Linguistic Structures as Weak Supervision for Visual Scene Graph Generation

Keren Ye Adriana Kovashka
Department of Computer Science, University of Pittsburgh
{yekeren, kovashka}@cs.pitt.edu
https://github.com/yekeren/WSSGG

Abstract

Prior work in scene graph generation requires categorical supervision at the level of triplets—subjects and objects, and predicates that relate them, either with or without bounding box information. However, scene graph generation is a holistic task: thus holistic, contextual supervision should intuitively improve performance. In this work, we explore how linguistic structures in captions can benefit scene graph generation. Our method captures the information provided in captions about relations between individual triplets, and context for subjects and objects (e.g. visual properties are mentioned). Captions are a weaker type of supervision than triplets since the alignment between the exhaustive list of human-annotated subjects and objects in triplets, and the nouns in captions, is weak. However, given the large and diverse sources of multimodal data on the web (e.g. blog posts with images and captions), linguistic supervision is more scalable than crowdsourced triplets. We show extensive experimental comparisons against prior methods which leverage instance- and image-level supervision, and ablate our method to show the impact of leveraging phrasal and sequential context, and techniques to improve localization of subjects and objects.

1. Introduction

Recognizing visual entities and understanding the relations among them are two fundamental problems in computer vision. The former task is known as object detection (OD) and the latter as visual relation detection (VRD). In turn, scene graph generation (SGGen) requires to jointly detect the entities and classify their relations.

While scene graphs are a holistic, contextual representation of an image, the types of supervision that have been used capture context in an impoverished way. In particular, prior methods use supervision in the form of either subject-predicate-object triplets with bounding boxes for the subject and object [27, 34, 49] or subject-predicate-object triplets at the image level only [54, 57]. Thus, information in the supervision is local (separate triplets) while the scene graph to be output captures the entire image. This discrepancy between the properties of the desired output (global) and training data (local) becomes problematic due to potential ambiguity in the visual input. For example, in Fig. 1, multiple *persons* are standing on the *rails*. Thus, standard supervision (top) which breaks down a scene graph into triplets, may create confusion.

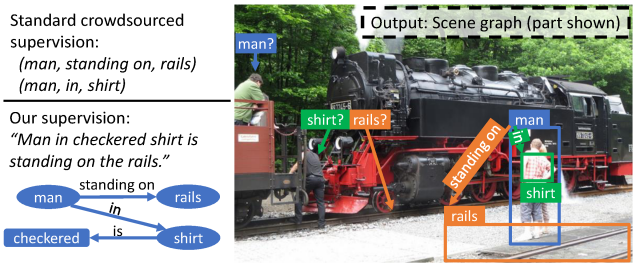

Figure 1. We tackle the problem of generating scene graphs with supervision in the form of captions at training time. Parsing from captions enables utilization of the huge amount of image-text data available on the internet. The linguistic structure extracted maintains the relational information described in the caption without the loss of cross-triplet references, and facilitates disambiguation.

In contrast, *captions* capture global context that allows us to link multiple triplets, and localize a man who is both standing on the rails, and wearing a (checkered) shirt. Captions are linguistic constructs, and language could be argued to capture common sense (e.g., BERT [11] models are good at question-answering and commonsense tasks). Captions are also advantageous in terms of cost: humans naturally provide language descriptions of visual content they upload, thus caption-like supervision can be seen as "free". However, caption supervision contains noise, which presents some challenges. First, captions provide supervision at the image level, similar to prior work in weakly-supervised scene graph generation [54]. Second, prior work [32, 51] shows that captions do not cover all relevant objects: not all content is mentioned, and some of the mentioned content is not referring to the image explicitly or is not easily localizable. Because captions are noisy, the supervision we use is even weaker than prior work [54].

We propose an approach that leverages global context, using captions as supervision. Our approach models context for scene graphs in two ways. First, it extracts information from captions beyond the subject-predicate-object entities (e.g., in the form of attributes like "checkered", in Fig. 1). This context enables more accurate representations of concepts, and thus more accurate localization of each subject-predicate-object triplet. Second, visuo-linguistic context provides a way to reason about common-sense relationships within each triplet, to prevent non-sensical triplets from being generated (e.g., "rails standing on man" is unlikely, while "man standing on rails" is likely). To cope with the challenges of the noise contained in captions, we rely on an iterative detection method which helps prune some spurious relations between caption words and image regions, via boostrapping. While the captions we use are crowdsourced, our method paves the road for using image-caption pairs harvested from the internet for free, using text accompanying images on the web, from blogs, social media posts, YouTube video descriptions, and instructional videos [31, 42, 53]. Note that our method internally uses a graph with broad types of nodes, including adjectives, even though these are not part of the graph that is being output at test time. A side contribution is an adaptation of techniques from weakly-supervised object detection to improve localization of subject and object through iterative refinement, which has not been used for scene graph generation before.

To isolate the contribution of global context from the noise contained in captions (i.e., objects not being mentioned), we verify our approach in two settings. First, we construct a ground-truth triplet graph by connecting triplets with certain overlap. We show that our full method greatly outperforms prior work (it boosts the performance of [54] by 59%-67%). Second, we use two types of actual captions. This causes overall performance to drop, but we observe that modeling phrasal (cross-triplet) and sequential (within-triplet) linguistic context achieves strong results, significantly better than more direct uses of captions, and competitive with methods using clean image-level supervision.

To summarize, our contributions are as follows:

- We examine a new mechanism for scene graph generation using a new type of weak supervision.
- We contextualize embeddings for subject/object entities based on linguistic structures (e.g. noun phrases).
- We propose new joint classification and localization of subject, object and predicate within a triplet.
- We leverage weakly-supervised object detection techniques to improve scene graph generation.

2. Related Work

Learning from textual descriptions: Open information extraction systems [3, 9, 12, 29, 50] produce relation triples using surface and dependency patterns, but target language-only relation extraction or question answering. On the vision end, method exist to parse a question or image into a structured, tree-like form, for composable visual reasoning [2, 13, 17, 20, 28, 52]. Following the emergence of scene graphs [18] as a global description of an image, automatic parsing from textual descriptions to scene graphs [41, 46] aims to fill the gap between texts and images. It tackles practical issues such as pronoun resolution and plural nouns, and duplicates some nodes in the scene graph if necessary. Though we use the parser designed in [41], our reliance on parsing is different. While the above methods tackle pure language tasks, visual question answering, and image retrieval, we use the parsed results as supervised signals to guide a scene graph generation model during training. Our work is similar to [7, 16, 51] since we extract or amplify information from captions. However, these works only extract *entities* from captions, while we also learn from the properties and relations described. Also related are recent methods that use supervision from visual-language pairs [10, 30, 33, 43, 47], but these learn general-purpose representations and do not perform scene graph generation.

