G^{3}RAPHGROUND: Graph-based Language Grounding

Mohit Bajaj[1,2] Lanjun Wang[3] Leonid Sigal[1,2,4]
[1]University of British Columbia [2]Vector Institute for AI
[3]Huawei Technologies [4]Canada CIFAR AI Chair
mbajaj@alumni.ubc.ca lanjun.wang@huawei.com lsigal@cs.ubc.ca

Abstract

In this paper we present an end-to-end framework for grounding of phrases in images. In contrast to previous works, our model, which we call G^{3}RAPHGROUND*, uses graphs to formulate more complex, non-sequential dependencies among proposal image regions and phrases. We capture intra-modal dependencies using a separate graph neural network for each modality (visual and lingual), and then use conditional message-passing in another graph neural network to fuse their outputs and capture cross-modal relationships. This final representation results in grounding decisions. The framework supports many-to-many matching and is able to ground single phrase to multiple image regions and vice versa. We validate our design choices through a series of ablation studies and illustrate state-of-the-art performance on Flickr30k and ReferIt Game benchmark datasets.*

1. Introduction

Over the last couple of years, phrase (or more generally language) grounding has emerged as a fundamental task in computer vision. Phrase grounding is a generalization of the more traditional computer vision tasks, such as object detection [11] and semantic segmentation [27]. Grounding requires spatial localization of free-form linguistic phrases in images. The core challenge is that the space of natural phrases is exponentially large, as compared to, for example, object detection or segmentation where the label sets are typically much more limited (*e.g.*, 80 categories in MS COCO [18]). This exponential expressivity of the label set necessitates amortized learning, which is typically formulated using continuous embeddings of visual and lingual data. Despite challenges, phrase grounding emerged as the core problem in vision due to the breadth of applications that span image captioning [19], visual question answering [2, 40] and referential expression recognition [20] (which is at the core of many HCI and HRI systems).

Significant progress has been made on the task in the last couple of years, fueled by large scale datasets (*e.g.*, Flickr30k [24] and ReferIt Game [14]) and neural architectures of various forms. Most approaches treat the problem as one of learning an embedding where class-agnostic region proposals [25] or attended images [8, 34] are embedded close to the corresponding phrases. A variety of embedding models, conditional [22] and unconditional [13, 29], have been proposed for this task. Recently, the use of contextual relationships among the regions and phrases have started to be explored and shown to substantially improve the performance. Specifically, [9] and [6] encode the context of previous decisions by processing multiple phrases sequentially, and/or contextualizing each decision by considering other phrases and regions [9]. Non-differentiable process using policy gradient is utilized in [6], while [9] uses an end-to-end differentiable formulation using LSTMs. In both cases, the contextual information is modeled using sequential propagation (*e.g.*, using LSTMs [6, 9]).

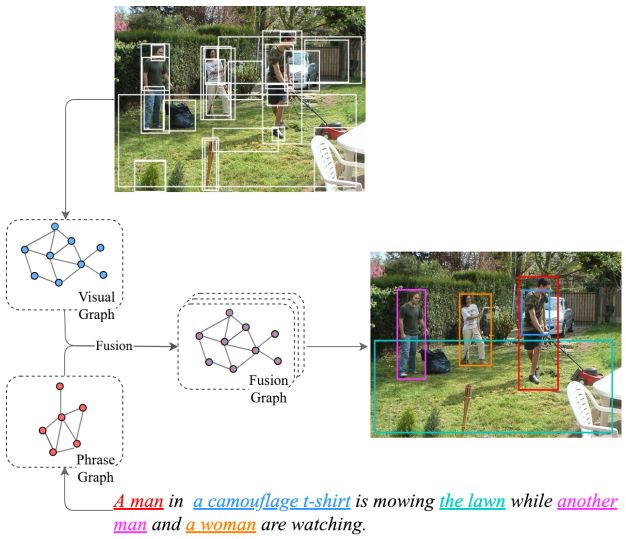

Figure 1. **Illustration of G^{3}RAPHGROUND.** Two separate graphs are formed for phrases and image regions respectively, and are then fused together to make final grounding predictions. The colored bounding-boxes correspond to the phrases in same color.

In reality, contextual information in the image, *e.g*., among the proposed regions, can hardly be regarded as sequential. Same can be argued for phrases, particularly in cases where they do not come from an underlying structured source like a sentence (which is explicitly stated as an assumption and limitation of [9]). In essence, previous methods impose sequential serialization of fundamentally non-sequential data for convenience. We posit that addressing this limitation explicitly can lead to both better performance and more sensibly structured model. Capitalizing on the recent advances in object detection, that have addressed conceptually similar limitations with the use of transitive reasoning in graphs (*e.g*., using convolutional graph neural networks [15, 17, 36]), we propose a new graph-based framework for phrase grounding. Markedly, this formulation allows us to take into account more complex, non-sequential dependencies among both proposal image regions and the linguistic phrases that require grounding.

Specifically, as illustrated in Figure 1, region proposals are first extracted from the image and encoded, using CNN and bounding-box coordinates, into node features of the *visual graph*. The phrases are similarly encoded, using bi-directional RNN, into node features of the *phrase graph*. The strength of connections (edge weights) between the nodes in both graphs are predicted based on the corresponding node features and the global image/caption context. Gated Graph Neural Networks (GG-NNs) [17] are used to refine the two feature representations through a series of message-passing iterations. The refined representations are then used to construct the *fusion graph* for each phrase by fusing the *visual graph* with the selected phrase. Again the fused features are refined using message-passing in GG-NN. Finally, the fused features for each node, that corresponds to the encoding of <$\texttt{phrase}_i$, $\texttt{image\_region}_j$> tuples, are used to predict the probability of grounding $\texttt{phrase}_i$ to $\texttt{image\_region}_j$. These results are further refined by simple scheme that does non-maxima suppression (NMS), and predicts whether a given phrase should be grounded to one or more regions. The final model, we call G^{3}RAPHGROUND, is end-to-end differentiable and is shown to produce state-of-the-art results.

