Dual Energy-Flow Enhanced Graph Neural Network for Visual Question Answering

Abstract

Scene graph, as a structural abstraction of natural images, contains massive, detailed information. Modeling visual reasoning through scene graphs can significantly improve the ability and strengthen the interpretability of reasoning. However, neither does one of these models *jointly* exploit objects, relations, and attributes information in scene graph, nor does one of them balance the importance of objects and relations. In this paper, we introduce a novel Dual Energy-Flow Enhanced Graph Neural Network (DE-GNN), which learns a comprehensive representation by encoding full-scale scene graph information from objects, attributes, and relations. Specifically, two types of scene graph structures are employed in the encoder: (i) *Object-significant graph* which embeds attribute and relation information into node representations. (ii) *Relation-significant graph* which intensifies the model perception of relation features. In addition, we design an *energy-flow mechanism* to enhance the information transferred from edges and adjacent nodes to updating nodes. We conduct extensive experiments on public GQA and Visual Genome datasets and achieve new state-of-the-art performances highlighting our method's benefits.

Introduction

Visual Question Answering (VQA) tasks require a model to answer a free-form natural language question using visual information from an image. Scene graph (SG) reasoning is an essential instance of VQA tasks (Hildebrandt et al. 2020). The model extracts objects' names, attributes, and relationships from the input images and organizes them into a graph representation to generate the scene graph.

SG representation modeling displays several virtues over classical techniques leveraging object features extracted from images since (a) the features in SG are presented in plain and free text form (Damodaran et al. 2021) and (b) the graph structures of SG, which have better interpretability (Zhang, Chao, and Xuan 2019). In this contribution, two reasoning methods on scene graphs are proposed. In particular, we: (i) consider scene graphs as probabilistic graphs and iteratively update nodes' probabilities using soft instructions extracted from questions such as Neural State Machine (NSM) (Hudson and Manning 2019b; Le et al. 2020); (ii) apply Graph Neural Network (GNN) into scene graphs (Singh et al. 2019; Li et al. 2019) to learn a joint representation of the nodes and their relations, and then feed this representation into a predictor to generate the answer.

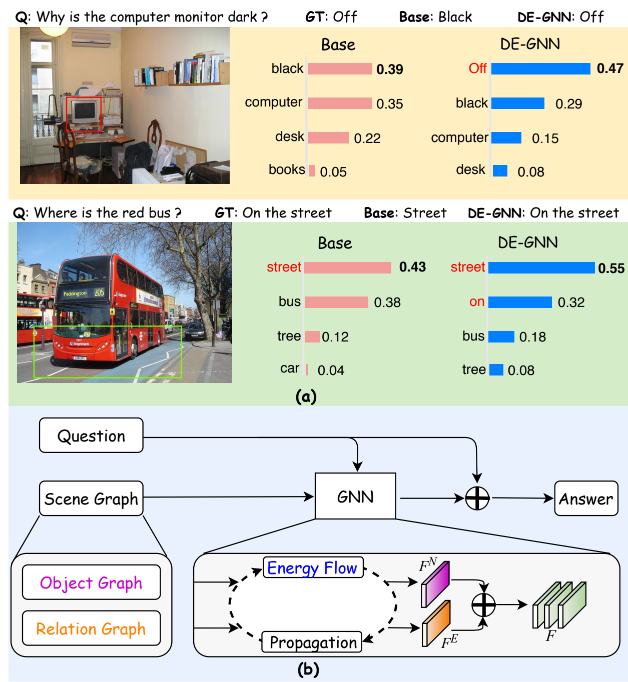


Figure 1: (a) Two key issues of traditional scene graph based models that we address: **false attribute selection** (bottom: select attribute "black" instead of attribute "off") and **missing relation** (top: missing "on the") (b) Overview of our DE-GNN model. F^N and F^E are the node feature map and the relation feature map. F is the full-scale feature map.

Scene graph reasoning frameworks have proven to be useful in VQA tasks, e.g. (Johnson et al. 2015; Yang et al. 2020). However, there still remains some imperfections.

First, models tend to answer wrong for complex reasoning questions. Consider the "why" question in Fig. 1(a) as an example, false attribute selection occurs because the model cannot associate "off" relation with "dark monitor" object. This is because attribute selections require supervision from objects, relations, and attributes, but existing methods fail at generating *joint* representations for objects by using features from their neighbors *and* their attributes. Generally, information from objects and relations connected to them are reconstructed into object features in GNN-based methods (Xu et al. 2019). However, these encoding methods lack information from objects' attributes and neighbor objects. The NSM methods use attention mechanisms to update answer possibilities of objects, attributes, and relations, but they cannot learn the joint representation of all three types of information.

Second, models answer poorly for questions that require information about the diverse relations and objects. For instance, for localization questions, as in the "where" type of questions, in Fig. 1(a), we observe that relation missing occurs because the model cannot capture the relation information "on the". This is because existing models have a strong bias towards node features, considering edge features as references. Generally in these models, nodes refer to objects, and edges refer to relations, which lead to the unbalance focus on objects and relations.

Therefore, as a fix to the current ineffective strategies, we propose the Dual Energy-Flow Enhanced Graph Neural Network (DE-GNN) for VQA, introducing a novel scene graph reasoning model that extracts balanced feature maps from objects, attributes, and relations information in scene graphs. Concretely, as shown Fig. 1(b), our DE-GNN model is composed of a scene graph generator, a question encoder, dual graph encoders, and a fusion module. Essentially, the scene graph generator extracts graphs out of the input images. Besides, to balance the importance of objects and relations, we transform scene graphs into a relation-significant modality, where nodes represent relations and edges represent objects, and an object-significant modality, in which nodes represent objects and edges represent relations. After receiving scene graphs in two modalities, dual graph encoders can produce feature maps focusing on relations and objects.