Visual grounding of phrases locates the entities in an image, based on a given natural language query. [19] align sentence fragments with image regions. [6, 40] attend to the relevant image regions to reconstruct the input phrase, similar to weakly-supervised object detection. [58] incorporate a spatial transformer [15] to refine object boxes relative to multi-scale anchors. We use a technique similar to visual grounding to find label-related regions, but our key innovation lies in our use of the linguistic structure. We allow context to propagate to language queries to improve entity detection. Our model only takes image inputs at test time.

Scene graph generation (SGGen) aims to localize and recognize all visual entities and predict predicates between them. Most approaches [8, 14, 24, 26, 27, 34, 38, 48, 49, 55] learn to generate graphs in a fully-supervised manner, in which training data involves both entities (bounding boxes and labels) and predicates. Inspired by weakly-supervised object detection (WSOD) [5, 35], [37, 54, 57] somewhat reduce the reliance on these *labor-intensive* annotations. [37] infer visual relations using only image-level triplets. [57] directly apply WSOD for entity localization and add a weakly-supervised visual relation detection (WSVRD) task for classifying entity pairs. [54] match predicates to entities and jointly infer the entities, predicates, and their alignments, using a bipartite graph. However, [37, 54, 57] still require clean triplet annotations from crowdsourcing, while our method only requires captions. Further, we capture visual properties in the internal graph our method uses at training time; these cannot be represented using triplets but help to enrich the visual representation and better ground entities. [54]'s method includes a more general (subject, predicate, $\emptyset$) graph, but it does not capture visual attributes.

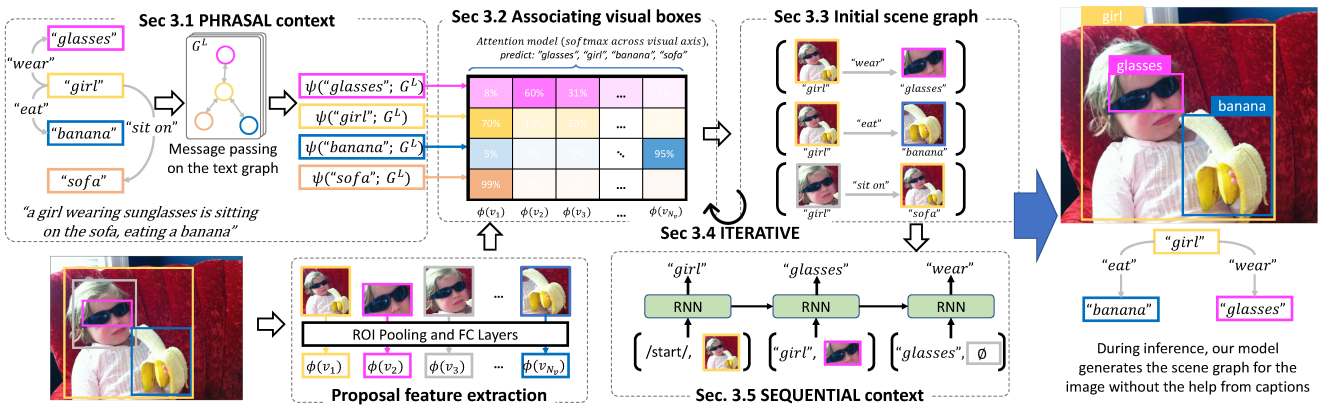

Figure 2. Model overview. Our model uses the image's paired caption as weak supervision to learn the entities in the image and the relations among them. At inference time, it generates scene graphs without help from texts. To learn our model, we first allow context information to propagate on the text graph to enrich the entity word embeddings (Sec. 3.1). We found this enrichment provides better localization of the visual objects. Then, we optimize a text-query-guided attention model (Sec. 3.2) to provide the image-level entity prediction and associate the text entities with visual regions best describing them. We use the joint probability (Eq. 6) to choose boxes associated with both subject and object (Sec. 3.3), then use the top scoring boxes (Eq. 7) to learn better grounding (Sec. 3.4). Finally, we use an RNN (Sec. 3.5) to capture the vision-language common-sense and refine our predictions. Our code is available at https://github.com/yekeren/WSSGG.

3. Approach

Inputs. Our method does not rely on dense human-annotated instances and relations, but takes in linguistic structures as supervised signals (Fig. 2 top-left). Such structural text information is abandoned in other weakly-supervised methods [54, 56, 57]. We first convert captions paired with images into text graphs using a language parser [41]. The resulting graphs describe the entities in the caption and the relations (e.g., verbs or prepositions) among them. We call this setting Cap-Graph. Our method's performance depends on how exhaustive the caption is, and how robust is the parser chosen. Thus, we also design a setting where we extract a ground-truth text graph from the scene graph annotations, ignoring bounding boxes (GT-Graph).

Training pipeline overview (Fig. 2): We extract the visual object proposals using FasterRCNN [39]. We extract the text graph from paired captions (Cap-Graph) or directly read the ground-truth text graph (GT-Graph). We use a graph neural network based on the phrasal structure to enrich the text node representation (Fig. 2 top-left, Sec. 3.1). This enrichment simplifies the later localization step because we can search for more specifically described regions (e.g., "girl eating banana," rather than "girl"). By optimizing the image-level entity scores and treating the text entities as queries, we obtain attention scores, which strongly imply the visual regions that best describe the text entities (Fig. 2 top-middle, Sec. 3.2). We design a way to learn from the weak signal of the attention scores and predict initial relation detection results in the form of 5-tuples (Sec. 3.3). These groundings are further refined using WSOD techniques [45] (Sec. 3.4). Finally, we capture visuo-linguistic common sense to further rule out unlikely relation tuples (Fig. 2 middle-bottom, Sec. 3.5). We use an RNN to model the fluency of scene graph tuples, enforcing that subject/object regions should be followed by their labels, and subject/object should be followed by object/predicate. This module reassigns labels and reranks 5-tuples to improve the relation detection: if an uncommon tuple is fed to the model, it will be assigned a low score.

3.1. Modeling PHRASAL context

We first determine how to represent the text entities to be matched in the image. A naive solution would be to use the word embeddings, but this method ignores the context captured in phrases. We advocate the use of the hints in the phrasal structure, namely mentions of related adjectives and objects. As shown in Fig. 2 top-left, "wearing sunglasses," "sitting on the sofa" and "eating a banana" provide context for the same "girl" and make her distinguishable from other potential instances of "girl". We infer the contextualized entity word features via the phrasal context and apply them in Sec. 3.2 to localize visual objects.

