Lightweight Visual Question Answering using Scene Graphs

Sai Vidyaranya Nuthalapati[1*], Ramraj Chandradevan[2*], Eleonora Giunchiglia[1], Bowen Li[1], Maxime Kayser[1], Thomas Lukasiewicz[1], Carl Yang[2]
[1]Department of Computer Science, University of Oxford, UK, [2]Department of Computer Science, Emory University, US
[1]thomas.lukasiewicz@cs.ox.ac.uk, [2]j.carlyang@emory.edu

ABSTRACT

Visual question answering (VQA) is a challenging problem in machine perception, which requires the deep joint understanding of both visual and textual data. Recent research has advanced the automatic generation of high-quality scene graphs from images, while powerful yet elegant models like graph neural networks (GNNs) have shown a great power in reasoning over graph-structured data. In this work, we propose to bridge the gap between scene graph generation and VQA by leveraging GNNs. In particular, we design a new model called Conditional Enhanced Graph ATtention network (CE-GAT) to encode pairs of visual and semantic scene graphs with both node and edge features, which is seamlessly integrated with a textual question encoder to generate answers through question-graph conditioning. Moreover, to alleviate the training difficulties of CE-GAT towards VQA, we enforce more useful inductive biases in the scene graphs through novel question-guided graph enriching and pruning. Finally, we evaluate the framework on one of the largest available VQA datasets (namely, GQA) with ground-truth scene graphs, achieving the accuracy of 77.87%, compared with the state of the art (namely, the neural state machine (NSM)), which gives 63.17%. Notably, by leveraging existing scene graphs, our framework is much lighter compared with end-to-end VQA methods (e.g., about 95.3% less parameters than a typical NSM).

ACM Reference Format:
Sai Vidyaranya Nuthalapati, Ramraj Chandradevan, Eleonora Giunchiglia, Bowen Li, Maxime Kayser, Thomas Lukasiewicz, Carl Yang. 2021. Lightweight Visual Question Answering using Scene Graphs. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, QLD, Australia.* ACM, New York, NY, USA, 5 pages. https://doi/10.1145/3459637.3482218

1 INTRODUCTION

Recently, visual question answering (VQA) has drawn increasing attention from researchers across different domains, due to its challenging requirement of understanding in vision, language, and commonsense knowledge [1, 6, 10, 14, 20, 25, 36]. Currently, most VQA systems are end-to-end, integrating deep neural networks such as CNNs [4] and RNNs [19] to model visual images and textual questions in the same latent space. Some of the state-of-the-art end-to-end methods even explicitly construct scene graphs (SGs) with objects and their relations in the raw image, and then reason over such graphs with neural network models (see, e.g., [8, 36]).

Meanwhile, active research is being conducted on the more specific task of automatic generation of SGs from raw images, and the field has reached rather satisfactory results [10, 12, 15, 16, 21, 22, 24, 27, 33, 35]. Furthermore, many major VQA datasets have been populated with ground-truth SGs [7, 13], ready to be leveraged by appropriate structured learning methods. However, to our knowledge, no current VQA model directly utilizes existing SGs, and the SGs that they construct within the parameter-heavy end-to-end VQA pipeline may not have a satisfactory quality, which can further degrade the VQA performance.

Present work. In this work, we bridge the gap between VQA and SGs through developing a novel and lightweight model based on graph neural networks (GNNs), called CE-GAT, and two question-guided graph editions to further facilitate the learning of CE-GAT. The direct leverage of existing SGs allows us to base our model on lightweight GNNs with manifested neat superiority in various downstream tasks thanks to their ability of learning representations from structured data [3, 5, 11, 17, 23, 28–32, 34]. Despite being intriguing, the idea of applying GNNs on SGs for VQA (SG-VQA in short) has not been materialised due to the limitations of existing GNNs regarding the gap between traditional graph mining tasks and SG-VQA. Moreover, information in the SGs is often agnostic of the questions (see, e.g., [7, 13]), whose insufficiency and redundancy can pose significant challenges for GNNs. The main contributions of this paper can thus be briefly summarized as follows:

Contribution 1: *Conditional Enhanced Graph ATtention network (CE-GAT).* To address the limitations of existing GNNs when applied on SGs, we design paired graph attention networks to encode the visual and semantic graphs in parallel, and design three-way attentions to additionally model predicate semantics as edge features. Moreover, we leverage a textual question encoder to generate conditions for the edge-enhanced GATs.