While we clearly designed our architecture with phrase grounding in mind, we want to highlight that it is much more general and would be useful for any multi-modal assignment problem where some contextual relations between elements in each modality exist. For example, text-to-clip [35] / caption-image [16, 39] retrieval or more general cross-modal retrieval and localization [3].

Contributions: Our contributions are multifold. First, we propose novel graph-based grounding architecture which consists of three connected sub-networks (visual, phrase and fusion) implemented using Gated Graph Neural Networks. Our design is modular and can model rich context both within a given modality and across modalities, without making strong assumptions on sequential nature of data. Second, we show how this architecture could be learned in an end-to-end manner effectively. Third, we propose a simple but very effective refinement scheme that in addition to NMS helps to resolve one-to-many groundings. Finally, we validate our design choices through a series of ablation studies; and illustrate up to 5.33% and 10.21% better than state-of-the-art performance on Flickr30k [24] and ReferIt Game [14] datasets.

2. Related Work

Our task of language (phrase) grounding is related to rich literature on vision and language; with architectural design building on recent advances in Graph Neural Networks. We review the most relevant literature and point the reader to recent surveys [1, 4] and [33, 41] for added context.

Phrase Grounding. Prior works, such as Karpathy *et al*. [13], propose to align sentence fragments and image regions in a subspace. Similarly, Wang *et al*. [30] propose a structured matching approach that encourages the semantic relations between phrases to agree with the visual relations between regions. In [29], Wang *et al*. propose to learn a joint visual-text embedding with symmetric distance where a given phrase is grounded to the closest bounding-box. The idea is further extended by similarity network proposed in [28] that uses a single vector for representing multi-modal features instead of an explicit embedding space. Plummer *et al*. [22] build on this idea and propose a concept weight branch to automatically assign the phrases to embeddings.

It has been shown that both textual and visual context information can aid phrase grounding. Plummer *et al*. [23] perform global inference using a wide range of visual-text constraints from attributes, verbs, prepositions, and pronouns. Chen *et al*. [6] try to leverage the semantic and spatial relationships between the phrases and corresponding visual regions by proposing a context policy network that accounts for the predictions made for other phrases when localizing a given phrase. They also propose and finetune the query guided regression network to boost the performance by better proposals and features. SeqGROUND [9] uses the full image and sentence as the global context while formulating the task as a sequential and contextual process that conditions the grounding decision of a given phrase on previously grounded phrases.Wang *et al*. [31] uses a graph to model the relationships between image-regions and localizes only one referring expression at a time.

Graph Neural Networks (GNNs). Graph Convolution Networks (GCNs) were first introduced in [15] for semi-supervised classification. Each layer of GCN can perform localized computations involving neighbourhood nodes.

These layers can be further stacked to form a deeper network that is capable of performing complex computations on graph data. In vision, Yang *et al*. [37] enhanced GCNs with attention and found them to be effective for scene graph generation; [32] deploy GCNs to model videos as space-time graphs and get impressive results for video classification task. Visual reasoning among image regions for object detection, using GCNs, was shown in [7] and served as conceptual motivation for our *visual graph* sub-network.

Recently, [36] present a theoretical framework for analyzing the expressive power of GNNs to capture different graph structures. They mention that message-passing in GNNs can be described by two fuctions: AGGREGATE and COMBINE. The AGGREGATE function aggregates the messages from the neighbourhood nodes and the COMBINE function updates the state of each node by combining the aggregated message and the previous state of each node. They prove that choice of these functions is crucial to the expressive power of GNNs. Li *et al*. [17] propose Gated Graph Neural Networks (GG-NNs) that use Gated Recurrent Units (GRUs) for the gating in the COMBINE step.

Our model is inspired by these works. We use one GG-NN to model the spatial relationships between the image regions and another to capture the semantic relationships between the phrases. We finally use the third GG-NN for the fusion of the text and visual embeddings obtained from the corresponding graphs. Output of the fusion network is used to predict if a given phrase should be grounded to a specific image region or not.

3. Approach

Phrase grounding is a challenging many-to-many matching problem where a single phrase can, in general, be grounded to multiple regions, or multiple phrases can be grounded to a single image region. The G^{3}RAPHGROUND framework uses graph networks to capture rich intra-modal and cross-modal relationships between the phrases and the image regions. We illustrate the architecture in Figure 2. We assume that the phrases are available, *e.g*. parsed from an image caption (Flickr30k [24] dataset) or exist independently for a single image (ReferIt Game [14] dataset).

We encode these phrases using a bi-directional RNN that we call *phrase encoder*. These encodings are then used to initialize the nodes of the *phrase graph* that is built to capture the relationships between the phrases. Similarly, we form the *visual graph* that models the relationships between the image regions that are extracted from the image using RPN and then encoded using the *visual encoder*. Caption and full image provide additional context information that we use to learn the edge-weights for both graphs. Message-passing is independently done for these graphs to update the respective node features. This allows each phrase/image region to be aware of other contextual phrases/image regions. We finally fuse the outputs of these two graphs by instantiating one *fusion graph* for each phrase. We concatenate the features of all nodes of the *visual graph* with the feature vector of a given node of the *phrase graph* to condition the message-passing in this new *fusion graph*.

The final state of each node of the *fusion graph*, that corresponds to a pair <phrase$_i$, image_region$_j$>, is fed to a fully connected *prediction network* to make a binary decision if phrase$_i$ should be grounded to image_region$_j$. Note that all predictions are implicitly inter-dependent due to series of message-passing iterations in three graphs. We also predict if the phrase should be grounded to a single or multiple regions and use this information for post processing to refine our predictions.