Furthermore, to learn a node's joint representation from its attributes, edges, and adjacent nodes, we modify the gated graph neural network (GGNN) structure in our proposed DE-GNN by adding the energy-flow module. It is a bidirectional GRU that guides the internal information flow. The encoder captures information from nodes, edges, and adjacent nodes that connect to them. As shown in Fig. 1(b), the output feature map of the encoder passes through multi-head attention layers using question features extracted from the question encoder. Hence, the model dynamically focuses on the critical parts of the questions and uses the most similar part of the scene graph as the most adequate answer.

In summary, our main contributions are as follows:

- We propose a novel DE-GNN model to learn a comprehensive and balanced representation of scene graphs by encoding graphs' object-significant modality and relation-significant modality.
- Our energy-flow module is suitable for processing graphs with meaningful edges and nodes with attributes.
- We conduct experiments on GQA and Visual Genome datasets. Experimental results demonstrate that DE-GNN effectively improves the reasoning accuracy on semantically complicated questions.

Related Work

Visual Question Answering. Most VQA approaches utilize a sequential model to encode the question and employ CNN-based pretrained models like Mask-RCNN or Faster-RCNN (Patro and Namboodiri 2018; Nam, Ha, and Kim 2017) to encode the image. The image encoder and question encoder then pass through a multimodal fusion part and the output fusion vector pass through an answer predictor. Many attention-based models (Anderson et al. 2018; Fan and Zhou 2018; Xu and Saenko 2016; Lu et al. 2016; Hudson and Manning 2018) are proposed to model the relations between the images and the questions. Transformer-based models such as Unicoder-VL (Li et al. 2020) can achieve outstanding performances on VQA tasks, yet these models are heavy to apply due to complicated pretraining strategies, extra datasets, time-consuming training, hard to explain and hard to update under the changeable environment. Instead, scene graph based models stand for an alternative that is more lightly and explainable.

Scene Graph Generation and Reasoning. Most scene graph generation (SGG) methods use object detection methods like mask-rcnn or faster-rcnn to extract region proposals from images (Xu et al. 2017; Yang et al. 2018; Zellers et al. 2018; Woo et al. 2018; Dai, Zhang, and Lin 2017; Li et al. 2017; Yin et al. 2018; Tang et al. 2020). Scene graph can promote explainable reasoning for downstream multimodal tasks such as VQA (Zhang, Chao, and Xuan 2019). In our work, scene graph generation methods are used to transform VQA datasets into scene graph datasets. Our model is tested on those generated datasets using various SGG methods.

In typical scene graph reasoning models, neural state machine (Hudson and Manning 2019b) performs sequential reasoning over the scene graph by iteratively traversing its nodes to answer a given question. FSTT (Singh et al. 2019) uses GGNN based model to encode scene graphs. Relation-aware Graph Attention Network (Li et al. 2019) models multitype inter-object relations via a graph attention mechanism. However, the previous works are hard to fully utilize the attribute information and learn the comprehensive representation of scene graphs.

Graph Neural Network. A group of graph neural networks (GNN) (Scarselli et al. 2009; Wang et al. 2016, 2018; Sun and Li 2019; Morris et al. 2019; Liu et al. 2019) were proposed for different graph tasks. Graph convolutional network (GCN) (Kipf and Welling 2017) improves GNNs efficiency with fast approximated spectral operations. GAT (Velickovic et al. 2018) introduces the attention mechanism to GNN, leveraging masked self-attentional layers to address the shortcomings of prior methods based on graph convolutions or their approximations. GGNN (Li et al. 2016) uses gated recurrent units (GRU) to accelerate the training speed and gain favorable inductive biases on large-scaled graphs. Our DE-GNN model can learn a comprehensive representation using full-scale scene graph information from objects, attributes, and relations to overcome these problems.