Contribution 2: *Question-Guided Graph Editions.* To facilitate the training of GNNs, we propose to edit the SGs according to each particular question by enriching the SGs with negative entities and predicates that appear in the question but not in the SGs, and pruning SGs by removing entities and predicates that are far away from the entities that appear in the questions.

Contribution 3: *Comprehensive Experimental Analysis.* We conduct extensive experiments on a popular VQA dataset called GQA [7], where we mainly compare our results with the state-of-the-art

*Both authors contributed equally.

CIKM '21, November 1–5, 2021, Virtual Event, QLD, Australia

ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3482218

method called NSM [8]. Comprehensive ablation studies demonstrate the effectiveness of our proposed SG-VQA framework and each of its novel components, which outperforms NSM by 23.27% with ground-truth SGs. Moreover, as we avoid a direct modeling of all raw images, the model size and training time of our framework are significantly smaller than end-to-end VQA models like NSM.

2 SG-VQA

2.1 Problem Definition

In this work, we focus on the task of SG-VQA, i.e., scene-graph-based visual question answering. Besides a textual question, the input includes the scene graph, which consists of a pair of visual and semantic graphs [7, 13]. Specifically, the scene graph (SG) $\mathcal{G} = \{\mathcal{G}_v, \mathcal{G}_s\}$, where $\mathcal{G}_v = \{\mathcal{V}_v, \mathcal{E}_v, \phi_v, \psi_v\}$ is the visual graph, and $\mathcal{G}_s = \{\mathcal{V}_s, \mathcal{E}_s, \phi_s, \psi_s\}$ is the semantic graph. $\mathcal{V}$, $\mathcal{E}$, ϕ, and ψ are the node (entity) set, edge (predicate) set, node features, and edge features, respectively. ϕ_v/ψ_v are visual features in the bounding boxes, whereas ϕ_s/ψ_s are semantic features from word embeddings.

The common availability of such scene graphs can be justified by the fact that many VQA datasets nowadays are already equipped with ground-truth scene graphs (e.g., GQA [7] and Visual Genome [13]), and many algorithms have been developed, which keeps pushing the accuracy of the automatic generation of scene graphs from raw images [10, 12, 15, 16, 21, 22, 24, 27, 33, 35].

The output of our SG-VQA is an answer, which is either a predicted categorical label or generated short text. In the general case of textual answers, we aim to model the following probability

$$p(A|Q,\mathcal{G}) = \prod_{i=1}^{m} p(a_i|a_{1:i-1}, Q, \mathcal{G}), \tag{1}$$

where $p(A|Q,\mathcal{G})$ is the probability of generating the answer A (sequence of words a_i) given the question Q and scene graph $\mathcal{G}$.

2.2 Conditional Enhanced Graph Attention Network

As shown in Figure 1, our SG-VQA framework follows a standard encoder-decoder architecture. To generate answers as categorical labels or sequential texts, we use the state-of-the-art attention-based greedy search sequence generator widely used in machine translation [2]. Therefore, the novelty of our work mostly resides in the encoder architecture.

In SG-VQA, the encoder takes a question and a pair of visual and semantic graphs as input. Several unique challenges naturally arise for our encoder: (1) the joint modeling of pairs of visual and semantic graphs, (2) the modeling of both node and edge features in the graphs, and (3) the modeling of graphs under the consideration of different questions.

To deal with these challenges, we design the novel encoder of the Conditional Enhanced Graph Attention Network (CE-GAT) model, which builds on the powerful GNN of the graph attention network (GAT) model [23].