3.1. Text and Visual Encoders

Phrase Encoder. We assume one or more phrases are available and need to be grounded. Each phrase consists of a word or a sequence of words. We encode each word using its GLoVe [21] embedding and then encode the complete phrase using the last hidden state of a bi-directional RNN. Finally, we obtain phrase encodings $\mathbf{p}_1,\ldots,\mathbf{p}_n$ for the corresponding n input phrases P_1 ... P_n.

Caption Encoder. We use another bi-directional RNN to encode the complete input caption C and obtain the caption encoding $\mathbf{c}_{enc}$. This is useful as it provides global context information missing in the encodings of individual phrases.

Visual Encoder. We use a region proposal network (RPN) to extract region proposals R_1 ... R_m from an image. Each region proposal R_i is fed to the pre-trained VGG-16 network to extract 4096-dimensional vector from the first fully-connected layer. We transform this vector to 300 dimensional vector $\mathbf{r}_i$ by passing it through a network with three fully-connected layers with ReLU activations and a batch normalization layer at the end.

Image Encoder. We use same architecture as the *visual encoder* to also encode the full image into the corresponding 300 dimensional vector $\mathbf{i}_{enc}$ that serves as global image context for the grounding network.

3.2. G^{3}RAPHGROUND Network

Phrase Graph. To model relationships between the phrases, we construct the *phrase graph* $\mathcal{G}^P$ where nodes of the graph correspond to the phrase encodings and the edges correspond to the context among them. The core idea is to make grounding decision for each phrase dependent upon other phrases present in the caption. This provides with the important context for the grounding of the given phrase. Formally, $\mathcal{G}^P = (\mathcal{V}^P, \mathcal{E}^P)$ where $\mathcal{V}^P$ are the nodes corresponding to the phrases and $\mathcal{E}^P$ are the edges connecting these nodes. We model this using Gated Graph Neural Network where AGGREGATE step of the message-passing for

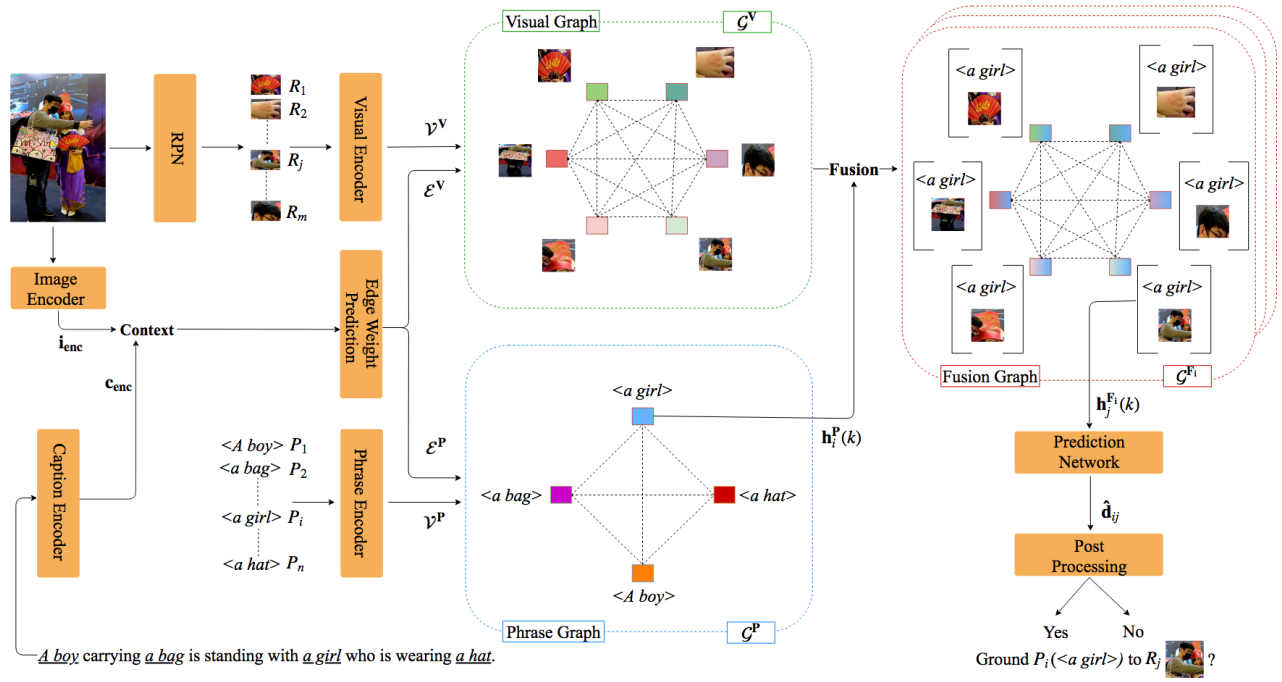

Figure 2. **G^{3}RAPHGROUND Architecture.** The phrases are encoded into the *phrase graph* while image regions are extracted and encoded into the *visual graph*. The *fusion graph* is formed by independently conditioning the *visual graph* on each node of the *phrase graph*. The output state of each node of the *fusion graph* after message-passing is fed to the *prediction network* to get the final grounding decision.

each node $v \in \mathcal{V}^P$ can be described as

$$\begin{aligned}\mathbf{a}_v^P(t) &= AGGREGATE(\{\mathbf{h}_u^P(t-1) : u \in \mathcal{N}(v)\}) \\ &= \sum_{u \in \mathcal{N}(v)} \{\mathbf{A}_{u,v}^P(\mathbf{W}_k^P \cdot \mathbf{h}_u^P(t-1))\},\end{aligned} \tag{1}$$

where $\mathbf{a}_v^P(t)$ is the aggregated message received by node v from its neighbourhood $\mathcal{N}$ during t^{th} iteration of message-passing, $\mathbf{h}_u^P(t-1)$ is a d-dimensional feature vector of phrase-node u before t^{th} iteration of message-passing, $\mathbf{W}_k^P \in \mathbb{R}^{d \times d}$ is a learnable $d \times d$ dimensional graph kernel matrix, and $\mathbf{A}_{u,v}^P$ corresponds to the scalar entry of learnable adjacency-matrix that denotes the weight of the edge connecting the nodes u and v.