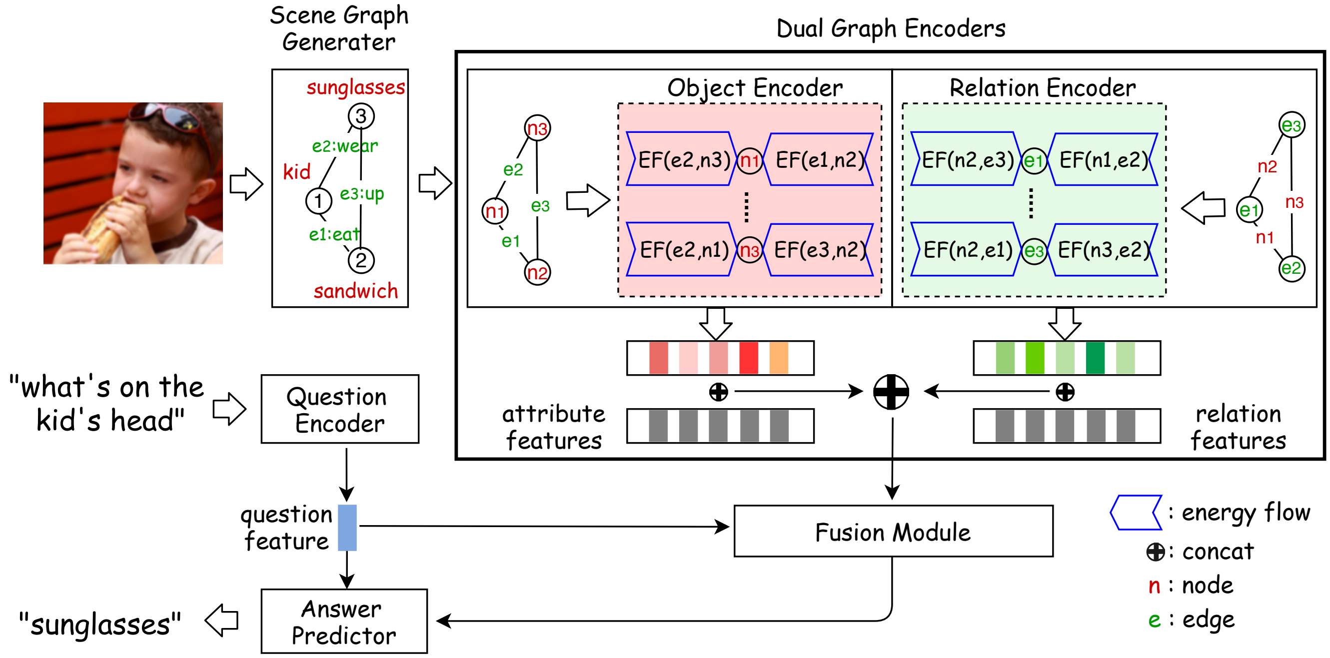


Figure 2: Model structure of the Dual Energy-Flow enhanced Graph Neural Networks. EF stands for the energy-flow module. Images are transformed into scene graphs by the scene graph generator. The object-significant form and relation-significant form of the scene graph are injected into the object encoder and the relation encoder. Nodes' representations are generated from the sum of the energy-flow modules. The representations are then fused with question representations to predict answers.

DE-GNN Methodology

Beforehand, we define the VQA task. It is a classification task where the goal is to output the correct answer for a given text question about an image. Formally, given question *q* and image *m*, the model aims to maximize a conditional distribution over candidate answers *a* as follows:

$$\hat{a} = \arg\max_{a \in A} p_\theta(a|q, m) \tag{1}$$

where *A* is the set of all possible answers, p_θ represents the VQA model with the trainable vector of parameters θ and $\hat{a}$ denotes the final answer.

Our proposed architecture designed for the VQA task is illustrated in Fig. 2. Our model contains a scene graph generator, a question encoder, dual graph encoders, and a fusion module. For the scene graph generator, we follow a codebase (Tang 2020) and other baselines referred in this work, which we will describe in the experiment section. For the question encoder, semantic questions are first projected into an embedding space using GLOVE pretrained word embedding model (Pennington, Socher, and Manning 2014). After adding a positional encoding matrix into questions, we use long short-term memory (LSTM) networks to generate questions embedding $q \in R^{dim}$. We introduce our dual GGNN encoders in the following subsection.

Object/Relation-Significant Graph

We organize scene graphs into object-significant and relation-significant modalities.

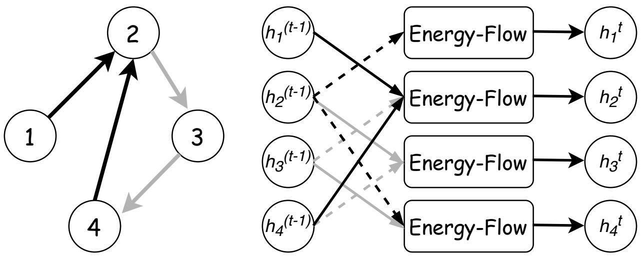
Object-Significant Graph. We define the object significant modality as G_{obj}, where each node represents an object in the image and each edge represents a relation between two objects. Define *N* as the node set and *E* as the edge set. For $n_i, n_j \in N$, $e_k \in E$, $< n_i$ - e_k - $n_j >$ denotes the relation tuple that represents the relation e_k from object n_i to object n_j. Note that relation tuples are not symmetrical: if $< n_i - e_k - n_j >$ is a valid relation tuple, $< n_j - e_k - n_i >$ may not exist. Also, n_i and n_j may have several relations.

Relation-Significant Graph. We define relation significant modality as G_{rel}, where each node represents a relation between objects in the image and each edge represents an object, which is completely opposed to the object-significant modality. For $e_i, e_j \in E$, $n_k \in N$, $< e_i - n_k - e_j >$ denotes the relation tuple that represents the relations e_i and e_j have a shared object n_k. Note that relation tuples are also not symmetrical.

Attribute types. Define *L* as attribute types (such as material, color, etc). For each node $n_i \in N$ that corresponds to an object in the image, we define a set of $L+1$ property variables $\left\{n_i^j\right\}_{l=0}^{L}$, where n_i^0 represents n_i's name embedding and n_i^l represents the embedding of node n_i's l^{th} attribute (such as wooden, blue, etc).

Dual Encoders

We use two GGNN models as the encoders for object significant graph and relation significant graph. The GGNN for



(a) (b)

Figure 3: Overview of the energy-flow module. (a) A graph example, where color denotes edge types. (b) Unrolled one timestep. Solid and dotted lines are edges' forward and backward representations.

object graph focuses on object features and the GGNN for relation graph focuses on relation features. The dual encoder combination can balance the importance of relations and objects. Prior to encoding, every input scene graph is transformed into an information tuple (N, E, A_{in}, A_{out}):

- N is a collection of node embeddings.
- E is a collection of directed edges that specify valid relation between nodes.
- A_{in} is the adjacency matrix of incident edges.
- A_{out} is the adjacency matrix of output edges.

Let h_i^t is the hidden state of node n_i in GGNN at timestep t, then at $t = 0$, we initialize h_i^0 as the GLOVE embedding of n_i with appropriate zero padding:

$$h_i^0 = [n_i^T, 0]^T \,. \tag{2}$$

The incident and output edges are retrieved in the respective adjacency matrices A_{in} and A_{out}.