Paired GAT. As shown in Figure 1, our scene graph encoder needs to jointly model a pair of visual and semantic graphs for each image, while highlighting important entities and predicates to generate the answer. To allow this, we devise paired GATs to encode both graphs with separate propagation functions and attention mechanisms. Specifically, a standard GAT can be implemented as follows:

$$h_i^{l+1} = \sigma\Big(\sum_{j\in N_i} \alpha_{ij}\mathbf{W}h_j^l\Big), \tag{2}$$

$$\alpha_{ij} = \mathrm{softmax}_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k\in N_i}\exp(e_{ik})}, \tag{3}$$

$$e_{ij} = a(\mathbf{W}h_i^l, \mathbf{W}h_j^l), \tag{4}$$

where $h_i^l, h_j^l \in \mathbb{R}^F$ are the node embeddings with dimension size F at the l-th GAT layer, $h_i^{l+1} \in \mathbb{R}^{F'}$ is the node feature with F' feature dimension at the $l+1^{th}$ layer, σ is a non-linear function, $\mathbf{W}$ are the learnable GAT parameters for embedding projections among layers, N_i is the neighbourhood of the node i, α_{ij} are the normalized attention coefficients, e_{ij} are the attention coefficients, and $a: \mathbb{R}^{F'}\times\mathbb{R}^{F'} \to \mathbb{R}$ is a single-layered feed-forward neural network whose parameterized weight matrix is a, which represents a shared attention mechanism. Moreover, h^0 is set to ϕ_s and ϕ_v in the visual and the semantic graph, respectively.

After computing the node embeddings on each of the two graphs, a readout function is devised on both GATs as follows:

$$h_{\mathcal{G}} = \sum_{v\in\mathcal{V}} h_v^L, \tag{5}$$

where node embeddings in the last (L-th) GAT layer are aggregated through the element-wise sum to yield a scene graph embedding $h_{\mathcal{G}}$, where $\mathcal{G}$ is to be replaced by $\mathcal{G}_s$ or $\mathcal{G}_v$ on the semantic and the visual graph, respectively. The final scene graph representations from both GATs on the semantic and the visual graphs are then the concatenation as a whole $h_{\mathcal{G}} = h_{\mathcal{G}_s} \odot h_{\mathcal{G}_v}$.

Edge-Enhanced GAT. The traditional GAT can only model node features [23]. However, as illustrated in Figure 1, the edges in both visual and semantic graphs are associated with corresponding features. To enable the modeling of such edge features (i.e., ψ_v/ψ_s), we design a novel edge-enhanced GAT by revising Eqs. (3)–(4) into

$$\alpha_{ij} = \frac{\exp\Big(a^T[\mathbf{W}h_i \odot \mathbf{W}h_j \odot \psi_{ij}]\Big)}{\sum_{k\in N_i}\exp\Big(a^T[\mathbf{W}h_i \odot \mathbf{W}h_k \odot \psi_{ik}]\Big)}, \tag{6}$$

where the attention weights are computed with the edge features ψ taken into consideration.

Conditional GAT. Existing GNNs including GATs are mostly unconditional, i.e., they compute a universal node or graph representation for each graph [26]. However, our scene graph representation should be conditioned on the current questions, so as to capture the most important information to generate relevant answers. To enable this, we design a novel question-conditional GAT. Specifically, we first use the multi-layer gated recurrent unit (GRU) to encode the question as $h_q = GRU(Q)$, where Q is the question consisting of question tokens $\{q_i\}$.

After that, we concatenate h_q to the original node features ϕ_v and ϕ_s in both the visual and the semantic graphs to allow the GATs to be aware of the question, which we call conditional GATs. As a simplified ablation, we can also directly concatenate h_q to the scene graph embeddings $h_{\mathcal{G}}$ after GATs without question, which we study in the experiments.