We initialize $\mathbf{h}_u^P(0)$ with the corresponding phrase encoding $\mathbf{p}_u \in \mathbb{R}^d$ produced by the *phrase encoder*. To obtain each entry of the adjacency-matrix $\mathbf{A}_{u,v}^P$, we concatenate the caption embedding ($\mathbf{c}_{enc}$), the full image embedding ($\mathbf{i}_{enc}$) and the sum of corresponding phrase embeddings: $\mathbf{p}_u$ and $\mathbf{p}_v$. The concatenated feature is passed through a two layer fully-connected network f_{adj} followed by sigmoid:

$$\mathbf{A}_{u,v}^P = \mathbf{A}_{v,u}^P = \sigma(f_{adj}(Concat(\mathbf{p}_u + \mathbf{p}_v, \mathbf{c}_{enc}, \mathbf{i}_{enc}))). \tag{2}$$

Aggregated message $\mathbf{a}_v^P(t)$ received by node v is used to update the state of node v during t^{th} iteration:

$$\mathbf{h}_v^P(t) = COMBINE(\{h_v^P(t-1), a_v^P(t)\}) \tag{3}$$

We use GRU gating in the COMBINE step as proposed by [17]. After k ($k = 2$ for all experiments) stages of message-passing on this graph-network, we obtain $\mathbf{h}_v^P(k)$ that encodes the final state for the phrase node $v \in \mathcal{V}^P$ of the *phrase graph*; these final states are then used in the fusion.

Visual Graph. Similarly, we instantiate another GG-NN to model the *visual graph* $\mathcal{G}^V$ that models the spatial relationships between the image regions present in the image. Each node of the graph corresponds to an image region extracted from RPN. To initialize the states of these nodes, we use the encoded features of the image regions produced by *visual encoder*, and concatenate them with the position of the corresponding image region in the image denoted by four normalized coordinates. $\mathcal{V}^V$ denotes the nodes of the *visual graph* $\mathcal{G}^V$. The AGGREGATE step of message-passing on this network for each node $v \in V^V$ can be described as:

$$\mathbf{a}_v^V(t) = \sum_{u \in \mathcal{N}(v)} \{\alpha_u(\mathbf{W}_k^V \cdot \mathbf{h}_u^V(t-1))\}, \tag{4}$$

where we initialize $\mathbf{h}_u^V(0)$ with the vector $[\mathbf{r}_u,\ x_u^{min},\ y_u^{min},\ x_u^{max},\ y_u^{max}]$ which is obtained after concatenating the visual encoding ($\mathbf{r}_u$) of the u^{th} image region and its normalized position, α_u represents the attention weight given to the node u during the message-passing. To obtain α_u, we concatenate the visual encoding $\mathbf{r}_u$ of that node with the caption encoding $\mathbf{c}_{enc}$ and the full image encoding $\mathbf{i}_{enc}$, and then pass this vector through a

fully-connected network f_{attn} followed by sigmoid:

$$\alpha_u = \sigma(f_{attn}(Concat(\mathbf{r}_u, \mathbf{c}_{enc}, \mathbf{i}_{enc}))). \quad (5)$$

This is similar to AGGREGATE step of message-passing on the *phrase graph* except we do not learn the complete adjacency matrix for this graph. We note that it is computationally expensive to learn this matrix as number of entries in adjacency matrix increase quadratically with the increase in number of the image regions. Instead we use unsupervised-attention $\boldsymbol{\alpha}$ over the nodes of the *visual graph* to decide the edge-weights. All edges that originate from the node u are weighted α_u where $\alpha_u \in [0, 1]$.

Similar to *phrase graph*, we use GRU mechanism [17] for the COMBINE step of message-passing on this graph. After k stages of message-passing on this graph-network we obtain $\mathbf{h}_v^V(k)$ that encodes the final state for the image region node $v \in V^V$ of the *visual graph*. The updated *visual graph* is conditioned on each node of the *phrase graph* in the fusion step that we explain next.

Fusion Graph. As we have phrase embeddings and image region embeddings from the *phrase graph* and the *visual graph* respectively, the *fusion graph* is designed to merge these embeddings before grounding decisions are made. One fusion graph is instantiated for each phrase. This instantiation is achieved by concatenating the features of all the nodes of the *visual graph* with the node features of the selected phrase node from the *phrase graph*. That is to say, the *fusion graph* has properties: 1) it has the same structure (*i.e.*, the number of nodes as well as the adjacency matrix) as the *visual graph*; 2) the number of fusion graphs instantiated is the same as the number of nodes in the *phrase graph*. We can also characterize this graph as *visual graph* conditioned on a node from the *phrase graph*.

After k iterations of message-passing in the *fusion graph*, we use the final state of each node to predict the grounding decision for the corresponding image region with respect to the phrase on which the corresponding fusion graph was conditioned. This is independently repeated for all of the phrases by instantiating a new *fusion graph* from the *visual graph* for each phrase, and conditioning the message-passing in this new graph on the selected phrase node of the *phrase graph*. Note that it may seem that message-passing in the *fusion graphs* occur independently for each phrase but it's not true. Each phrase embedding that is used to condition message-passing in the *fusion graph* is output of the *phrase graph*, and hence, is aware of other phrases present in the caption.