Energy-Flow Module To enhance the information transfer from edges and adjacent nodes to the updating nodes, we use the Energy-Flow module (EF) in Fig. 3. EF module comes as a replacement of the fully-connected layers from the original GGNN model. Consider a tuple $< n_i, e_k, n_j >$ as the processing sample of the energy-flow module. The embedding state, noted e_k, of the edge e_k and neighbor node n_j's hidden state h_j are injected into a bidirectional GRU network as input sequence while the node n_i's hidden state h_i is injected as the GRU's initial hidden state. The output of the GRU represents the updating information for hidden state h_i, which corresponds to the key information from edge e_k and node n_j that is related to node n_i. The sum of every GRU output is n_i's total information gain from n_i's adjacent nodes and edges. We detail the complete energy-flow module formula as follows:

$$EF_i(A_{in}) = \sum_{k,j}^{<n_i, e_k, n_j> \in A_{in}} \mathrm{GRU}([e_k, h_j], h_i) \,, \tag{3}$$

$$EF_i(A_{out}) = \sum_{k,j}^{<n_j, e_k, n_i> \in A_{out}} \mathrm{GRU}([e_k, h_j], h_i) \,, \tag{4}$$

where $EF_i(A_{in})$ is n_i's incident information gain, and $EF_i(A_{out})$ is n_i's output information gain.

Propagation Model At timestep t, the hidden states of all nodes are updated by the following gated propagator module:

$$k_i^t = [EF_i^t(A_{in}), EF_i^t(A_{out})] \,, \tag{5}$$

where k_i^t is the node n_i's representation from all its incident edges, output edges and adjacent nodes.

Then, we adopt GRU-like updates to incorporate information from adjacent nodes and from the previous timestep leading to an update of each node's hidden state:

$$c_i^t = [h_i^{(t-1)}, k_i^{(t-1)}]W + b \,, \tag{6}$$

$$z_i^t = \sigma(U^z c_i^t) \,, \tag{7}$$

$$r_i^t = \sigma(U^r c_i^t) \,, \tag{8}$$

where W, U^z and U^r are referred to as the trainable weight matrices and b as a bias term. At timestep t, we denote by z_i^t and r_i^t the update and reset gates, respectively. Then we have:

$$\tilde{h}_i^t = \tanh(U_1 k_i^{(t-1)} + U_2(r_i^t \odot h_i^{(t-1)})) \,, \tag{9}$$

$$h_i^t = (1 - z_i^t) \odot h_i^{(t-1)} + z_i^t \odot \tilde{h}_i^t \,. \tag{10}$$

Here, U_1 and U_2 denote the trainable parameters of the linear layers, the operator $\odot$ is the element-wise multiplication. After T steps, the GGNN encoder generates the final hidden state map G of the graph. Finally, we compute the graph embedding $g_i \in G$ for node n_i as follows:

$$g_i = \sigma(f(h_i^T, n_i)) \,, \tag{11}$$

where $f(h_i^T, n_i)$ is the multi-layer perceptron (MLP) which receives the concatenation of h_i^T and n_i, then generates the final representation of node n_i.

Fusion Module and Answer Predictor

Once the dual encoders, embedded in our model, output the node and relation features, we first fuse the attributes into feature maps. For node feature map G^N and relation feature map G^E, the fusion feature map F^N and F^E are defined as

$$F_i^N = \begin{cases} [g_i^N, n_i^0] \\ \cdots, \\ [g_i^N, n_i^L] \end{cases}, \; F_j^E = [g_j^E, e_j], \; F = [F^N, F^E], \tag{12}$$

where F_i^N indicates the fusion features of node i and g_i^N is node i's representation from the GGNN encoder. We denote by the vector $(n_i^0, \cdots, n_i^L)$ the embeddings attributes of node i. F_j^E corresponds to the fusion feature of edge j. g_j^E is edge j's representation from the GGNN encoder. e_j is j-th edge original embedding. The full-scale feature map, noted F, is obtained by concatenating F^N and F^E.

Then, the question embedding q generated from the LSTM encoder and the full-scale feature map F are fed into a multi-head attention layer, where the query is stored in F and the key and values are stored in q. The reasoning vector, noted r, and which stems from the graph and the question, is computed using a weighted sum of the feature map using the scores output from the attention layer, i.e.,

$$r = \text{Attention}(F, q)\,. \tag{13}$$

Regarding the answer predictor module, we adopt a two-layer MLP noted by $f(\cdot)$. This MLP can be viewed as a classifier over the set of candidate answers. The input of the answer predictor is the concatenation vector (q, r). Such a classifier has been applied in many VQA models as in NSM (Hudson and Manning 2019b) and MacNet (Lu et al. 2016). Formally, the output answer reads:

$$\hat{a} = \arg\max(\text{softmax}(f((q, r))))\,. \tag{14}$$

We provide in the next section, various numerical experiments to validate our newly introduced model.

Experiments

Datasets

– The **Visual Genome** dataset contains 108 077 images with comprehensively annotated objects, attributes, and relations. To enrich the scene graph annotation in Visual Genome, we use a scene graph generation method and motifs (Zellers et al. 2018) to generate a new scene graph dataset called **motif-VG**. Compared with the Visual Genome dataset, motif-VG has the same images and questions-answers tuples, but has scene graph annotations with different qualities and biases.
– The **GQA** dataset (Hudson and Manning 2019a) focuses on real-world reasoning, scene understanding and compositional question answering. It comprises 113k images and 22M questions of assorted types and varying compositionality degrees, measuring performance on an array of reasoning skills such as object and attribute recognition, transitive relation tracking, spatial reasoning, logical inference and comparisons.

Implementation Details

We use the 50-dimensional GLOVE word embeddings model (Pennington, Socher, and Manning 2014) to embed scene graphs and questions. In order to record the questions' position information, we set up the positional encoding matrix PE as follows:

$$\text{PE}_{\texttt{pos}=2i} = \sin(\texttt{pos}/10000^{2i/d_\text{m}})\,, \tag{15}$$

$$\text{PE}_{\texttt{pos}=2i+1} = \cos(\texttt{pos}/10000^{2i/d_\text{m}})\,, \tag{16}$$

where $\texttt{pos}$ is the position of the word in the question sequence. The question embeddings are injected into a single-directional GRU network. The dimension of the hidden layers of the GRU is 100, and the dropout rate is 0.2.

