Scene Graph Reasoning for Visual Question Answering

Marcel Hildebrandt [*] [1] [2] Hang Li [*] [1] [3] Rajat Koner [*] [2] Volker Tresp [1] [2] Stephan Günnemann [3]

Abstract

Visual question answering is concerned with answering free-form questions about an image. Since it requires a deep linguistic understanding of the question and the ability to associate it with various objects that are present in the image, it is an ambitious task and requires techniques from both computer vision and natural language processing. We propose a novel method that approaches the task by performing context-driven, sequential reasoning based on the objects and their semantic and spatial relationships present in the scene. As a first step, we derive a scene graph which describes the objects in the image, as well as their attributes and their mutual relationships. A reinforcement agent then learns to autonomously navigate over the extracted scene graph to generate paths, which are then the basis for deriving answers. We conduct a first experimental study on the challenging GQA dataset with manually curated scene graphs, where our method almost reaches the level of human performance.

1. Introduction

Visual Question Answering (VQA) is a demanding task that involves understanding and reasoning over two data modalities: images and natural language. Given an image and a free-form question—which formulates a question about the presented scene— the issue is for the algorithm to find the correct answer.

VQA has recently found interest in different research communities and various real-world data sets, such as the *VQA* data set (Antol et al., 2015), have been generated. It has been argued that, in the *VQA* data set, many of the apparently challenging reasoning tasks can be solved by an algorithm by exploiting trivial prior knowledge, and thus by shortcuts to proper reasoning (e.g., clouds are white or doors are made of wood). To address these shortcomings, the *GQA* dataset (Hudson & Manning, 2019b) has been developed. Compared to other real-world datasets, *GQA* is more suitable to evaluate reasoning abilities since the images and questions are carefully filtered to make the data less prone to biases.

Plenty of VQA approaches are agnostic towards the explicit relational structure of the objects in the presented scene and rely on monolithic neural network architectures that process regional features of the image separately (Anderson et al., 2018; Yang et al., 2016). While these methods led to promising results on previous datasets, they lack explicit compositional reasoning abilities which results in weaker performance on more challenging datasets such as *GQA* . Other works (Teney et al., 2017; Shi et al., 2019; Hudson & Manning, 2019a) perform reasoning on explicitly detected objects and interactive semantic and spatial relationships among them. These approaches are closely related to the scene graph representations (Johnson et al., 2015) of an image, where detected objects are labeled as nodes and relationship between the objects are labeled as edges.

In this work we aim to combine VQA techniques with recent research advances in the area of statistical relation learning on knowledge graphs (KGs). KGs provide human readable, structured representations of knowledge about the real world via collections of factual statements. Inspired by multi-hop reasoning methods on KGs such as (Das et al., 2018; Hildebrandt et al., 2020), we model the VQA task as a path-finding problem on scene graphs. The underlying idea can be summarized with the phrase: Learn to walk to the correct answer. More specifically, given an image, we consider a scene graph and train a reinforcement learning agent to conduct a policy-guided random walk on the scene graph until a conclusive inference path is obtained. In contrast to purely embedding-based approaches, our method provides explicit reasoning chains that leads to the derived answers. To sum up, our major contributions are as follows.

- To the best of our knowledge, we propose the first VQA method that employs reinforcement learning for reasoning on scene graphs.
- We conduct an experimental study to analyze the reasoning capabilities of our method. Instead of generat-

[*] Equal contribution [1] Siemens AG, Munich, Germany [2] Ludwig Maximilian University of Munich, Munich, Germany [3] Technical University of Munich, Munich Germany. Correspondence to: Marcel Hildebrandt <marcel.hildebrandt@siemens.com>.

ICML Workshop GRL+, Vienna, Austria, 2020. Copyright 2020 by the author(s).