Let $\mathcal{G}^{F_i}$ denote the *fusion graph* obtained by conditioning the *visual graph* on node i of the *phrase graph*. The initialization of node j in this *fusion graph* can be described as:

$$\mathbf{h}_j^{F_i}(0) = Concat(\mathbf{h}_i^P(k), \mathbf{h}_j^V(k)), \forall j \in \mathcal{V}^V \quad (6)$$

where $\mathbf{h}_j^V(k)$ corresponds to the final feature vector of node j in the *visual graph* and $\mathbf{h}_i^P(k)$ is the final feature vector of the selected node i in the *phase graph*.

The AGGREGATE and COMBINE steps of message-passing on each *fusion graph* remain same as described for the *visual graph* in Eqs. (4) and (3).

Prediction Network. While grounding, we predict a scalar $\hat{d}_{ij}$ for each phrase-region pair that denotes the probability whether the phrase P_i is grounded to the image region R_j. The probability of this decision conditioned on the given image and caption can be approximated from the fused-embedding of that image region conditioned on the given phrase. We pass the fused-embedding of the node j of the *fusion graph* $\mathcal{G}^{F_i}$ through the *prediction network* f_{pred} which consists of three fully-connected layers with interleaved ReLU activations and a sigmoid function at the end.

$$P(\hat{d}_{ij} = 1 | \mathbf{h}_j^{F_i}(k)) = \sigma(f_{pred}(\mathbf{h}_j^{F_i}(k))) \quad (7)$$

Post Processing. Note that a given phrase may be grounded to a single or multiple regions. We find that the model's performance can be significantly boosted if we post process the grounding predictions for two cases separately. Hence, we predict a scalar $\hat{\beta}_v$ for each phrase $v \in \mathcal{V}^P$ which denotes the probability of the phrase to be grounded to more than one image region. We pass the updated phrase-embedding $\mathbf{h}_v^P(k)$ of node v obtained from the *phrase graph* through a 2-layered fully-connected network f_{count}:

$$\hat{\beta}_v = \sigma(f_{count}(h_v^P(k))), \quad (8)$$

If $\hat{\beta}_v$ is greater than 0.5, we select those image regions for which the output of the *prediction network* are above a fixed threshold and then apply non-maximum suppression (NMS) as a final step. Otherwise, we simply ground the phrase to the image region with the maximum decision probability output from the *prediction network*.

Training. We pre-train the encoders to provide them with good initialization for end-to-end learning. First, we pre-train the *phrase encoder* in autoencoder format, and then keeping it fixed, we pre-train the *visual encoder* using a ranking loss. The loss enforces the cosine similarity $S_c(.)$ between the phrase-encoding and the visual-encoding for ground-truth pair $(\mathbf{p}_i, \mathbf{r}_j)$ to be more than that of a contrastive pair by least the margin γ:

$$\mathcal{L} = \sum (\mathbb{E}_{\widetilde{\mathbf{p}} \neq \mathbf{p}_i} max\{0, \gamma - S_C(\mathbf{p}_i, \mathbf{r}_j) + S_C(\widetilde{\mathbf{p}}, \mathbf{r}_j)\} + \mathbb{E}_{\widetilde{\mathbf{r}} \neq \mathbf{r}_j} max\{0, \gamma - S_C(\mathbf{p}_i, \mathbf{r}_j) + S_C(\mathbf{p}_i, \widetilde{\mathbf{r}})\}) \quad (9)$$

where $\widetilde{\mathbf{r}}$ and $\widetilde{\mathbf{p}}$ denote randomly sampled constrastive image region and phrase respectively. The *caption encoder* and the

Method	Accuracy
SMPL [30]	42.08
NonlinearSP [29]	43.89
GroundeR [26]	47.81
MCB [10]	48.69
RtP [24]	50.89
Similarity Network [28]	51.05
IGOP [38]	53.97
SPC+PPC [23]	55.49
SS+QRN (VGG_{det}) [6]	55.99
CITE [22]	59.27
SeqGROUND [9]	61.60
CITE [22] (finetuned)	61.89
QRC Net [6] (finetuned)	65.14
G^{3}RAPHGROUND++	**66.93**

Table 1. **State-of-the-art comparison on Flickr30k.** Phrase grounding accuracy on the test set reported in percentages.

Method	Accuracy
SCRC [12]	17.93
MCB + Reg + Spatial [5]	26.54
GroundeR + Spatial [26]	26.93
Similarity Network + Spatial [28]	31.26
CGRE [20]	31.85
MNN + Reg + Spatial [5]	32.21
EB+QRN (VGG_{cls}-SPAT) [6]	32.21
CITE [22]	34.13
IGOP [38]	34.70
QRC Net [6] (finetuned)	44.07
G^{3}RAPHGROUND++	**44.91**

Table 2. **State-of-the-art comparison on ReferIt Game.** Phrase grounding accuracy on the test set reported in percentages.

image encoder are pre-trained in similar fashion. After pre-training the encoders, we jointly train the model end-to-end.

For end-to-end training, we formulate this as a binary-classification task where the model predicts grounding decision for each phrase-region pair. We minimize binary cross-entropy loss $BCE(\cdot)$ between the model prediction and the ground-truth label. We also jointly train f_{count} and apply binary cross-entropy loss for the binary-classification task of predicting if a phrase should be grounded to a single region or multiple regions. The total training loss is described as:

$$\mathcal{L}_{train} = BCE(\hat{d}_{i,j}, d_{i,j}) + \lambda BCE(\hat{\beta}_i, \beta_i), \qquad (10)$$

where $\hat{d}_{i,j}$ and $d_{i,j}$ are the prediction and ground-truth grounding decision for i^{th} phrase and j^{th} region respectively, meanwhile, $\hat{\beta}_i$ and β_i are the prediction and ground-truth on whether i^{th} phrase is grounded to multiple regions or not; λ is a hyperparameter that is tuned using grid search.