In the GGNN encoder, the propagator time step is 5. We set the dimension of energy-flow's GRU hidden layer to 50.

In the fusion module, we apply a multi-head attention layer with five heads and no dropout. We select the top-2000 answer candidates and use a 2-layer MLP as the output classifier regarding the answer predictor.

We use Adam (Kingma and Ba 2015) as the optimizer, and Cross Entropy Loss as the loss function during the training of our model. More details are presented in supplementary materials.

Empirical Results

In this subsection, we provide the experimental results on various datasets mentioned above. The different baselines compared in our experiments all use different methods to generate the scene graphs for images. In order to ensure general fairness across the baselines, we implement them from scratch, removing their scene graph generation parts to eliminate the interference of different generation methods.

Results on VG dataset Table 1 reports the results on the test sets of the VG ground truth dataset and the motif-VG dataset. Compared to the baseline models, we can observe that our DE-GNN model outperforms the others at **3%-4%**. In addition, we provide detailed results on the VG dataset and motif-VG dataset with different question types. Compared to the other scene graph based VQA models, our model performs well in "what", "where", "who" and "why" types. On the VG dataset, our model has **6.5%** accuracy improvement in "**why**" type questions, which highly requires VQA models' ability to jointly exploit objects, relations and attributes. However, our model's comprehensive representations influence the accuracy on simple questions. In "color" type, our model achieves 2nd score.

Results on GQA dataset We report in Table 3 the detailed results on the test sets of the GQA dataset. Compared to the baseline models, our DE-GNN model achieves state-of-the-art accuracy performance. We also evaluate our model and other baselines across GQA dataset's various metrics, where "Binary" represents binary-answer questions, "Open" stands for open domain questions and "Distribution" corresponds to the distance between prediction distribution and standard answer distribution. For open domain questions which are difficult for reasoning, our model outperforms the others almost by **10%**. For distribution metric, our model also achieves 2nd score compared to other baselines. However, the comprehensive representation may interfere with DE-GNN's judgment of simple problems such as "Binary" questions.

Error Rate Analysis To demonstrate that our dual encoders structure can intensify the model's perception of relation features and learn a comprehensive representation from nodes, attributes, and relations information, we establish an error rate analysis for baselines and DE-GNN on motif-VG.

The badcases are classified into object, relation and attribute. We present in Table 4 the results for our error rate analysis. Our DE-GNN model surpasses all baselines in the terms of objects detection and our model does well in relation retrieval, outperforming GNN based FSTT and Re-GAT. This proves our model does alleviate the unbalance focus on objects and relations. Also, our model reduces nearly half of the wrong answers in FSTT, Re-GAT in the attribute aspect, which greatly improves the false attribute selection phenomenon.

Ablation Study

We compare several ablated forms of DE-GNN with our complete one. The accuracy results for each variant of DE-GNN are reported in Table 2. We use the original GGNN

Question type	**What**	**Color**	**Where**	**How**	**Who**	**When**	**Why**	**Overall**
Percentage	(54%)	(14%)	(17%)	(3%)	(5%)	(4%)	(3%)	(100%)
				VG-GroundTruth				
NSM (Hudson and Manning 2019b)	33.1	52.4	51.0	52.9	49.8	77.9	12.3	45.1
MLP (Jabri, Joulin, and van der Maaten 2016)	-	-	-	-	-	-	-	58.5
F-GN (Zhang, Chao, and Xuan 2019)	60.9	53.6	62.0	46.2	63.3	**83.7**	50.9	60.1
U-GN (Zhang, Chao, and Xuan 2019)	61.6	54.0	62.4	45.9	63.9	83.2	50.3	60.5
SAN (Fan and Zhou 2018)	-	-	-	-	-	-	-	62.6
FSTT (Singh et al. 2019)	65.5	45.6	70.1	47.8	68.3	82.1	91.5	65.6
ReGAT (Li et al. 2019)	72.1	**70.8**	64.4	**68.9**	72.7	65.0	92.3	71.2
DE-GNN (ours)	**75.9**	64.9	**73.1**	66.8	**82.6**	81.4	**98.8**	**75.4**
			Motif-VG					
NSM (Hudson and Manning 2019b)	31.8	62.4	53.1	51.4	47.6	83.3	10.9	43.1
FSTT (Singh et al. 2019)	48.8	40.4	49.2	40.1	40.6	54.5	70.3	48.1
F-GN (Zhang, Chao, and Xuan 2019)	58.7	60.8	60.4	47.2	61.8	84.8	49.0	60.0
U-GNN (Zhang, Chao, and Xuan 2019)	59.4	58.2	60.3	54.3	66.6	**85.3**	48.1	60.5
ReGATT (Li et al. 2019)	75.4	**69.2**	57.6	**69.9**	69.1	57.4	91.8	69.9
DE-GNN (ours)	**79.4**	67.6	**62.7**	65.3	**72.8**	63.0	**96.1**	**72.9**

Table 1: Performance on different question types of VG dataset.

Models	**Acc.**
Base	35.4%
+EF	*unstable*
+Obj	35.4%
+Obj+EF	39.3%
+Rel	35.2%
+Rel+EF	38.8%
+Obj+Rel	**67.9**%
+ (w/o *QF*)	54.9%
+ (w/o *attr*)	71.6%
+ (w/o *rela*)	**74.5**%
DE-GNN(ours)	**75.2**%

Table 2: Ablation study.