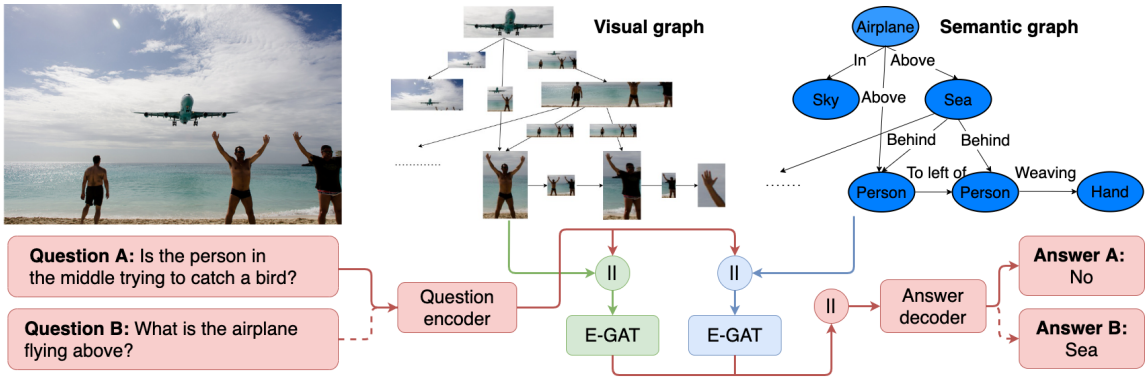

Figure 1: An illustration of our SG-VQA framework with CE-GAT.

2.3 Question-Guided Scene Graph Enriching and Pruning

The scene graph of each image contains an average of 16 objects and 50 predicates. However, many of them can be irrelevant to the particular questions, which adds an unnecessary burden to the deep learning models, making them hard to train and possibly easy to overfit. To this end, we aim to explicitly help our CE-GAT model through simple question-guided scene graph editions, which essentially generate a question-oriented scene graph for each question based on enriching and pruning the original scene graphs.

SG Enriching. Consider again our toy example in Figure 1. Question A is asking about BIRD, while there is no BIRD at all in the image. However, this straightforward knowledge of *"no bird in the image"* is not directly available in the scene graphs, and likely the visual and semantic features of a BIRD are indeed close to those of an AIRPLANE, which is indeed in the image. The lack of such straightforward knowledge can really confuse the model. To deal with this, we propose to enrich the scene graphs with *negative* entities and predicates that appear in the questions but not in the scene graphs.

Specifically, we extract part-of-speech tags from questions, and select all nouns as question-relevant nodes. For each semantic graph and question pair, we compute the cosine similarity score for each pair of semantic graph node and question-relevant node. By thresholding at 0.5, we get all matched and unmatched pairs of nodes. We regard question-relevant nodes that are not matched to any semantic graph node as the negative nodes ($\mathcal{V}_s^1$), and add them into the semantic graph together with a self-link of "does not exist" for each of them ($\mathcal{E}_s^1$). After such question-guided scene graph enriching, our populated semantic graph would be $\mathcal{G}_s^{en} = \{\mathcal{V}_s^{en}, \mathcal{E}_s^{en}, \phi_s^{en}, \psi_s^{en}\}$, where $\mathcal{V}_s^{en} = \mathcal{V}_s \cup \mathcal{V}_s^1$, $\mathcal{E}_s^{en} = \mathcal{E}_s \cup \mathcal{E}_s^1$, $\phi_s^{en} = \phi_s \cup \phi_s^1$, and $\psi_s^{en} = \psi_s \cup \psi_s^1$.

SG Pruning. Consider again our toy example in Figure 1. To answer Question B, one only needs to look at the triple of "(airplain, above, sea)", instead of the whole scene graph. In fact, to answer most questions, it is intuitive that one only needs the local information around the entities and predicates mentioned in the question, and most other things far away in the scene graphs are just confusing and misleading for the model. To deal with this, we propose to prune the scene graphs by removing the *irrelevant* entities and predicates that are not within the k-hop neighbourhoods of those mentioned in the question.