4. Experiments

4.1. Setup and Inference

We use Faster R-CNN [25] with VGG-16 backbone as a mechanism for extracting proposal regions from the images. We treat those image regions (*i.e.*, bounding-boxes) proposed by RPN as positive labels during training which have IoU of more than 0.7 with the ground-truth boxes annotations of the dataset. For phrases where no such box exists, we reduce the threshold to 0.5. We sample three negative boxes for every positive during training. This ensures that the learned model is not biased towards negatives.

During inference, we feed all the proposal image regions to the model and make two predictions. The first prediction is for each phrase, to determine whether the phrase should be grounded to a single or multiple image regions. The second prediction is for each phrase-region pair, to determine the probability of grounding the given phrase to the given image region. Based on the first prediction, results of the second prediction are accordingly post processed, and the phrase is grounded to a single or multiple image regions.

4.2. Datasets and Evaluation

We validate our model on Flickr30k [24] and Referit Game [14] datasets. Flickr30k contains 31,783 images where each image is annotated with five captions/sentences. Each caption is further parsed into phrases, and the corresponding bounding-box annotations are available. A phrase may be annotated with more than one ground-truth bounding-box, and a bounding-box may be annotated to more than one phrase. We use the same dataset split as previous works [22, 24] which use 29,783 images for training, 1000 for validation, and 1000 for testing.

Referit Game dataset contains 20,000 images and we use same split as used in [12, 22] where we use 10,000 images for training and validation while other 10,000 for testing. Each image is annotated with multiple referring expressions (phrases) and corresponding bounding-boxes. We note that the phrases corresponding to a given image of this dataset do not come from a sentence but exist independently.

Consistent with the prior work [26], we use grounding accuracy as the evaluation metric which is the ratio of correctly grounded phrases to total number of phrases in the test set. If a phrase is grounded to multiple boxes, we first take the union of the predicted boxes over the image plane. The phrase is correctly grounded if the predicted region has IoU of more than 0.5 with the ground-truth.

4.3. Results and Comparison

Flickr30k. We test our model on Flickr30k dataset and report our results in Table 1. Our full model

Method	Flickr30k	ReferIt Game
GG - PhraseG	60.82	38.12
GG - VisualG	62.23	38.82
GG - FusionG	59.13	36.54
GG - VisualG - FusionG	56.32	32.89
GG - ImageContext	62.32	40.92
GG - CaptionContext	62.73	41.79
GGFusionBase	60.41	38.65
G^{3}RAPHGROUND (GG)	63.87	41.79
G^{3}RAPHGROUND++	**66.93**	**44.91**

Table 3. **Ablation results.** Flickr30k and ReferIt Game datasets.

G^{3}RAPHGROUND++ surpasses all other works by achieving the best accuracy of 66.93%. The model achieves 5.33% increase in the grounding accuracy over the state-of-the-art performance of SeqGROUND [9]. Most methods, as do we, do not finetune the features on the target dataset. Exceptions include CITE [22] and QRC Net [6] designated as (finetuned) in the table. We highlight that comparison to those methods isn't strictly fair as they use Flickr30k dataset itself to finetune feature extractors. Despite this, we outperform them, by 5% and 1.8% respectively, without utilizing specialized feature extractors. When compared to the versions of these models (CITE and SS+QRN (VGG$_{\text{det}}$)) that are not finetuned, our model outperform them by 7.7% and 10.9% respectively. This highlights the power of our contextual reasoning in G^{3}RAPHGROUND. Finetuning of features is likely to lead to additional improvements.

Table 4 shows the phrase grounding performance of the models for different coarse categories in Flickr30k dataset. we observe that G^{3}RAPHGROUND++ achieves consistent increase in accuracy compared to other methods in all of the categories except for the "instruments"; in fact our model performs best in six out of eight categories even when compared with the finetuned methods like [6, 22]. Improvement in the accuracy for "clothing" and "body parts" categories is more than 8% and 9% respectively.

We also consider a stricter metric for the box-level accuracy. We call the phrase correctly grounded if: 1) every box in the ground truth for the phrase has an IOU > 0.5 with at least one box among those that are matched to the phrase by the model; and 2) Every box among those matched to the phrase by the model has an IOU > 0.5 with at least one box from the ground truth for the phrase. We report this metric for phrases with single ($n = 1$) and multiple ($n > 1$) ground truth annotations below. We also consider *Top1* version of our model that grounds every phrase to one max score box.

Method	Acc ($n = 1$)	Acc ($n > 1$)	mean Acc
G^{3}RAPHGROUND (*Top1*)	**69.03**	4.80	56.12
G^{3}RAPHGROUND (GG)	53.17	**25.78**	48.08
G^{3}RAPHGROUND++	67.46	25.61	**59.07**

(a) A young boy is looking at a man painted in all gold.

(b) A man is checking his blue sneakers next to two men having a conversation.

(c) A brown dog jumps high on a field of grass.

(d) A woman stands in a field near a car and looks through binoculars.

Figure 3. **Sample attention results for *visual graph*.** Aggregated attention over each image region projected in an image.

ReferIt Game. We report results of our model on ReferIt Game dataset in Table 2. G^{3}RAPHGROUND++ clearly outperforms all other state-of-the-art techniques and achieves the best accuracy of 44.91%. Our model improves the grounding accuracy by 10.21% over the state-of-the-art IGOP [38] model that uses similar features.

4.4. Qualitative Results

In Figure 3 we visualize the attention (α) on the nodes (image regions) of the *visual graph* (image). We find that the model is able to differentiate the important image regions from the rest, for example, in (a), the model assigns higher attention weights to important foreground objects such as child and man than the background objects like wall and pillar. Similarly in (d), woman and car get more attention than any other region in the image.