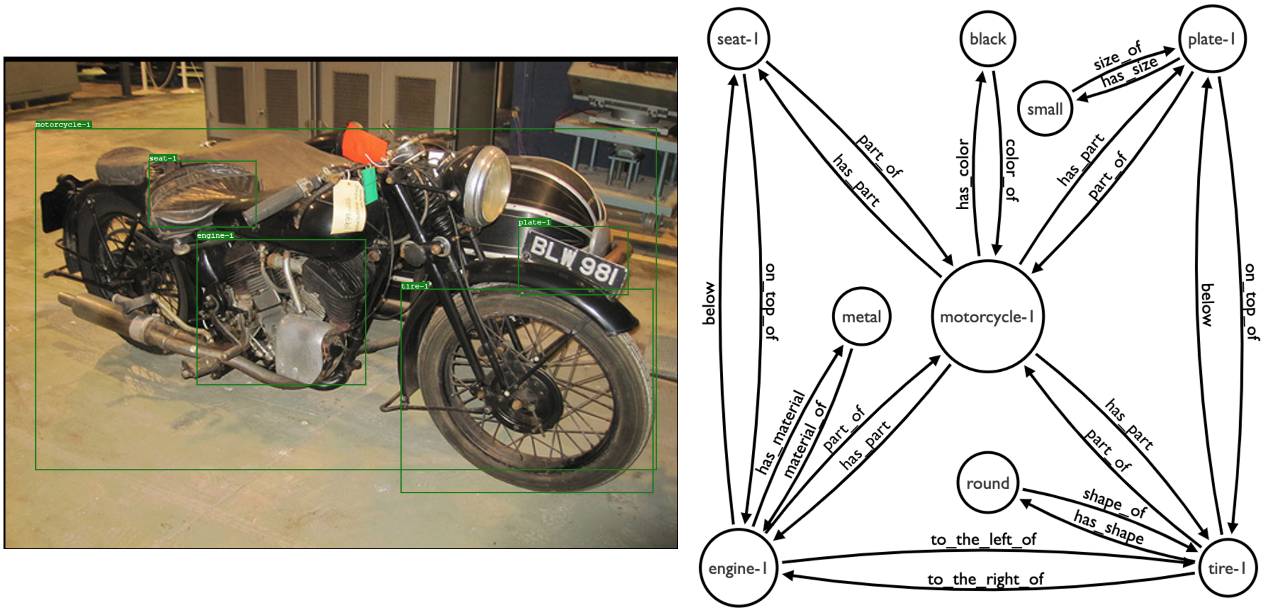


Figure 1. Example of an image and the corresponding scene graph.

ing our own scene graphs, we consider manually curated scene graphs from the *GQA* dataset for these first experiments. This setting allows to isolate the noise associated to the visual perception task and focuses solely on the language understanding and reasoning task. Thereby, we can show that our method achieves human-like performance.

2. Method

The task of VQA is framed as a scene graph traversal problem. Starting from a hub node that is connected to all other nodes, an agent sequentially samples transition to a neighboring node on the scene graph until the node corresponding to the answer is reached. In this way, by adding transitions to the current path, the reasoning chain is successively extended. Before describing the decision problem of the agent, we introduce the notation that we use throughout this work.

Notation A scene graph is a directed multigraph where each node corresponds to a scene entity which is either an object associated with a bounding box or an attribute of an object. Each scene entity comes with a type that corresponds to the predicted object or attribute label. Typed edges specify how scene entities are related to each other. More formally, let $\mathcal{E}$ denote the set of scene entities and consider the set of binary relations $\mathcal{R}$. Then a scene graph $\mathcal{SG} \subset \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ is a collection of ordered triples (s, p, o) – subject, predicate, and object. For example, as shown in Figure 1, the triple *(motorcycle-1, has_part, tire-1)* indicates that both a motorcycle (subject) and a tire (object) are detected in the image. The predicate *has_part* indicates the relation between the entities. Moreover, we denote with p^{-1} the inverse relation corresponding to the predicate p. For the remainder of this work, we impose completeness with respect to inverse relations in the sense that for every $(s, p, o) \in \mathcal{SG}$ it is implied that $(o, p^{-1}, s) \in \mathcal{SG}$. Moreover, we add a so-called hub node (*hub*) to every scene graph which is connected to all other nodes.

Environment The state space of the agent $\mathcal{S}$ is given by $\mathcal{E} \times \mathcal{Q}$ where $\mathcal{E}$ are the nodes of a scene graph $\mathcal{SG}$ and $\mathcal{Q}$ denotes the set of all questions. The state at time t is the entity e_t at which the agent is currently located and the question Q. Thus, a state $S_t \in \mathcal{S}$ for time $t \in \mathbb{N}$ is represented by $S_t = (e_t, Q)$. The set of available actions from a state S_t is denoted by $\mathcal{A}_{S_t}$. It contains all outgoing edges from the node e_t together with their corresponding object nodes. More formally, $\mathcal{A}_{S_t} = \{(r, e) \in \mathcal{R} \times \mathcal{E} : S_t = (e_t, Q) \wedge (e_t, r, e) \in \mathcal{SG}\}$. Moreover, we denote with $A_t \in \mathcal{A}_{S_t}$ the action that the agent performed at time t. We include self-loops for each node in $\mathcal{SG}$ that produce a *NO_OP*-label. These self-loops allow the agent to remain at the current location if it reaches the answer node. To answer binary questions, we also include artificial *yes* and *no* nodes in the scene graph. The agent can transition to these nodes in the final step.