We have summarized all notations in Tab. 1 to facilitate reading the following text. The linguistic structure (Fig. 2 top-left) parsed from a caption is represented using a text graph $G^L = (E, R)$. $E = [e_1 \cdots e_{n_e}]^T$ denotes the n_e text graph entities where each $e_i \in \{1 \dots c_e\}$ represents an entity class ID (c_e classes in total, which are defined by [54] or [48] in our experiments; in Fig. 2 top-left, E = ["glasses", "girl", "banana", "sofa"]^T). $R = [(r_1, s_1, o_1) \cdots (r_{n_r}, s_{n_r}, o_{n_r})]^T$ describes the n_r relations. For the i-th relation: $r_i \in \{1 \dots c_r\}$ is the relation class ID; $s_i, o_i \in \{1 \dots n_e\}$ are entity indices: e_{s_i} denotes the subject entity and e_{o_i} the object entity; in Fig. 2 top-left,

Visual features		
V_{prop}	Region proposals	$n_v \times 1$
V_{feat}	Region proposal features	$n_v \times d_{cnn}$
$n_v = 20$	Number of region proposals	
$d_{cnn} = 1536$	Feature dimension	
Text graph $G^L(E, R)$, parsed from caption		
$E = [e_i]_{i=1}^{n_e}$	Entities (graph nodes)	$\|E\| = n_e$
$R = [(r_i, s_i, o_i)]_{i=1}^{n_r}$	Relations (graph edges)	$\|R\| = n_r$
n_e, n_r	Number of entities/relations in a graph	
c_e, c_r	Number of entity/relation classes (vocab size)	
e_i	The i-th entity node, $e_i \in \{1 \cdots c_e\}$	
r_i	The i-th relation edge, $r_i \in \{1 \cdots c_r\}$	
s_i, o_i	Subject/object index of i-th relation, $s_i, o_i \in \{1 \cdots n_e\}$, e_{s_i}, e_{o_i} refer to subject/object	
Frozen GloVe embeddings		
W_{ent}	Entity embedding matrix	$c_e \times d$
W_{rel}	Relation embedding matrix	$c_r \times d$

Image-level labels parsed from G^L		
Y_{ent}	$Y_{ent}[i,:]$ is the one-hot representation of e_i	$n_e \times c_e$
Y_{rel}	$Y_{rel}[i,:]$ is the one-hot representation of r_i	$n_r \times c_r$
Y_{cssub}, Y_{csobj}	$Y_{cssub}[i,:], Y_{csobj}[i,:]$ are one-hot repr of e_{s_i}, e_{o_i}	$n_r \times c_e$
Y_{cspred}	Alias of Y_{rel}	$n_r \times c_r$
Instance-level pseudo labels		
n_t	Number of iterations to improve $\boldsymbol{g}$	
$\boldsymbol{g}^{(t)}, t \in \{0 \cdots n_t\}$	Grounding vector, if E=[*girl*, *banana*], $\boldsymbol{g}$=[10, 17] means proposal v_{10} is class *girl* and v_{17} is *banana*	$n_e \times 1$
$Y_{det}^{(t)}, t \in \{0 \cdots n_t\}$	Entity detection label, $Y_{det}[i,j]$=1 means the proposal v_i involves the j-th entity class	$n_v \times c_e$
Y_{relsub}, Y_{relobj}	Relation detection label, $Y_{relsub}[i,j]$=1 means the proposal v_i may serve as a subject, and can apply the j-th relation to an unknown object; $Y_{relobj}[i,j]$=1 means the proposal v_i may serve as an object, some unknown subject can apply the j-th relation to v_i	$n_v \times c_r$

Table 1. Overview of notation for the visual features, linguistic structure G^L and supervision parsed from G^L.

$R = \{$("wear", 2, 1), ("eat", 2, 3), ("sit", 2, 4))$\}$. Given the GloVe embedding [36] of the entity and relation classes $W_{ent} \in \mathbb{R}^{c_e \times d}$, $W_{rel} \in \mathbb{R}^{c_r \times d}$, and the one-hot representation of entities and relations $Y_{ent} \in \mathbb{R}^{n_e \times c_e}$, $Y_{rel} \in \mathbb{R}^{n_r \times c_r}$ (each row is a c_e or c_r-dim one-hot vector, and there are n_e and n_r rows, respectively), the initial entity and relation word embeddings can be represented as $H_{ent}^{(0)} = Y_{ent}W_{ent} \in \mathbb{R}^{n_e \times d}$ and $H_{rel}^{(0)} = Y_{rel}W_{rel} \in \mathbb{R}^{n_r \times d}$.

Now we compute phrasal contextualized entity embeddings $\psi(E; G^L) \in \mathbb{R}^{n_e \times d}$. Alg. 1 shows the process, and can be stacked several times. We update relation edge embeddings, then aggregate the relation features into the connected entity nodes, using linear layers ϕ^r and ϕ^α applied on the concatenation of inputs. We use $\psi(E; G^L) = H_{ent}^{(t)}, (t > 1)$ in the next section, to localize visual entities.

Algorithm 1: Message passing to utilize phrasal context. We use TF-GraphNets [4] to implement.

Input : Text graph $G^L = (E, R)$
Initial entity features $H_{ent}^{(t)} = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_{n_e}]^T$
Initial relation features $H_{rel}^{(t)} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{n_r}]^T$
Output: Updated $H_{ent}^{(t+1)}$, $H_{rel}^{(t+1)}$
for $i \leftarrow 1$ *to* n_r **do**
  $\boldsymbol{r}'_i \leftarrow \phi^r(\boldsymbol{r}_i, \boldsymbol{e}_{s_i}, \boldsymbol{e}_{o_i})$ // Update edge, $\boldsymbol{r}'_i \in \mathbb{R}^{d \times 1}$
  $\alpha_i \leftarrow \phi^\alpha(\boldsymbol{r}_i, \boldsymbol{e}_{s_i}, \boldsymbol{e}_{o_i})$ // Update edge weight, $\alpha_i \in \mathbb{R}^1$
for $i \leftarrow 1$ *to* n_e **do**
  $\boldsymbol{e}'_i \leftarrow \sum_{\substack{j=1:n_r, \\ o_j=i}} \{\frac{\exp(\alpha_j)}{\sum_{\substack{k=1:n_r, \\ o_k=i}} \exp(\alpha_k)}\} \boldsymbol{r}'_j$ //Aggregate, $\boldsymbol{e}'_i \in \mathbb{R}^{d \times 1}$
return $H_{ent}^{(t+1)} = [\boldsymbol{e}'_1 \cdots \boldsymbol{e}'_{n_e}]^T, H_{rel}^{(t+1)} = [\boldsymbol{r}'_1 \cdots \boldsymbol{r}'_{n_r}]^T$

3.2. Associating text entities with visual boxes

After getting the contextualized entity embeddings $\psi(E; G^L) \in \mathbb{R}^{n_e \times d}$, we seek their associated visual regions $\boldsymbol{g}^{(0)} \in \mathbb{R}^{n_e \times 1}$ (i.e., grounding vector), where each $\boldsymbol{g}_i^{(0)}$ ranges in $\{1 \cdots n_v\}$ and $v_{\boldsymbol{g}_i^{(0)}}$ denotes the visual box best describing the text entity e_i. We obtain $\boldsymbol{g}$ using an attention mechanism. By optimizing the image-level prediction, we expect the model to learn to focus on the most informative and distinguishable regions, which can often be used as instance references for training object detectors.