We also visualize some phrase grounding results in Figure 4. We find that our model is successful in grounding phrases for challenging scenarios. In (f) the model is able to distinguish *two women* from *other women* and is also able to infer that *colorful clothing* corresponds to the dress of *two women* not *other women*. In (b), (d) and (f) our model is able to ground single phrase to multiple corresponding bounding-boxes. Also note correct grounding of *hand* in (i) despite the presence of other *hand* candidate. We also point out few mistakes, for example in (i), *blue Bic pen* is incorrectly grounded to a bracelet which is spatially close. In (h), *curly hair* is grounded to a larger bounding-box.

4.5. Ablation

We conduct ablation studies on our model to clearly understand the benefits of each component. Table 3 shows the results on both datasets. G^{3}RAPHGROUND++ is our full model which achieves the best accuracy. G^{3}RAPHGROUND lacks the separate count prediction branch, and therefore post processes all the predictions of the network using

Method	people	clothing	body parts	animals	vehicles	instruments	scene	other
SMPL [30]	57.89	34.61	15.87	55.98	52.25	23.46	34.22	26.23
GroundeR [26]	61.00	38.12	10.33	62.55	68.75	36.42	58.18	29.08
RtP [24]	64.73	46.88	17.21	65.83	68.72	37.65	51.39	31.77
IGOP [38]	68.71	56.83	19.50	70.07	73.75	39.50	60.38	32.45
SPC+PPC [23]	71.69	50.95	25.24	76.23	66.50	35.80	51.51	35.98
SeqGROUND [9]	76.02	56.94	26.18	75.56	66.00	39.36	68.69	40.60
CITE [22] (finetuned)	75.95	58.50	30.78	77.03	**79.25**	48.15	58.78	43.24
QRC Net [6] (finetuned)	76.32	59.58	25.24	80.50	78.25	**50.62**	67.12	43.60
G^{3}RAPHGROUND++	**78.86**	**68.34**	**39.80**	**81.38**	76.58	42.35	**68.82**	**45.08**

Table 4. **Phrase grounding accuracy comparison** over coarse categories on Flickr30k dataset.

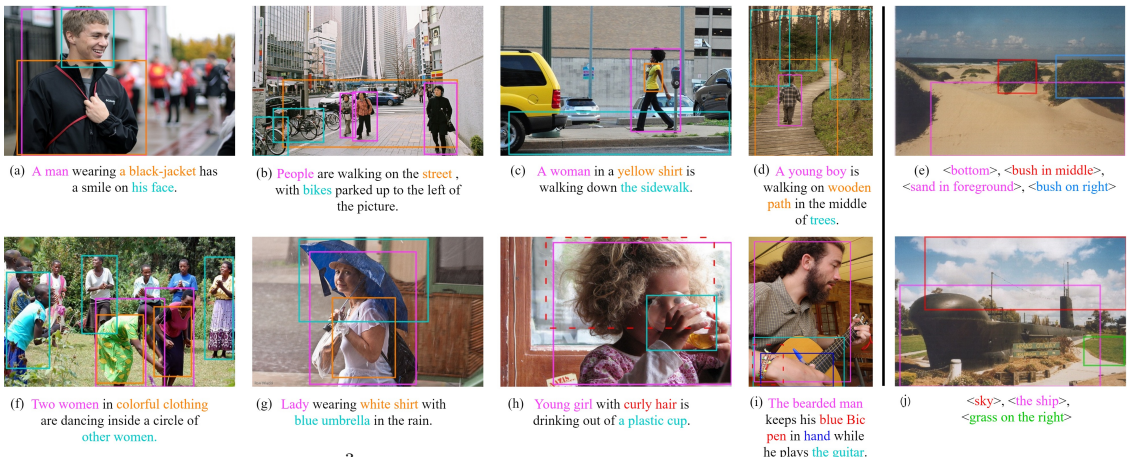

Figure 4. **Sample results obtained by G^{3}RAPHGROUND** .The colored bounding-boxes correspond to the phrases in same color.

the threshold mechanism. The model *GG-PhraseG* lacks the *phrase graph* to share information across the phrases, and directly uses the output of the *phrase encoder* during the fusion step. In a similar approach, the model *GG-VisualG* lacks the *visual graph*, *i.e.*, there occurs no message-passing among proposal image regions. The output of the *visual encoder* is directly used during the fusion. The model *GG-FusionG* lacks the *fusion graph*, *i.e.*, the *prediction network* makes the predictions directly from the output of the *visual graph* concatenated with the output of the *phrase graph*. *GG-VisualG-FusionG* is missing both the *visual graph* and the *fusion graph*. *GG-ImageContext* and *GG-CaptionContext* do not use the full image and caption embedding respectively in the context information. We design another strong baseline *GGFusionBase* for G^{3}RAPHGROUND to validate our *fusion graph*. In this method we do not instantiate one *fusion graph* on each phrase for conditional massage-passing, but instead perform fusion through message-passing on a single big graph that consists of the updated nodes of both, the *phrase graph* and the *visual graph*, such that each phrase node is connected to each image region node with an edge of unit weight; no edges between the nodes of the same modality exist.

We find that the results show consistent patterns in both of the datasets. The worse performance of *GG-PhraseG* and *GG-VisualG* as compared to G^{3}RAPHGROUND confirms the importance of capturing intra-modal relationships. *GG-VisualG-FusionG* performs worst for both of the datasets. Even when either one of the *visual graph* or the *fusion graph* is present, accuracy is significantly boosted. However, the *fusion graph* is the most critical individual component of our model as its absence causes the maximum drop in accuracy. *GGFusionBase* is slightly better than *GG-FusionG* but still significantly worse than G^{3}RAPHGROUND . This is strong proof of the efficacy of our *fusion graph*. The role of our post processing technique is also evident from the performance gap between G^{3}RAPHGROUND and G^{3}RAPHGROUND++. Since each ablated model performs significantly worse than the combined model, we conclude that each module is important.