Embeddings We initialize words in Q with GloVe embeddings (Pennington et al., 2014) with dimension $d = 300$.

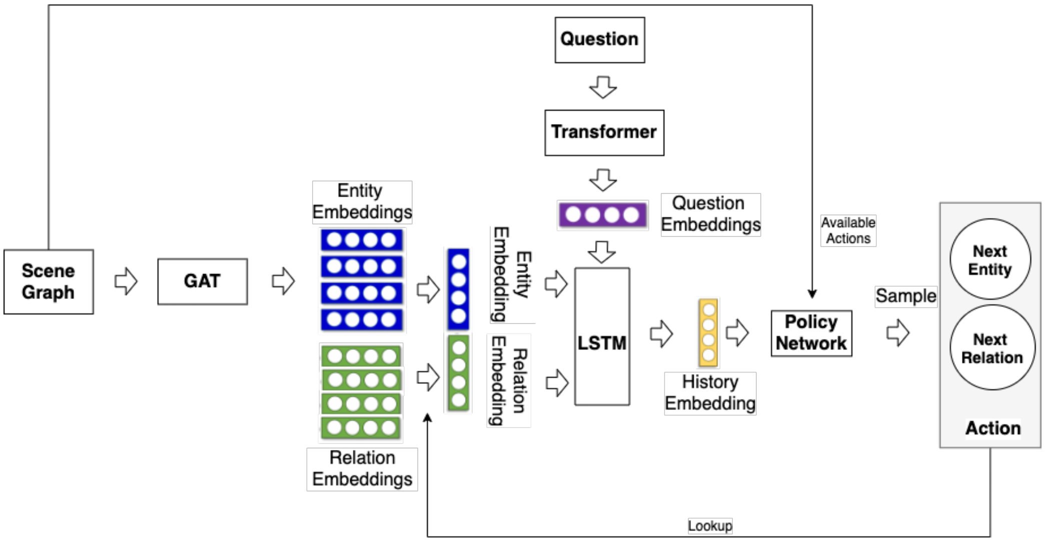


Figure 2. The architecture of our scene graph reasoning module.

Similarly we initialize entities and relations in $\mathcal{SG}$ with the embeddings of their type labels. In the scene graph, the node embeddings are passed through a multi-layered graph attention network (GAT) (Veličković et al., 2017). Extending the idea from graph convolutional networks (Kipf & Welling, 2016) with a self-attention mechanism, GATs mimic the convolution operator on regular grids where an entity embedding is formed by aggregating node features from its neighbors. Thus, the resulting embeddings are context-aware, which makes nodes with the same type but different graph neighborhoods distinguishable. To produce an embedding for the question Q, we first apply a Transformer (Vaswani et al., 2017), followed by a mean pooling operation.

Policy We denote the agent's history until time t with the tuple $H_t = (H_{t-1}, A_{t-1})$ for $t \geq 1$ and $H_0 = hub$ along with $A_0 = \emptyset$ for $t = 0$. The history is encoded via a multilayered LSTM (Hochreiter & Schmidhuber, 1997)

$$\mathbf{h}_t = \text{LSTM}\left(\mathbf{a}_{t-1}\right) \,, \tag{1}$$

where $\mathbf{a}_{t-1} = [\mathbf{r}_{t-1}, \mathbf{e}_t] \in \mathbb{R}^{2d}$ corresponds to the embedding of the previous action with $\mathbf{r}_{t-1}$ and $\mathbf{e}_t$ denoting the embeddings of the edge and the target node into $\mathbb{R}^d$, respectively. The history-dependent action distribution is given by

$$\mathbf{d}_t = \text{softmax}\left(\mathbf{A}_t \left(\mathbf{W}_2 \text{ReLU}\left(\mathbf{W}_1 \left[\mathbf{h}_t, \mathbf{Q}\right]\right)\right)\right) \,, \tag{2}$$

where the rows of $\mathbf{A}_t \in \mathbb{R}^{|\mathcal{A}_{S_t}| \times d}$ contain latent representations of all admissible actions. Moreover, $\mathbf{Q} \in \mathbb{R}^d$ encodes the question Q. The action $A_t = (r, e) \in \mathcal{A}_{S_t}$ is drawn according to categorical $(\mathbf{d}_t)$.Equations (1) and (2) induce a stochastic policy π_θ, where θ denotes the set of trainable parameters.