We first project $V_{feat} \in \mathbb{R}^{n_v \times d_{cnn}}$ to the d-dim visual-language space, resulting in attention and classification heads $H_{att}, H_{cls} \in \mathbb{R}^{n_v \times d}$. Then, we compute $D_{dot} \in \mathbb{R}^{n_e \times n_v}$, in which $D_{dot}[i,j]$ measures the compatibility between text entity e_i and visual region v_j. We softmax-normalize D_{dot} to get the attention matrix $A^{(0)} \in \mathbb{R}^{n_e \times n_v}$, and obtain $\boldsymbol{g}^{(0)}$ by selecting the max-valued entry.

$$H_{att} = V_{feat}W_{att},\ H_{cls} = V_{feat}W_{cls}$$
$$D_{dot} = \psi(E; G^L)H_{att}^T,\ A^{(0)}[i,j] = \frac{\exp(D_{dot}[i,j])}{\sum_{k=1}^{n_v} \exp(D_{dot}[i,k])}$$
$$\boldsymbol{g}_i^{(0)} = \underset{j \in \{1 \cdots n_v\}}{\arg\max}\ A^{(0)}[i,j] \quad (1)$$

We use image-level entity labels $Y_{ent} \in \mathbb{R}^{n_e \times c_e}$ as supervision to learn proper attention scores. We first aggregate the image-level weighted visual features $F = [\boldsymbol{f}_1 \cdots \boldsymbol{f}_{n_e}]^T \in \mathbb{R}^{n_e \times d}$, where $\boldsymbol{f}_i$ denotes the image-level feature encoded with proper attention to highlight text entity e_i. For example, given e_i = "glasses" in Fig. 2, the model needs to shift attention to the glasses visual region by adjusting the i-th row of $A^{(0)}$. The final image-level entity classification score is given by $P_{cls} \in \mathbb{R}^{n_e \times c_e}$, and the grounding module is trained using cross-entropy.

$$F = A^{(0)}H_{cls},\ F' = FW_{ent}^T$$
$$P_{cls}[i,j] = \frac{\exp(F'[i,j])}{\sum_{k=1}^{c_e} \exp(F'[i,k])} \quad (2)$$
$$L_{grd} = -\sum_{i=1}^{n_e}\sum_{j=1}^{c_e} Y_{ent}[i,j] \log P_{cls}[i,j] \quad (3)$$

3.3. Initial scene graph generation

Thus far, the text entity embeddings $H_{ent}^{(0)}$ played a role in the grounding procedure, and so did the one-hot encoded

label Y_{ent} extracted from the caption. Next, the model learns to predict the entities and relations without help from captions, which will not be available at inference time.

To this end, given entities $E = [e_1 \cdots e_{n_e}]^T$, relations $R = [(r_1, s_1, o_1) \cdots (r_{n_r}, s_{n_r}, o_{n_r})]^T$, and grounded boxes $[v_{\boldsymbol{g}_1^{(0)}} \cdots v_{\boldsymbol{g}_{n_e}^{(0)}}]^T$, we first parse the *target* instance labels. We extract $Y_{det}^{(0)} \in \mathbb{R}^{n_v \times c_e}$ and $Y_{relsub}, Y_{relobj} \in \mathbb{R}^{n_v \times c_r}$ using Eq. 4, in which all non-mentioned matrix entries are set to 0. $Y_{det}^{(0)}[i,j] = 1$ means visual region v_i involves the j-th entity class. $Y_{relsub}[i,j] = 1$ denotes the potential subject visual region v_i (e.g. a "person" region) may apply the j-th relation (e.g. "ride") to an unknown object. $Y_{relobj}[i,j] = 1$ denotes an unknown subject may apply the j-th relation to the potential object visual region v_i (e.g. a "horse" region). We add $_{rel}$ to highlight Y_{relsub}, Y_{relobj} are relation instance-level labels, but are attached to the grounded subject and object visual boxes respectively.

$$
\begin{aligned}
&Y_{det}^{(0)}[i,j] = 1 \text{ if } \exists k \in \{1 \cdots n_e\}, s.t. (\boldsymbol{g}_k^{(0)} = i, e_k = j) \\
&Y_{relsub}[i,j] = 1 \text{ if } \exists k \in \{1 \cdots n_r\}, s.t. (\boldsymbol{g}_{s_k}^{(n_t)} = i, r_k = j) \\
&Y_{relobj}[i,j] = 1 \text{ if } \exists k \in \{1 \cdots n_r\}, s.t. (\boldsymbol{g}_{o_k}^{(n_t)} = i, r_k = j)
\end{aligned} \tag{4}
$$

We next learn to predict the instance-level labels based on these targets, using entity detection head $H_{det}^{(0)} \in \mathbb{R}^{n_v \times d}$, and relation detection heads $H_{relsub}, H_{relobj} \in \mathbb{R}^{n_v \times d}$. Then, we matrix-multiply the three heads to the entity embedding $W_{ent} \in \mathbb{R}^{c_e \times d}$ and relation embedding $W_{rel} \in \mathbb{R}^{c_r \times d}$, and softmax-normalize, resulting in entity detection scores $P_{det}^{(0)} \in \mathbb{R}^{n_v \times c_e}$ and subject/object detection scores $P_{relsub}, P_{relobj} \in \mathbb{R}^{n_v \times c_r}$. We use cross-entropy loss terms $L_{det}^{(0)}, L_{relsub}, L_{relobj}$ similar to Eq. 3 to approximate $P_{det}^{(0)} \sim Y_{det}^{(0)}$, $P_{relsub} \sim Y_{relsub}$, and $P_{relobj} \sim Y_{relobj}$.

$$
\begin{aligned}
&X \in \{det, relsub, relobj\},\ W' \in \{W_{ent}, W_{rel}\} \\
&H_X = V_{feat} W_X, F_X = H_X W'^T \\
&P_X[i,j] = \frac{\exp(F_X[i,j])}{\sum_k \exp(F_X[i,k])}
\end{aligned} \tag{5}
$$

After training the aforementioned model, we can detect entities using $P_{det}^{(0)} \in \mathbb{R}^{n_v \times c_e}$ and detect relations using $P_{rel} \in \mathbb{R}^{n_v \times n_v \times c_r}$, where $P_{rel}[i,j,k] = \min(P_{relsub}[i,k], P_{relobj}[j,k])$. Intuitively, we treat the relation as valid if it could be both implied from the subject and object visual regions. For example, if the model infers "ride" from the "person" region and estimates "ride" can also apply to object region "horse", it determines that "ride" is the proper predicate bridging the two regions. [54, 57] proposed similar architectures to infer relation from a single region, [54] for optimizing runtime and [57] to avoid bad solutions. We use this idea because it is simple and effective, in combination with our stronger module in Sec. 3.5.