Conclusion. In this paper, we proposed G^{3}RAPHGROUND framework that deploys GG-NNs to capture intra-modal and cross-modal relationships between the phrases and the image regions to perform the task of language grounding. G^{3}RAPHGROUND encodes the phrases into the *phrase graph* and image regions into the *visual graph* to finally fuse them into the *fusion graph* using conditional message-passing. This allows the model to jointly make predictions for all phrase-region pairs without making any assumption about the underlying structure of the data. The effectiveness of our approach is demonstrated on two benchmark datasets, with up to 10% improvement on state-of-the-art.

References

[1] Nayyer Aafaq, Syed Zulqarnain Gilani, Wei Liu, and Ajmal Mian. Video description: A survey of methods, datasets and evaluation metrics. *CoRR*, abs/1806.00186, 2018. 2

[2] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[3] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *European Conference on Computer Vision (ECCV)*, 2018. 2

[4] Lisa Beinborn, Teresa Botschen, and Iryna Gurevych. Multimodal grounding for language processing. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2325–2339. Association for Computational Linguistics, 2018. 2

[5] Kan Chen, Rama Kovvuri, Jiyang Gao, and Ram Nevatia. Msrc: Multimodal spatial regression with semantic context for phrase grounding. In *ACM on International Conference on Multimedia Retrieval*, pages 23–31, 2017. 6

[6] Kan Chen, Rama Kovvuri, and Ram Nevatia. Query-guided regression network with context policy for phrase grounding. In *IEEE International Conference on Computer Vision (ICCV)*, pages 824–832, 2017. 1, 2, 6, 7, 8

[7] Xinlei Chen, Li-Jia Li, Li Fei-Fei, and Abhinav Gupta. Iterative visual reasoning beyond convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[8] Chaorui Deng, Qi Wu, Qingyao Wu, Fuyuan Hu, Fan Lyu, and Mingkui Tan. Visual grounding via accumulated attention. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[9] Pelin Dogan, Leonid Sigal, and Markus Gross. Neural sequential phrase grounding (seqground). In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2, 6, 7, 8

[10] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016. 6

[11] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 1

[12] Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. Natural language object retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4555–4564, 2016. 6

[13] Andrej Karpathy, Armand Joulin, and Li Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1889–1897, 2014. 1, 2

[14] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 787–798, 2014. 1, 2, 3, 6

[15] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017. 2

[16] Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. In *Transactions of the Association for Computational Linguistics (TACL)*, 2015. 2

[17] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. In *International Conference on Learning Representations (ICLR)*, 2016. 2, 3, 4, 5

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755, 2014. 1

[19] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Neural baby talk. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[20] Ruotian Luo and Gregory Shakhnarovich. Comprehension-guided referring expressions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7102–7111, 2017. 1, 6

[21] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. 3

[22] Bryan Plummer, Paige Kordas, Hadi Kiapour, Shuai Zheng, Robinson Piramuthu, and Svetlana Lazebnik. Conditional image-text embedding networks. In *European Conference on Computer Vision (ECCV)*, pages 249–264, 2018. 1, 2, 6, 7, 8

[23] Bryan Plummer, Arun Mallya, Christopher Cervantes, Julia Hockenmaier, and Svetlana Lazebnik. Phrase localization and visual relationship detection with comprehensive image-language cues. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1928–1937, 2017. 2, 6, 8

[24] Bryan Plummer, Liwei Wang, Chris Cervantes, Juan Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2641–2649, 2015. 1, 2, 3, 6, 8

[25] 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)*, pages 91–99, 2015. 1, 6

[26] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision (ECCV)*, pages 817–834, 2016. 6, 8

[27] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 39(4), 2017. 1

[28] Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning two-branch neural networks for image-text matching tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 41(2):394–407, 2019. 2, 6

[29] Liwei Wang, Yin Li, and Svetlana Lazebnik. Learning deep structure-preserving image-text embeddings. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5005–5013, 2016. 1, 2, 6

[30] Mingzhe Wang, Mahmoud Azab, Noriyuki Kojima, Rada Mihalcea, and Jia Deng. Structured matching for phrase localization. In *European Conference on Computer Vision (ECCV)*, pages 696–711, 2016. 2, 6, 8

[31] Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1960–1968, 2019. 2

[32] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *European Conference on Computer Vision (ECCV)*, pages 399–417, 2018. 3

[33] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A comprehensive survey on graph neural networks. *CoRR*, abs/1901.00596, 2019. 2

[34] Fanyi Xiao, Leonid Sigal, and Yong Jae Lee. Weakly-supervised visual grounding of phrases with linguistic structures. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[35] H. Xu, K. He, B. Plummer, L. Sigal, S. Sclaroff, and K. Saenko. Multilevel language and vision integration for text-to-clip retrieval. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2019. 2

[36] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019. 2, 3

[37] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *European Conference on Computer Vision (ECCV)*, pages 670–685, 2018. 3

[38] Raymond Yeh, Jinjun Xiong, Wen-Mei Hwu, Minh Do, and Alexander Schwing. Interpretable and globally optimal prediction for textual grounding using image concepts. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1912–1922, 2017. 6, 7, 8

[39] Ying Zhang and Huchuan Lu. Deep cross-modal projection learning for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 686–701, 2018. 2

[40] Yundong Zhang, Juan Carlos Niebles, and Alvaro Soto. Interpretable visual question answering by visual grounding from attention supervision mining. 1

[41] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, and Maosong Sun. Graph neural networks: A review of methods and applications. *CoRR*, abs/1812.08434, 2018. 2