Models	**Binary**$\uparrow$	**Open**$\uparrow$	**Validity**$\uparrow$	**Distribution**$\downarrow$	**Accuracy**$\uparrow$
Human	91.20	87.40	98.90	-	89.30
BottomUp	66.64	34.83	96.18	5.98	49.74
MAC	71.23	38.91	96.16	5.34	54.06
SK T-Brain	77.42	43.10	96.26	7.54	59.19
PVR	77.69	43.01	**96.45**	5.80	59.27
GRN	77.53	43.35	96.18	6.06	59.37
Dream	77.84	43.72	96.38	8.40	59.72
LXRT	77.76	44.97	96.30	8.31	60.34
NSM	78.94	49.25	96.41	**3.71**	63.17
ReGAT	**83.57**	62.58	92.70	9.32	70.50
DE-GNN(ours)	69.79	**72.21**	93.80	3.78	**71.21**

Table 3: Performance on the GQA dataset.

Models	FSTT	ReGAT	DE-GNN	Case Number
Relation	45.9%	44.6%	**32.9**%	7913
Object	58.6%	47.3%	**25.2**%	4414
Attribute	49.6%	22.8%	**16.8**%	15483

Table 4: Error rate analysis on motif-VG dataset.

network as the *base* model to encode scene graphs. In Table 2, the *Obj* model represents the original GGNN network processing *object-significant* graph. The *Rel* model represents the original GGNN network processing *relation-significant* graph. The *Obj-EF* model corresponds to the object encoder part of our DE-GNN model, which contains one energy-flow enhanced GGNN network to encode *object-significant* graph. The *Rel-EF* model is the other half of our DE-GNN model, which contains one energy-flow enhanced GGNN network to encode *relation-significant* graph.

Effect of dual encoder structure We first validate the efficacy of applying dual structure to balance the importance of relations and objects by splitting our DE-GNN into an object-single model (noted *Obj*) and a relation-single model (noted *Rel*). Table 2 shows that both *Obj* and *Rel* models perform poorly, at $35.3\% \pm 0.1\%$. It also shows that both relations and objects are vital to VQA performance. Absence of any of those modules leads to severe accuracy recession. Combining the object-single model and the relation-single model leads to an empirical gain of 32.6% accuracy upward, which shows that the dual structure is significant in balancing relation and object information.

Effect of the energy-flow module We corroborate the effectiveness of applying energy-flow structure to learn a more comprehensive representation for scene graphs than the original GGNN structure, which represents the baseline in Table 2. We compare the *Obj+EF* model and the *Obj* model, which both learn representations from object-significant graphs, and note that after adding the energy-flow structure, there is an accuracy improvement of 3.9%. We also compare the *Rel+EF* model and the *Rel* model and observe an accuracy improvement of 3.6%. The *Obj+Rel* model is DE-GNN without energy-flow module. Comparing with DE-GNN, there is a 7.3% accuracy improvement

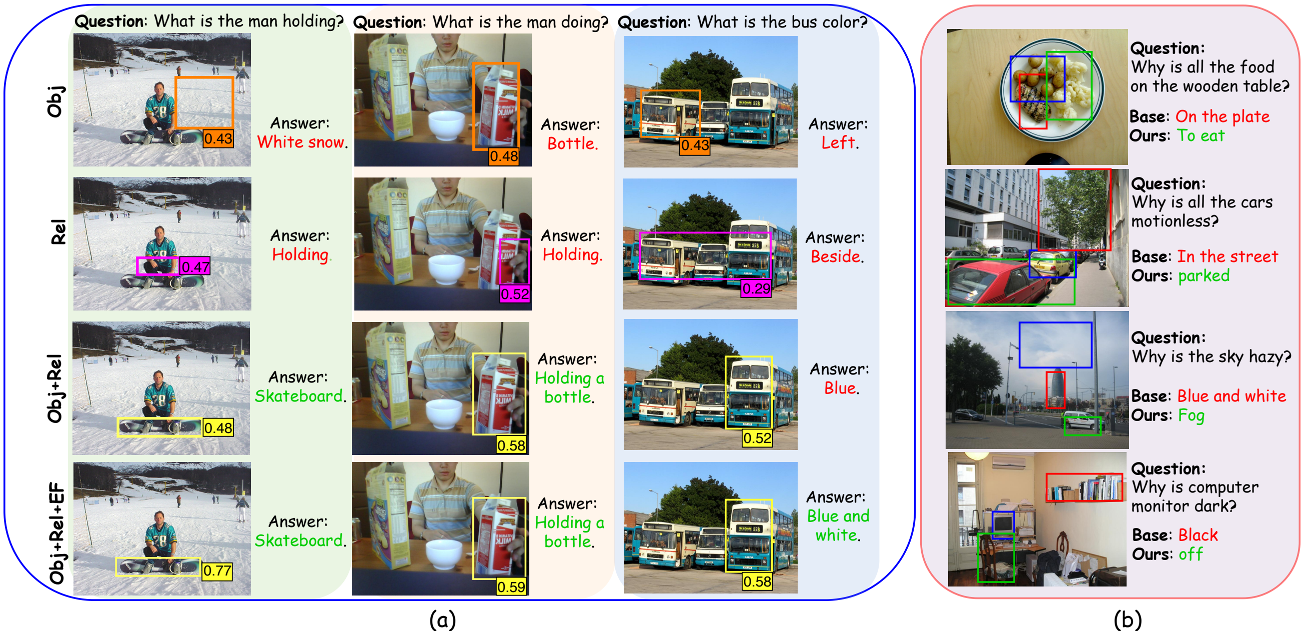


Figure 4: Examples of generated answers and top attention scores. Answers in red means wrong and answers in green means right. The left plot and right plot show the visualization of "what" and "why" question types. From the left plot (a), we observe that our approach can correctly select answer from objects, relations and attributes. From the right plot (b), we note that our model can handle comprehensive reasoning questions.