Test time post-processing. Given $P_{det}^{(0)}$, and P_{rel}, we adopt the *top-K predictions* (in experiments, k=50, 100) denoted in Eq. 6 as the initial scene graph generation (SGGen) results. In Eq. 6, the universal set $U = \{(v_{s_i^v}, v_{o_i^v}, s_i^e, p_i^r, o_i^e)\}_i$ denotes all possible 5-tuple combinations and B is a subset of U of size k. The goal is to seek the subset $B (B \subset U \text{ and } |B| = k)$ such that the sum of log probabilities is maximized. Within a specific B, $s^v, o^v \in \{1 \cdots n_v\}$ are the indices of proposal boxes to represent the subject and object regions, respectively; $s^e, o^e \in \{1 \ldots c_e\}$ are subject and object entity class IDs; $p^r \in \{1 \ldots c_r\}$ is the relation class ID. To implement Eq. 6 in practice, we use non-max suppression on $P_{det}^{(0)}$ to reduce the search space (ruling out unlikely classes and boxes).

$$
\begin{aligned}
SG_{init} = \underset{B \subset U, |B|=k}{\arg\max} \sum_{(s^v, o^v, s^e, p^r, o^e) \in B} \Big(&\log P_{det}^{(0)}[s^v, s^e] \\
&+ \log P_{rel}[s^v, o^v, p^r] + \log P_{det}^{(0)}[o^v, o^e] \Big)
\end{aligned} \tag{6}
$$

3.4. Iterative detection scores estimation

Careful readers may notice the superscript $^{(0)}$ in grounding vector $\boldsymbol{g}^{(0)}$, attention $A^{(0)}$, instance label $Y_{det}^{(0)}$, and instance prediction $P_{det}^{(0)}$. We use the superscript $^{(0)}$ to denote these are initial grounding results, which could be improved by the WSOD iterative refining technique proposed in [45]. Suppose loss $L_{det}^{(t)}$ $(t \geq 0)$ brings $P_{det}^{(t)} \in \mathbb{R}^{n_v \times c_e}$ close to $Y_{det}^{(t)} \in \mathbb{R}^{n_v \times c_e}$, where $Y_{det}^{(t)}$ is the caption-guided target label and $P_{det}^{(t)}$ is the prediction without help from captions. We could then incorporate the entity information $E = [e_1 \cdots e_{n_e}]^T$ of the caption into $P_{det}^{(t)}$ to turn it into a stronger instance-level label $Y_{det}^{(t+1)}$. The motivation is that the initial label $Y_{det}^{(0)}$ extracted from attention (Eq. 1, 4) will be easily influenced by the *noise in captions*. Since the attention scores always sum to one, some region will be assigned a higher score than others, regardless of whether the objects have consistent visual appearance. In an extreme case, mentioned but not visually present entities also have a matched proposal. Using $P_{det}^{(t)}$ is an indirect way to also consider the visual model's (Eq. 5) output, which encodes the objects' consistent appearance.

To turn $P_{det}^{(t)}$ into $Y_{det}^{(t+1)}$, we first extract $A^{(t+1)} \in \mathbb{R}^{n_e \times n_v}$ (same shape as the attention matrix $A^{(0)}$). We simply select the columns (denoted as $[:, i]$) from $P_{det}^{(t)}$ according to E to achieve $A^{(t+1)}$, and compute $\boldsymbol{g}^{(t+1)}$ and $Y_{det}^{(t+1)}$.

$$
\begin{aligned}
&A^{(t+1)} = \left[P_{det}^{(t)}[:, e_1] \cdots P_{det}^{(t)}[:, e_{n_e}] \right]^T \\
&\boldsymbol{g}^{(t+1)} = \underset{j \in \{1 \cdots n_v\}}{\arg\max}\ A^{(t+1)}[i,j] \\
&Y_{det}^{(t+1)}[i,j] = 1 \text{ if } \exists k \in \{1 \cdots n_e\}, s.t. (\boldsymbol{g}_k^{(t+1)} = i, e_k = j)
\end{aligned} \tag{7}
$$

We refine the model n_t times, and in Eq. 4, we use $\boldsymbol{g}^{(n_t)}$ from the last iteration to compute Y_{relsub} and Y_{relobj}.

3.5. Modeling SEQUENTIAL context

We observed the model sometimes generates triplets that violate common sense, e.g., plate-on-pizza in Fig. 5 top, because the aforementioned test time post-processing (Eq. 6) considers predictions from P_{det} and P_{rel} separately. When joined, the results may not form a meaningful triplet. To solve the problem, we propose a vision-language module to consider sequential patterns summarized from the dataset (Fig. 2 middle-bottom). The idea is inspired by [27], but different because: (1) we encode the language and vision priors within the same multi-modal RNN while [27] models vision and language separately, and (2) our label generation captures a language N-gram such that the later generated object and predicate will not contradict the subject.

Specifically, we gather the grounded tuples $D_{gt} = \{(v_{\boldsymbol{g}_{s_i}}, v_{\boldsymbol{g}_{o_i}}, e_{s_i}, r_i, e_{o_i})\}_{i=1}^{n_r}$ within each training example to learn the sequential patterns. Compared to the SGGen 5-tuple (Eq. 6), the e_{s_i}, r_i, e_{o_i} here are from the ground-truth (E, R) and are always correct (e.g., no "cake-eat-person"). Since the module receives high-quality supervision from captions, it will assign low scores or adjust the prediction (Eq. 6) for imprecise 5-tuples at test time, using its estimate of what proper 5-tuples look like.

Fig. 2 middle-bottom shows the idea. We use an RNN (LSTM in our implementation) to consume both word embeddings and visual features of the subject and object. The training outputs are subject prediction $P_{cssub} \in \mathbb{R}^{n_r \times c_e}$ (cs for *c*ommon *s*ense), object prediction $P_{csobj} \in \mathbb{R}^{n_r \times c_e}$, and predicate prediction $P_{cspred} \in \mathbb{R}^{n_r \times c_r}$. We now explain how to generate their i-th row (to match true e_{s_i}-r_i-e_{o_i}).

First, we feed into the RNN a dummy */start/* embedding and the grounded subject visual feature $\boldsymbol{v}_{\boldsymbol{g}_{s_i}}$. The subject prediction $P_{cssub}[i,:]$ is achieved by a linear layer projection (from RNN output to d-dim) and matrix multiplication (using $W_{ent} \in \mathbb{R}^{c_e \times d}$). We predict the object $P_{csobj}[i,:]$ similarly, but using the grounded object visual feature $\boldsymbol{v}_{\boldsymbol{g}_{o_i}}$ concatenated with the subject word embedding e_{s_i} as inputs. If we do not consider the visual input, this step is akin to learning a subject-object 2-gram language model. Next, the RNN predicts predicate label $P_{cspred}[i,:]$ (using $W_{rel} \in \mathbb{R}^{c_r \times d}$ instead of W_{ent}), using object word embedding e_{o_i} and a dummy visual feature $\emptyset$ as inputs.