Specifically, continuing with the procedures that we employ in the SG enrichment, we pick the unmatched nodes in the semantic graph as our candidates for pruning. Then, we expand the local k-hop neighbourhood of each matched node based on the connecting predicates (of either directions) and only prune nodes $\mathcal{V}_s^2$ that are outside all such k-hop neighbourhoods (e.g., k=1 or 2). We remove $\mathcal{V}_s^2$ together with all predicates $\mathcal{E}_s^2$ that have them on at least one end from the semantic graph. That is, we only searched for pruning nodes ($\mathcal{V}_s^{"}$) in the expanded local neighbourhood, and selected any connected edges as pruning predicates ($\mathcal{E}_s^{"}$). In this way, our pruning step results in a refined semantic graph $\mathcal{G}_s^{pr} = \{\mathcal{V}_s \setminus \mathcal{V}_s^2, \mathcal{E}_s \setminus \mathcal{E}_s^2, \phi_s \setminus \phi_s^2, \psi_s \setminus \psi_s^2\}$.

The SG enriching and SG pruning can be conducted either alone or together (one by one, and the order does not matter).

3 EXPERIMENTS

Experimental settings. We use GQA [7] to comprehensively evaluate our proposed framework. Compared to other publicly available VQA datasets [9, 13, 18, 37], GQA consists of high-quality ground-truth image scene graphs that can be used for compositional and semantic understanding of real-world images. The dataset is available with 113K images with ground-truth scene graphs and 22M question-answer pairs, which we use to evaluate our models. It is also possible to evaluate our model on automatically inferred scene graphs in future work.

GQA has two different data settings based on imbalanced and balanced question-answer pairs. The balanced training and evaluation dataset is ten times smaller than the imbalanced one. Besides, the balanced evaluation dataset is not publicly available and cannot be used for rapid model development. For these reasons, we used the imbalanced data setting and limited the training sample size to 10M question-answer pairs. We evaluated the trained models on a testing set of 2M question-answer pairs. Following [8], we use a similar set of evaluation metrics[1] in Table 1.

We implemented our framework using PyTorch. The training and evaluation take about 30 hours and 6 hours, respectively, on a single NVIDIA Quadro RTX 8000 GPU server with 48 GB memory and an Intel Xeon Gold 6248R @ 3.00 GHz CPU. Regarding model

[1] https://cs.stanford.edu/people/dorarad/gqa/evaluate.html

Table 1: Performance of our model variants on GQA in comparison to human judgements and the SOTA VQA framework.

Model	Binary	Open	Consistency	Validity	Plausibility	Accuracy
Human judgements [7]	91.20	87.40	98.40	98.90	97.20	89.30
Neural State Machine [8]	78.94	49.25	**93.25**	96.41	84.28	63.17
CE-GAT on visual graph alone	74.17	50.54	86.06	96.96	96.28	71.02
CE-GAT on semantic graph alone	75.43	47.26	86.05	96.79	96.04	71.68
CE-GAT w/o graph edge attention	72.95	49.94	83.56	96.89	96.04	69.89
CE-GAT w/o question conditioning	70.02	49.80	81.74	96.54	95.59	67.33
CE-GAT Full	76.99	50.96	85.55	96.84	96.14	73.52
CE-GAT Full w. question-guided enriching	**78.97**	54.19	87.98	96.93	96.30	75.67
CE-GAT Full w. question-guided pruning	78.21	55.78	87.92	96.95	96.31	75.22
CE-GAT Full w. both editions	77.91	**77.63**	87.43	**97.13**	**96.61**	**77.87**