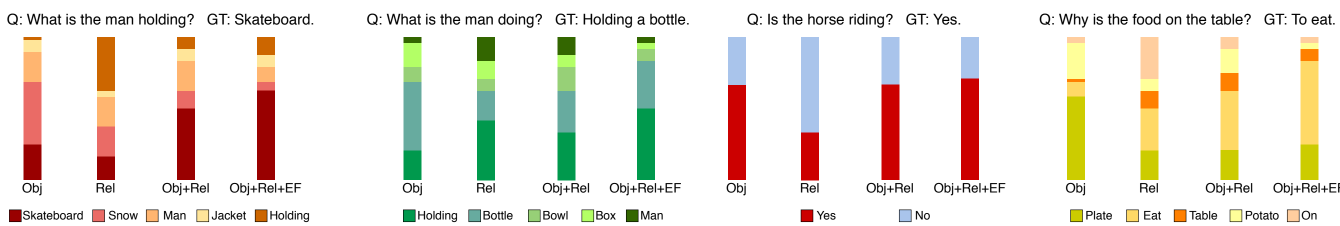


Figure 5: Examples of top 5 attention scores of candidate answers for different types of questions.

after adding the energy-flow module. These empirical results show that energy-flow structure can successfully improve the representation quality of scene graphs.

Effect of explicit modeling To demonstrate the impact of the explicit modeling of attributes and relations in DE-GNN, the *w/o attr* model removes the explicit attribute modeling part from DE-GNN and the *w/o rela* model removes the explicit relation modeling part. Removing attribute modeling leads to a drop of 3.6% in accuracy, while the relation modeling has only a 0.7% accuracy positive influence.

Visualization

To illustrate the effectiveness of the dual encoder structure and the energy-flow module in our DE-GNN model, we compare the top attention score learned by DE-GNN model with the scores learned by our *Obj*, *Rel* and *Obj+Rel* models. Fig. 4 provides a visualization of our results.

The left blue plot shows the visualization on "what" question type. Examples on all three columns of Fig. 4(a) correspond to "what" type of questions aimed at retrieving either object, relation or attribute information. Comparing row 1, row 2 with row 3, *Obj* and *Rel* models have strong attention bias toward objects and relations, while their combination, i.e. *Obj+Rel*, balances the attention on both sides and captures correct answers. Comparing row 3 with row 4, the addition of the energy-flow module increases the attention score of correct answers.

The right red plot shows the visualization on "why" type of question. For "why" questions, models jointly exploit objects, relations and attributes to generate answers. Using dual encoder structures ans the energy-flow module, our DE-GNN can generate correct answers and achieves **96.1%** accuracy on "why" questions.

Fig. 5 exhibits the details of top 5 attention scores for different types of questions. Our dual encoder structures and energy-flow module significantly increase the attention scores of correct answers.

Conclusion

In this work, we propose the DE-GNN model, which encodes each scene graph into feature representations via an object encoder and a relation encoder generating full-scale feature maps using objects, attributes, and relations information, and demonstrate our model can effectively boost performances on various datasets. In the future, we will dynamically adjust the information of SG for the model according to the difficulty of the questions which can improve our model's accuracy on simple questions.

References

Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; and Zhang, L. 2018. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 6077–6086. Computer Vision Foundation / IEEE Computer Society.

Dai, B.; Zhang, Y.; and Lin, D. 2017. Detecting Visual Relationships with Deep Relational Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 3298–3308. IEEE Computer Society.

Damodaran, V.; Chakravarthy, S.; Kumar, A.; Umapathy, A.; Mitamura, T.; Nakashima, Y.; Garcia, N.; and Chu, C. 2021. Understanding the Role of Scene Graphs in Visual Question Answering. *CoRR*, abs/2101.05479.

Fan, H.; and Zhou, J. 2018. Stacked Latent Attention for Multimodal Reasoning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 1072–1080.

Hildebrandt, M.; Li, H.; Koner, R.; Tresp, V.; and Günnemann, S. 2020. Scene Graph Reasoning for Visual Question Answering. *CoRR*, abs/2007.01072.

Hudson, D. A.; and Manning, C. D. 2018. Compositional Attention Networks for Machine Reasoning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

Hudson, D. A.; and Manning, C. D. 2019a. GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 6700–6709.

Hudson, D. A.; and Manning, C. D. 2019b. Learning by Abstraction: The Neural State Machine. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 5901–5914.

Jabri, A.; Joulin, A.; and van der Maaten, L. 2016. Revisiting Visual Question Answering Baselines. In Leibe, B.; Matas, J.; Sebe, N.; and Welling, M., eds., *Computer Vision - ECCV 2016*.

Johnson, J.; Krishna, R.; Stark, M.; Li, L.-J.; Shamma, D.; Bernstein, M.; and Fei-Fei, L. 2015. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3668–3678.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In Bengio, Y.; and LeCun, Y., eds., *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Le, T. M.; Le, V.; Venkatesh, S.; and Tran, T. 2020. Neural Reasoning, Fast and Slow, for Video Question Answering. In *2020 International Joint Conference on Neural Networks, IJCNN 2020, Glasgow, United Kingdom, July 19-24, 2020*, 1–8.

Li, G.; Duan, N.; Fang, Y.; Gong, M.; and Jiang, D. 2020. Unicoder-VL: A Universal Encoder for Vision and Language by Cross-Modal Pre-Training. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, 11336–11344.

Li, L.; Gan, Z.; Cheng, Y.; and Liu, J. 2019. Relation-Aware Graph Attention Network for Visual Question Answering. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, 10312–10321.