To learn $P_{cssub}, P_{csobj}, P_{cspred}$, we extract labels $Y_{cssub}, Y_{csobj}, Y_{cspred}$ (Eq. 8) and use cross-entropy losses $L_{cssub}(P_{cssub} \sim Y_{cssub})$, $L_{csobj}(P_{csobj} \sim Y_{csobj})$, $L_{cspred}(P_{cspred} \sim Y_{cspred})$ to optimize the RNN model.

$$Y_{cssub} = \left[Y_{ent}[e_{s_i},:]^T \cdots Y_{ent}[e_{s_{n_r}},:]^T\right]^T \tag{8}$$

$$Y_{csobj} = \left[Y_{ent}[e_{o_i},:]^T \cdots Y_{ent}[e_{o_{n_r}},:]^T\right]^T,\ Y_{cspred} = Y_{rel}$$

At test time, we feed to the RNN the visual features from SG_{init} (Eq. 6) and the */start/* embedding. We let the RNN re-label the subject-object-predicate using beam search. The final score for each re-labeled 5-tuple is the sum of log probabilities of generating subject, object, and predicate. We generate the object before the predicate because objects are usually more distinguishable than predicates, so this order simplifies inference, allowing the use of a smaller beam size. We re-rank the beam search results using the final scores and keep the top ones to compute the Recall@k to evaluate (examples in Fig. 5, Fig. 6).

Our final model is trained using the following multi-task loss, where β is set to 0.5 since at the core of the task is the grounding of visual objects.

$$\begin{aligned} L = & L_{grd} + \beta\Big(\sum_{t=0}^{n_t} L_{det}^{(t)} + L_{relsub} + L_{relobj} \\ & + L_{cssub} + L_{csobj} + L_{cspred}\Big) \end{aligned} \tag{9}$$

4. Experiments

Datasets. We use the Visual Genome (VG) [22] and Common Objects in Context (COCO) [25] datasets, which both provide captions describing the visual contents. VG involves 108,077 images and 5.4 million region descriptions. The associated annotations of 3.8 million object instances and 2.3 million relationships enable us to evaluate the scene graph generation performance. To fairly compare to the counterpart weakly-supervised scene graph generation methods [57, 54], we adopt the VG split used in Zareian *et al.* [54]: keeping the most frequent $c_e = 200$ entity classes and $c_r = 100$ predicate classes, resulting in 99,646 images with *subject-predicate-object* annotations. We use the same 73,791/25,855 train/test split[1]. We also adopt the split in Xu *et al.* [48], more commonly used by fully-supervised methods. It contains 75,651/32,422 train/test images and keeps $c_e = 150$ entity and $c_r = 50$ predicate classes. Both VG splits are preprocessed by [54].

For COCO data, we use the 2017 training split (118,287 images). We rule out the duplicated images in the VG test set, resulting in 106,401 images for Zareian *et al.*'s split and 102,786 images for Xu *et al.*'s.

Learning tasks. The linguistic structure supervision for training is from the following three sources:

- VG-GT-Graph imagines an ideal scenario (an upper bound with the noise in captions and parsers' impacts isolated) where we have the ground-truth text graph annotations instead of a set of image-level *subject-predicate-object* triplets, for training on VG. To get these ground-truth graphs, we check the visual regions associated with the entities (subjects and objects) and connect entities if their regions have IoU greater than 0.5. We do *not* use box annotations to improve detection results.

[1] We follow [54], but [57] reports 73,801/25,857 train/test split

Zareian *et al.*'s split (weakly sup)			Xu *et al.*'s split (fully sup)		
Method	R@50	R@100	Method	R@50	R@100
			IMP [48]	3.44	4.24
VtranE-MIL [56]	0.71	0.90	MotifNet [55]	6.90	9.10
PPR-FCN-single [57]	1.08	1.63	Asso.Emb. [34]	9.70	11.30
PPR-FCN [57]	1.52	1.90	MSDN [24]	10.72	14.22
VSPNet [54]	3.10	3.50	Graph R-CNN [49]	11.40	13.70
			VSPNet (Full) [54]	12.60	14.20
BASIC	2.20	2.88	BASIC	3.82	4.96
+ PHRASAL	2.77	3.62	+ PHRASAL	4.04	5.21
+ ITERATIVE	3.26	4.15	+ ITERATIVE	6.06	7.60
+ SEQUENTIAL	4.92	5.84	+ SEQUENTIAL	7.30	8.73

Table 2. SGGen recall (%) under VG-GT-Graph setting. We compare our method to the state-of-the-art methods. High recall (R@50, R@100) is good.

- VG-Cap-Graph utilizes the VG *region* descriptions. We use [41] to extract text graphs from these descriptions, but we ignore the region coordinates and treat the graphs as image-level annotations.
- COCO-Cap-Graph uses captions from COCO and applies the same parsing technique as VG-Cap-Graph. The difference is that these captions are image-level, and describe the objects and relations as a whole.

Metrics. We measure how accurately the models generate scene graphs, using the densely-annotated scene graphs in the VG test set. Following [48], a predicted triplet is considered correct if the three text labels are correct and the boxes for subject/object have ≥ 0.5 IoU with ground-truth boxes. We then compute the Recall@50 and Recall@100 as the fraction of the ground-truth triplets that are successfully retrieved in the top-50 and top-100 predictions, respectively.

Methods compared. We conduct ablation studies to verify the benefit of each component of our method.

- BASIC model refers to our Sec. 3.2-3.3 without applying the phrasal contextualization. We set $\psi(E, G^L) = H^{(0)}_{ent}$.
- +PHRASAL context (Sec. 3.1) uses contextualized entity embeddings $\psi(E, G^L)$ instead of $H^{(0)}_{ent}$.
- +ITERATIVE (Sec. 3.4) gradually improves the grounding vector $\boldsymbol{g}$. We iterate $n_t = 3$ times by default.
- +SEQUENTIAL context (Sec. 3.5) revises the prediction presented in Eq. 6, using the RNN encoded with knowledge regarding sequential patterns.

We compare to weakly-supervised scene graph generation methods that published results on Zareian *et al.*'s split: VtransE-MIL [56], PPR-FCN-single [57], PPR-FCN [57] and VSPNet [54]. We also compare to fully-supervised methods on Xu *et al.*'s split: Iterative Message Passing (IMP) [48], Neural Motif Network (MotifNet) [55], Associative Embedding (Asso.Emb.) [34], Multi-level Scene Description Network (MSDN) [24], Graph R-CNN [49], and fully-supervised VSPNet [54].