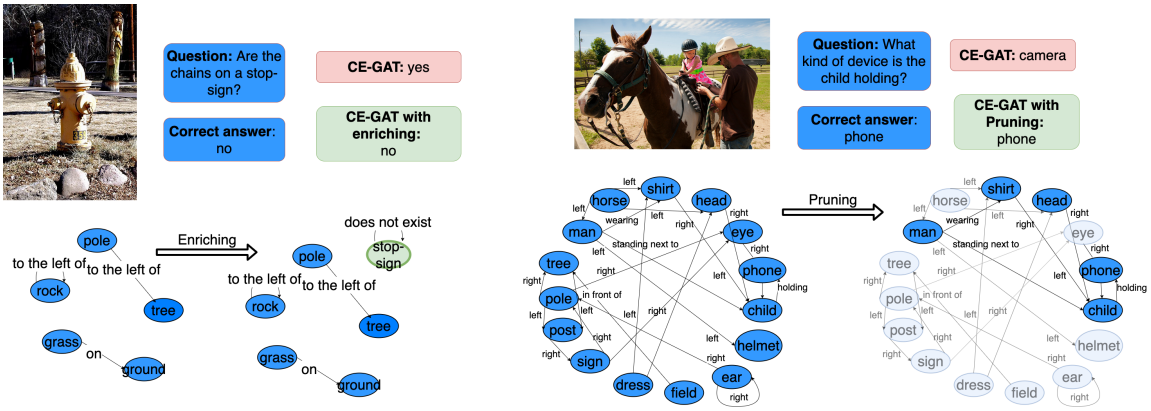

Figure 2: Case studies on the effectiveness of question-guided scene graph enriching and pruning.

hyperparameters, we set the dimensions of all hidden layers in the LSTM, MLP, and GCN models to 100, and the numbers of layers in all GCN models to 2. We fix the batch size to 64, learning rate to 1e-3, and max epoch to 200 with patience-based early stop.

Quantitative evaluations. Table 1 shows the performance of different CE-GAT variants compared with the state-of-the-art VQA model of NSM and human judgement. In terms of accuracy, our CE-GAT Full model is able to achieve a significantly better performance compared with NSM, not only due to its direct access to the scene graphs but also its appropriate modeling of them. Such improvements are consistent almost across all other evaluation metrics as well, indicating the overall effectiveness of CE-GAT.

Moreover, we comprehensively evaluated different ablations of CE-GAT, including CE-GAT on the visual/semantic graph alone, CE-GAT without graph edge attention, and CE-GAT without question conditioning. As we can observe from the results, given accuracy as the major metric, applying paired GATs on visual and semantic graphs brings around 3.5% relative gains, applying edge attention brings around 5.2% relative gains, and applying node-level question conditioning brings around 9.2% relative gains. Such results clearly corroborate the sanity of our model designs.

Finally, we evaluate the CE-GAT Full model with the novel question-guided graph editions. We can observe that using either of the two approaches alone can bring around 2.3% and 2.7% improvements in accuracy on top of the CE-GAT Full model, while combining the two approaches together can further boost that improvement to 5.8%. The results are also consistent across other evaluation metrics, which demonstrates the effectiveness of our novel yet simple graph edition operations.

Case Studies. Figure 2 illustrates two cases where question-guided enriching and pruning allow the CE-GAT model to correct the otherwise wrong answers on two specific questions and images. In the first case, there is no stop-sign in the image or the scene graph, while the question mentions stop-sign. Our framework enriches the scene graph by adding a node of "stop-sign" and a self-link of "does not exist", which helped the model to notice its absence and predict the correct answer of "no". In the second case, there are too many nodes and links in the original scene graph, while the question is focused on the "child". Our framework prunes the scene graph by removing many irrelevant nodes and links far away from the node "child", which helped the model to get rid of the distractions which may lead to noisy answers such as "camera" and predict the correct answer of "phone".

4 ACKNOWLEDGMENTS

This work has been partially supported by the Alan Turing Institute under the EPSRC grant EP/N510129/1, by the AXA Research Fund, the EPSRC grant EP/R013667/1, and by the EU TAILOR grant. Eleonora Giunchiglia is supported by an Oxford-DeepMind Graduate Scholarship and by the EPSRC grant EP/N509711/1. Maxime Kayser is supported by Elsevier BV. We also acknowledge the use of Oxford's Advanced Research Computing (ARC) facility, of the EPSRC-funded Tier 2 facility JADE (EP/P020275/1), and of GPU computing support by Scan Computers International Ltd.

REFERENCES

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual question answering. In *ICCV*.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

[3] Jie Chen, Tengfei Ma, and Cao Xiao. 2018. FastGCN: Fast learning with graph convolutional networks via importance sampling. In *ICLR*.

[4] Ross Girshick. 2015. Fast R-CNN. In *ICCV*.

[5] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *NIPS*.

[6] Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. 2017. Learning to reason: End-to-end module networks for visual question answering. In *ICCV*.

