiv:1805.06549, arXiv, May 2018. arXiv:1805.06549 [cs] type: article. 2

[33] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with Deep Reinforcement Learning. *arXiv:1312.5602 [cs]*, Dec. 2013. arXiv: 1312.5602. 5

[34] Dong Huk Park, Trevor Darrell, and Anna Rohrbach. Robust Change Captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4624–4633, 2019. 3

[35] John Pavlopoulos, Vasiliki Kougia, Ion Androutsopoulos, and Dimitris Papamichail. Diagnostic captioning: A survey. *Knowledge and Information Systems*, 64(7):1691–1722, July 2022. 1

[36] Yue Qiu, Shintaro Yamamoto, Kodai Nakashima, Ryota Suzuki, Kenji Iwata, Hirokatsu Kataoka, and Yutaka Satoh. Describing and Localizing Multiple Changes With Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1971–1980, 2021. 3

[37] Fawaz Sammani and Luke Melas-Kyriazi. Show, Edit and Tell: A Framework for Editing Image Captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4808–4816, 2020. 3

[38] Sebastian Schuster, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D. Manning. Generating Semantically Precise Scene Graphs from Textual Descriptions for Improved Image Retrieval. In *Proceedings of the Fourth Workshop on Vision and Language*, pages 70–80, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. 2

[39] Sahand Sharifzadeh, Sina Moayed Baharlou, and Volker Tresp. Classification by Attention: Scene Graph Classification with Prior Knowledge. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(6):5025–5033, May 2021. 3

[40] Ravi Shekhar, Sandro Pezzelle, Yauhen Klimovich, Aurélie Herbelot, Moin Nabi, Enver Sangineto, and Raffaella Bernardi. FOIL it! Find One mismatch between Image and Language caption. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 255–265, Vancouver, Canada, July 2017. Association for Computational Linguistics. 2

[41] Jiaxin Shi, Hanwang Zhang, and Juanzi Li. Explainable and Explicit Visual Reasoning Over Scene Graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8376–8384, 2019. 3

[42] Xiangxi Shi, Xu Yang, Jiuxiang Gu, Shafiq Joty, and Jianfei Cai. Finding It at Another Side: A Viewpoint-Adapted Matching Encoder for Change Captioning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, Lecture Notes in Computer Science, pages 574–590, Cham, 2020. Springer International Publishing. 3

[43] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased Scene Graph Generation From Biased Training. pages 3716–3725, 2020. 7

[44] Zhen Wang, Long Chen, Wenbo Ma, Guangxing Han, Yulei Niu, Jian Shao, and Jun Xiao. Explicit image caption editing. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI*, pages 113–129. Springer, 2022. 2, 6

[45] Zhen Wang, Long Chen, Wenbo Ma, Guangxing Han, Yulei Niu, Jian Shao, and Jun Xiao. Explicit Image Caption Editing. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, Lecture Notes in Computer Science, pages 113–129, Cham, 2022. Springer Nature Switzerland. 3

[46] Yang Wu, Pengxu Wei, and Liang Lin. Scene Graph to Image Synthesis via Knowledge Consensus. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(3):2856–2865, June 2023. Number: 3. 3

[47] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. Scene Graph Generation by Iterative Message Passing. pages 5410–5419, 2017. 5

[48] Chiao-An Yang, Cheng-Yo Tan, Wan-Cyuan Fan, Cheng-Fu Yang, Meng-Lin Wu, and Yu-Chiang Frank Wang. Scene Graph Expansion for Semantics-Guided Image Outpainting, May 2022. arXiv:2205.02958 [cs]. 3

[49] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Visual Curiosity: Learning to Ask Questions to Learn Visual Recognition. In *Proceedings of The 2nd Conference on Robot Learning*, pages 63–80. PMLR, Oct. 2018. ISSN: 2640-3498. 3

[50] Xu Yang, Chongyang Gao, Hanwang Zhang, and Jianfei Cai. Hierarchical Scene Graph Encoder-Decoder for Image Paragraph Captioning. In *Proceedings of the 28th ACM Inter-*

national Conference on Multimedia, MM '20, pages 4181–4189, New York, NY, USA, Oct. 2020. Association for Computing Machinery. 3

[51] Linli Yao, Weiying Wang, and Qin Jin. Image Difference Captioning with Pre-training and Contrastive Learning, Feb. 2022. 3

[52] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Bridging Knowledge Graphs to Generate Scene Graphs. *arXiv:2001.02314 [cs]*, July 2020. arXiv: 2001.02314. 3

[53] Alireza Zareian, Zhecan Wang, Haoxuan You, and Shih-Fu Chang. Learning Visual Commonsense for Robust Scene Graph Generation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, Lecture Notes in Computer Science, pages 642–657, Cham, 2020. Springer International Publishing. 3, 8

[54] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural Motifs: Scene Graph Parsing with Global Context. (arXiv:1711.06640), Mar. 2018. arXiv:1711.06640 [cs] type: article. 2, 3, 5, 6, 7

[55] Yifeng Zhang, Ming Jiang, and Qi Zhao. Explicit Knowledge Incorporation for Visual Reasoning. pages 1356–1365, 2021. 3

[56] Yiwu Zhong, Liwei Wang, Jianshu Chen, Dong Yu, and Yin Li. Comprehensive Image Captioning via Scene Graph Decomposition. *arXiv:2007.11731 [cs]*, July 2020. arXiv: 2007.11731. 3