Eval split	VG-Cap-Graph				COCO-Cap-Graph			
	Zareian *et al.*'s		Xu *et al.*'s		Zareian *et al.*'s		Xu *et al.*'s	
	R@50	R@100	R@50	R@100	R@50	R@100	R@50	R@100
BASIC	0.81	0.91	0.99	1.09	1.20	1.51	2.09	2.63
+ PHRASAL	0.90	1.04	1.39	1.69	1.17	1.47	1.65	2.16
+ ITERATIVE	1.11	1.32	1.79	2.22	1.41	1.75	2.41	3.02
+ SEQUENTIAL	1.83	1.94	3.85	4.04	1.95	2.23	3.28	3.69

Table 3. SGGen recall (%) under Cap-Graph settings. High recall (R@50, R@100) is good.

4.1. Results on GT-Graph setting

The GT-Graph setting allows our method to be fairly compared to the state-of-the-art methods because in this setting, the information ours and those methods receive is comparable (sets of triplets, in our case connected). Further, the word distribution is the same for training/testing, while the caption setting causes a train-test shift (described shortly).

In Tab. 2 left, we show our results on Zareian *et al.*'s VG split and baselines of weakly-supervised methods. Our BASIC method already surpasses VtransE-MIL, PPR-FCN-single, and PPR-FCN. This may be due to the low quality of the EdgeBox proposals used in them. Compared to VSPNet, which also uses Faster RCNN proposals, our BASIC method is slightly worse, but our components greatly improve upon BASIC, and our final model achieves 4.92, a 59% improvement over VSPNet (using R@50). +PHRASAL context improves BASIC by 26% (2.77 v.s. 2.20), +ITERATIVE improves +PHRASAL by 18% (3.26 v.s. 2.77), and +SEQUENTIAL gains 51% (4.92 v.s. 3.26).

In Tab. 2 right, we compare to fully-supervised methods on Xu *et al.*'s split. We observe our method is very competitive even though we only use image-level annotations. In terms of Recall@50, our final method (7.30) outperforms IMP (3.44) and MotifNet (6.90). As for the relative improvement, +PHRASAL context improves BASIC by 6% (4.04 v.s. 3.82), +ITERATIVE gains 50% (6.06 v.s. 4.04), and +SEQUENTIAL gains 20% (7.30 v.s. 6.06).

4.2. Results on Cap-Graph setting

Our proposed Cap-Graph setting is an under-explored and challenging one, as the learned SGGen model depends on the captions' exhaustiveness and the parser's quality, but it allows learning from less expensive image-text data.

In Tab. 3, we show the SGGen performance of models learned from VG region captions (VG-Cap-Graph) and COCO image captions (COCO-Cap-Graph). We see the same trend as in GT-Graph setting: our components (+PHRASAL, +ITERATIVE, and +SEQUENTIAL) have positive effects. Further, our final models learned from both VG-Cap-Graph (R@50 1.83) and COCO-Cap-Graph (1.95) are better than all weakly-supervised methods except VSPNet (in Tab. 2 left). Our models learned from captions are even comparable (VG-Cap-Graph 3.85, COCO-Cap-Graph 3.28) to the fully-supervised IMP (R@50 3.44).

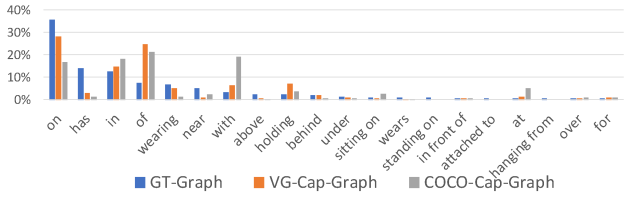

Figure 3. Relation frequencies in the three settings.

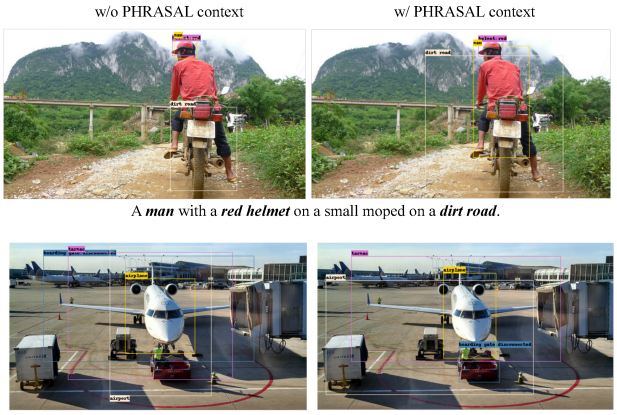

Figure 4. Importance of PHRASAL context; best seen with zoom.

Fig. 3 shows the relation frequencies in our settings. We observe that some relations ("has," "near") rarely appear in text descriptions but are often annotated in ground-truth scene graphs. Meanwhile, there are frequently mentioned prepositions in captions ("of," "with," "at") which are rarely denoted as relations. These train/test discrepancies (train on captions, test on triplets) explain our methods' relative performance in Tab. 3, where +PHRASAL helps under VG-Gap-Graph (more similar to GT-Graph) but hurts slightly under COCO-Cap-Graph (less similar to GT-Graph).

4.3. Qualitative examples

Fig. 4 compares using and not using PHRASAL context. Without out PHRASAL module (left), the grounding procedure gets stuck on the same distinguishable local region (top-left: head of man) or erroneously attends to the whole image (bottom-left: boarding gate). When using the PHRASAL module (right), our model is better at localizing visual objects. It knows there should be a complete person in the scene (top-right) and the boarding gate is a concept related to the plane (bottom-right).

Fig. 5 shows how the learned sequential patterns help correct imprecise predictions. For the corrections (beam size=5), we show the log-probability of the 5-tuple and individual probabilities. Given that *plate* cannot be put *on pizza*, our model corrects it to *plate-under-pizza*. In the bottom example, our model corrects *person-wear-person* to *person-wear-shirt* and *person-behind-person*. In Fig. 6, we compare our BASIC and final methods.

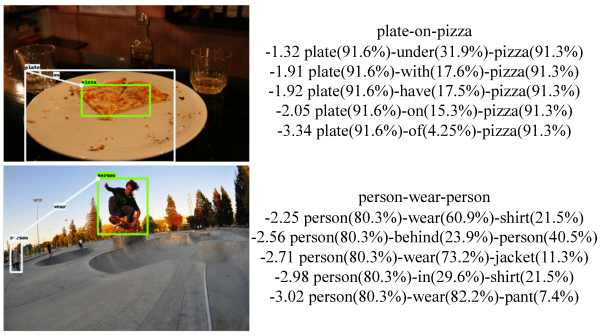

Figure 5. Importance of SEQUENTIAL context.