[7] Drew A. Hudson and Christopher D. Manning. 2019. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*.

[8] Drew A. Hudson and Christopher D. Manning. 2019. Learning by abstraction: The neural state machine. In *NeurIPS*.

[9] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. 2017. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*.

[10] Xuan Kan, Hejie Cui, and Carl Yang. 2021. Zero-shot scene graph relation prediction through commonsense knowledge integration. In *ECML-PKDD*.

[11] Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR*.

[12] Boris Knyazev, Harm de Vries, Cătălina Cangea, Graham W. Taylor, Aaron Courville, and Eugene Belilovsky. 2020. Graph density-aware losses for novel compositions in scene graph generation. In *BMVC*.

[13] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV* 123, 1 (2017), 32–73.

[14] Linjie Li, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Relation-aware graph attention network for visual question answering. In *ICCV*.

[15] Yikang Li, Wanli Ouyang, Bolei Zhou, Jianping Shi, Chao Zhang, and Xiaogang Wang. 2018. Factorizable Net: An efficient subgraph-based framework for scene graph generation. In *ECCV*.

[16] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. 2020. GPS-Net: Graph property sensing network for scene graph generation. In *CVPR*.

[17] Ziqi Liu, Chaochao Chen, Longfei Li, Jun Zhou, Xiaolong Li, Le Song, and Yuan Qi. 2019. GeniePath: Graph neural networks with adaptive receptive paths. In *AAAI*.

[18] Mateusz Malinowski and Mario Fritz. 2014. A multi-world approach to question answering about real-world scenes based on uncertain input. In *NeurIPS*.

[19] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Interspeech*.

[20] Kevin J. Shih, Saurabh Singh, and Derek Hoiem. 2016. Where to look: Focus regions for visual question answering. In *CVPR*.

[21] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. 2020. Unbiased scene graph generation from biased training. In *CVPR*.

[22] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. 2019. Learning to compose dynamic tree structures for visual contexts. In *CVPR*.

[23] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph attention networks. In *ICLR*.

[24] Sanghyun Woo, Dahun Kim, Donghyeon Cho, and In So Kweon. 2018. LinkNet: Relational embedding for scene graph. In *NeurIPS*.

[25] Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2017. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding* (2017).

[26] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S. Yu Philip. 2020. A comprehensive survey on graph neural networks. *TNNLS* (2020).

[27] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. 2017. Scene graph generation by iterative message passing. In *CVPR*.

[28] Carl Yang, Aditya Pal, Andrew Zhai, Nikil Pancha, Jiawei Han, Chuck Rosenberg, and Jure Leskovec. 2020. MultiSage: Empowering GraphSage with contextualized multi-embedding on web-scale multipartite networks. In *KDD*.

[29] Carl Yang, Yuxin Xiao, Yu Zhang, Yizhou Sun, and Jiawei Han. 2020. Heterogeneous network representation learning: A unified framework with survey and benchmark. In *TKDE*.

[30] Carl Yang, Jieyu Zhang, and Jiawei Han. 2020. Co-embedding network nodes and hierarchical labels with taxonomy based generative adversarial nets. In *ICDM*.

[31] Carl Yang, Jieyu Zhang, Haonan Wang, Sha Li, Myungwan Kim, Matt Walker, Yiou Xiao, and Jiawei Han. 2020. Relation learning on social networks with multi-modal graph edge variational autoencoders. In *WSDM*.

[32] Carl Yang, Peiye Zhuang, Wenhan Shi, Alan Luu, and Pan Li. 2019. Conditional structure generation through graph variational generative adversarial nets. In *NIPS*.

[33] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. 2018. Graph R-CNN for scene graph generation. In *ECCV*.

[34] Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. 2018. Hierarchical graph representation learning with differentiable pooling. In *NIPS*.

[35] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *CVPR*.

[36] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. VinVL: Revisiting visual representations in vision-language models. *arXiv preprint arXiv:2101.00529* (2021).

[37] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. 2016. Visual7W: Grounded question answering in images. In *CVPR*.