Li, Y.; Ouyang, W.; Zhou, B.; Wang, K.; and Wang, X. 2017. Scene Graph Generation from Objects, Phrases and Region Captions. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 1270–1279.

Li, Y.; Tarlow, D.; Brockschmidt, M.; and Zemel, R. S. 2016. Gated Graph Sequence Neural Networks. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

Liu, Z.; Chen, C.; Li, L.; Zhou, J.; Li, X.; Song, L.; and Qi, Y. 2019. GeniePath: Graph Neural Networks with Adaptive Receptive Paths. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, 4424–4431.

Lu, J.; Yang, J.; Batra, D.; and Parikh, D. 2016. Hierarchical Question-Image Co-Attention for Visual Question Answering. In Lee, D. D.; Sugiyama, M.; von Luxburg, U.; Guyon, I.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, 289–297.

Morris, C.; Ritzert, M.; Fey, M.; Hamilton, W. L.; Lenssen, J. E.; Rattan, G.; and Grohe, M. 2019. Weisfeiler and Leman Go Neural: Higher-Order Graph Neural Networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, 4602–4609. AAAI Press.

Nam, H.; Ha, J.; and Kim, J. 2017. Dual Attention Networks for Multimodal Reasoning and Matching. In *2017*

IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, 2156–2164. IEEE Computer Society.

Patro, B. N.; and Namboodiri, V. P. 2018. Differential Attention for Visual Question Answering. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 7680–7688. Computer Vision Foundation / IEEE Computer Society.

Pennington, J.; Socher, R.; and Manning, C. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. Doha, Qatar: Association for Computational Linguistics.

Scarselli, F.; Gori, M.; Tsoi, A. C.; Hagenbuchner, M.; and Monfardini, G. 2009. The Graph Neural Network Model. *IEEE Trans. Neural Networks*, 20(1): 61–80.

Singh, A.; Mishra, A.; Shekhar, S.; and Chakraborty, A. 2019. From Strings to Things: Knowledge-Enabled VQA Model That Can Read and Reason. 4601–4611.

Sun, M.; and Li, P. 2019. Graph to Graph: a Topology Aware Approach for Graph Structures Learning and Generation. In Chaudhuri, K.; and Sugiyama, M., eds., *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, 2946–2955. PMLR.

Tang, K. 2020. A Scene Graph Generation Codebase in PyTorch. https://github.com/KaihuaTang/Scene-Graph-Benchmark.pytorch.

Tang, K.; Niu, Y.; Huang, J.; Shi, J.; and Zhang, H. 2020. Unbiased Scene Graph Generation From Biased Training. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 3713–3722. IEEE.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Wang, Y.; Che, W.; Guo, J.; and Liu, T. 2018. A Neural Transition-Based Approach for Semantic Dependency Graph Parsing. In McIlraith, S. A.; and Weinberger, K. Q., eds., *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, 5561–5568. AAAI Press.

Wang, Y.; Guo, J.; Che, W.; and Liu, T. 2016. Transition-Based Chinese Semantic Dependency Graph Parsing. In Sun, M.; Huang, X.; Lin, H.; Liu, Z.; and Liu, Y., eds., *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data - 15th China National Conference, CCL 2016, and 4th International Symposium, NLP-NABD 2016, Yantai, China, October 15-16, 2016, Proceedings*, volume 10035 of *Lecture Notes in Computer Science*, 12–24.

Woo, S.; Kim, D.; Cho, D.; and Kweon, I. S. 2018. LinkNet: Relational Embedding for Scene Graph. In Bengio, S.; Wallach, H. M.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 558–568.

Xu, D.; Zhu, Y.; Choy, C. B.; and Fei-Fei, L. 2017. Scene Graph Generation by Iterative Message Passing. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 3097–3106. IEEE Computer Society.

Xu, H.; Jiang, C.; Liang, X.; and Li, Z. 2019. Spatial-aware graph relation network for large-scale object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9298–9307.

Xu, H.; and Saenko, K. 2016. Ask, Attend and Answer: Exploring Question-Guided Spatial Attention for Visual Question Answering. In Leibe, B.; Matas, J.; Sebe, N.; and Welling, M., eds., *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, 2016, Proceedings, Part VII*, volume 9911 of *Lecture Notes in Computer Science*, 451–466. Springer.

Yang, J.; Lu, J.; Lee, S.; Batra, D.; and Parikh, D. 2018. Graph R-CNN for Scene Graph Generation. In Ferrari, V.; Hebert, M.; Sminchisescu, C.; and Weiss, Y., eds., *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part I*, volume 11205 of *Lecture Notes in Computer Science*, 690–706. Springer.

Yang, Z.; Qin, Z.; Yu, J.; and Wan, T. 2020. Prior Visual Relationship Reasoning For Visual Question Answering. In *2020 IEEE International Conference on Image Processing (ICIP)*, 1411–1415. IEEE.

Yin, G.; Sheng, L.; Liu, B.; Yu, N.; Wang, X.; Shao, J.; and Loy, C. C. 2018. Zoom-Net: Mining Deep Feature Interactions for Visual Relationship Recognition. In Ferrari, V.; Hebert, M.; Sminchisescu, C.; and Weiss, Y., eds., *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part III*, volume 11207 of *Lecture Notes in Computer Science*, 330–347. Springer.

Zellers, R.; Yatskar, M.; Thomson, S.; and Choi, Y. 2018. Neural Motifs: Scene Graph Parsing With Global Context. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 5831–5840. IEEE Computer Society.

Zhang, C.; Chao, W.; and Xuan, D. 2019. An Empirical Study on Leveraging Scene Graphs for Visual Question Answering. In *30th British Machine Vision Conference 2019, BMVC 2019, Cardiff, UK, September 9-12, 2019*, 288. BMVA Press.