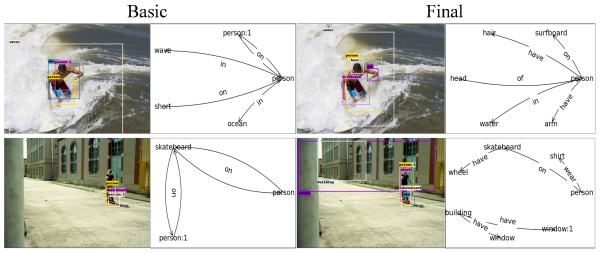

Figure 6. BASIC v.s. our final model; best viewed with zoom.

4.4. Implementation details

We pre-extract text graphs using [47]'s implementation of [41]. We use the same proposals ($n_v = 20$ per image) and features ($d_{cnn} = 1536$) as [54], extracted using Faster-RCNN [39] (InceptionResnet backbone [44]) pre-trained on OpenImage [23]. During training, we use GraphNets [4] to encode phrasal context. W_{ent}, W_{rel} are $d = 300$ frozen GloVe embeddings [36]. To train our model, we use a batch size of 32, learning rate 0.00001, the Adam optimizer [21], and Tensorflow distributed training [1]. We use weight decay of 1e-6 and the random normal initializer (mean=0.0, stdev=0.01) for all fully-connected layers. We use LSTM cell, 100 hidden units, and dropout 0.2, for the SEQUENTIAL module. For the non-max-suppression of Eq. 6, we use score threshold 0.01, IoU threshold 0.4, and limit the maximum instances per entity class to 4. We set beam size to 5 for the SEQUENTIAL module post-processing.

5. Conclusion

We introduced a method that leverages caption supervision for scene graph generation. Captions are noisy, but can be obtained "for free," and allow us to understand high-order relations between entities and triplets. To leverage caption supervision, we proposed to capture commonsense relations (phrasal and sequential context) and iteratively refine detection scores. In the future, we will explicitly handle distribution shifts between captions and text graphs.

Acknowledgements: This work was partly supported by National Science Foundation Grant No. 1718262 and a Univ. of Pittsburgh Computer Science fellowship. We thank the reviewers and AC for their encouragement.

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning. In *Proceedings of the USENIX Conference on Operating Systems Design and Implementation (OSDI)*, November 2016. 8

[2] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2

[3] Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, July 2015. 2

[4] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018. 4, 8

[5] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2

[6] Kan Chen, Jiyang Gao, and Ram Nevatia. Knowledge aided consistency for weakly supervised phrase grounding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[7] Kai Chen, Hang Song, Chen Change Loy, and Dahua Lin. Discover and learn new objects from documentaries. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2

[8] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[9] Luciano Del Corro and Rainer Gemulla. Clausie: clause-based open information extraction. In *Proceedings of the International Conference on World Wide Web (WWW)*, 2013. 2

[10] Karan Desai and Justin Johnson. VirTex: Learning Visual Representations from Textual Annotations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021. 2

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, June 2019. 1

[12] Anthony Fader, Stephen Soderland, and Oren Etzioni. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, July 2011. 2

[13] Omer Goldman, Veronica Latcinnik, Ehud Nave, Amir Globerson, and Jonathan Berant. Weakly supervised semantic parsing with abstract examples. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, July 2018. 2

[14] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[15] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015. 2

[16] Achiya Jerbi, Roei Herzig, Jonathan Berant, Gal Chechik, and Amir Globerson. Learning object detection from captions via textual scene attributes. *arXiv preprint arXiv:2009.14558*, 2020. 2

[17] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Judy Hoffman, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. Inferring and executing programs for visual reasoning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[18] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2

[19] Andrej Karpathy, Armand Joulin, and Li F Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014. 2

[20] Eun-Sol Kim, Woo Young Kang, Kyoung-Woon On, Yu-Jung Heo, and Byoung-Tak Zhang. Hypergraph attention networks for multimodal learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[21] Diederick P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015. 8

[22] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision (IJCV)*, 123(1), 2017. 6

[23] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision (IJCV)*, 2020. 8

[24] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2, 7

[25] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014. 6

[26] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[27] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016. 1, 2, 6

[28] Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, and Jiajun Wu. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. In *Proceedings of the International Conference on Learning Representations (ICLR)*, May 2019. 2

[29] Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. Open language learning for information extraction. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, July 2012. 2

[30] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[31] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[32] Ishan Misra, C. Lawrence Zitnick, Margaret Mitchell, and Ross Girshick. Seeing through the human reporting bias: Visual classifiers from noisy human-centric labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1

[33] Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[34] Alejandro Newell and Jia Deng. Pixels to graphs by associative embedding. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 1, 2, 7

[35] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Is object localization for free? - weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2

[36] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Oct. 2014. 4, 8

[37] Julia Peyre, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Weakly-supervised learning of visual relations. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[38] Mengshi Qi, Weijian Li, Zhengyuan Yang, Yunhong Wang, and Jiebo Luo. Attentive relational networks for mapping images to scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[39] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015. 3, 8

[40] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of textual phrases in images by reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016. 2

[41] 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 *Workshop on Vision and Language (VL15)*, Sept. 2015. 2, 3, 7, 8

[42] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, July 2018. 2

[43] Dídac Surís, Dave Epstein, Heng Ji, Shih-Fu Chang, and Carl Vondrick. Learning to learn words from visual scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[44] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, number 1, 2017. 8

[45] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3, 5

[46] Yu-Siang Wang, Chenxi Liu, Xiaohui Zeng, and Alan Yuille. Scene graph parsing as dependency parsing. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, June 2018. 2

[47] Hao Wu, Jiayuan Mao, Yufeng Zhang, Yuning Jiang, Lei Li, Weiwei Sun, and Wei-Ying Ma. Unified visual-semantic embeddings: Bridging vision and language with structured meaning representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2, 8

[48] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 3, 6, 7

[49] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 1, 2, 7

[50] Alexander Yates, Michele Banko, Matthew Broadhead, Michael Cafarella, Oren Etzioni, and Stephen Soderland. TextRunner: Open information extraction on the web. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, Apr. 2007. 2

[51] Keren Ye, Mingda Zhang, Adriana Kovashka, Wei Li, Danfeng Qin, and Jesse Berent. Cap2det: Learning to amplify weak caption supervision for object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, October 2019. 1, 2

[52] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 2

[53] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics (TACL)*, 2, 2014. 2

[54] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Weakly supervised visual semantic parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2, 3, 5, 6, 7, 8

[55] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2, 7

[56] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3, 7

[57] Hanwang Zhang, Zawlin Kyaw, Jinyang Yu, and Shih-Fu Chang. Ppr-fcn: Weakly supervised visual relation detection via parallel pairwise r-fcn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 1, 2, 3, 5, 6, 7

[58] Fang Zhao, Jianshu Li, Jian Zhao, and Jiashi Feng. Weakly supervised phrase localization with multi-scale anchored transformer network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2