Rewards and Optimization After sampling T transitions, a terminal reward is assigned according to

$$R = \begin{cases} 1 & \text{if } e_T \text{ is the answer to } Q, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

We employ REINFORCE (Williams, 1992) to maximize the expected rewards. Thus, the agent's maximization problem is given by

$$\underset{\theta}{\arg\max}\, \mathbb{E}_{Q \sim \mathcal{T}} \mathbb{E}_{A_1, A_2, \ldots, A_N \sim \pi_\theta} \left[R \,\middle|\, e_c \right] \,, \tag{4}$$

where $\mathcal{T}$ denote the set of training questions.

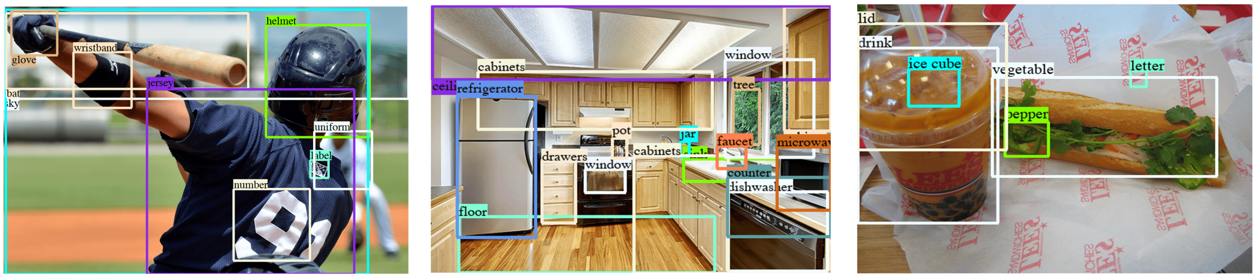
3. Experiments

Dataset and Experimental Setup Hudson & Manning (2019b) introduced the *GQA* dataset with the goal of addressing key shortcomings of previous VQA datasets, such as *CLEVR* (Johnson et al., 2017) or the *VQA* dataset (Antol et al., 2015). *GQA* is more suitable for evaluating reasoning and compositional abilities of a model, in a realistic setting. The *GQA* dataset contains 113K images and around 1.2M questions split into roughly $80\%/10\%/10\%$ for the training, validation and testing. The overall vocabulary size consists of 3097 words.

Results and Discussion In this work, we report first results on an experimental study on manually curated scene

Table 1. A comparison of our method with human performance based on manually curated scene graphs.

Method	Binary	Open	Consistency	Validity	Plausibility	Accuracy
Human (Hudson & Manning, 2019b)	91.2	87.4	98.4	98.9	97.2	89.3
TRRNet	77.91	50.22	89.84	85.15	96.47	63.20
Our Method	90.41	90.86	91.92	93.68	93.13	90.63



HUB $\xrightarrow{HUB}$ white $\xrightarrow{has\_color^{-1}}$ number $\xrightarrow{RESET}$ HUB $\xrightarrow{HUB}$ black $\xrightarrow{has\_color^{-1}}$ wristband → no

(a) Question: Is the color of the number the same as that of the wristband?
Answer: No.

HUB $\xrightarrow{HUB}$ large $\xrightarrow{NO\_OP}$ large $\xrightarrow{has\_size^{-1}}$ refrigerator

(b) Question: What is the name of the appliance that is not small?
Answer: Refrigerator.

HUB $\xrightarrow{HUB}$ green $\xrightarrow{has\_color^{-1}}$ pepper $\xrightarrow{RESET}$ HUB $\xrightarrow{HUB}$ green $\xrightarrow{has\_color^{-1}}$ vegetable → yes

(c) Do both the pepper and the vegetable to the right of the ice cube have green color?
Answer: Yes.

Figure 3. Three examples question and the corresponding images and paths.

graphs that are provided in the *GQA* dataset. In this setting, the true reasoning and language understanding capabilities of our model can be analyzed. Table 1 shows the performance of our method and compares it with the human performance reported in (Hudson & Manning, 2019b) and with TRRNet[1], the best performing single method submission to the *GQA* Real-World Visual Reasoning Challenge 2019. Along with the accuracy on open questions ("Open"), binary questions (yes/no) ("Binary"), and the overall accuracy ("Accuracy"), we also report the additional metric "Consistency" (answers should not contradict themselves), "Validity" (answers are in the range of a question; e.g., *red* is a valid answer when asked for the color of an object), "Plausibility" (answers should be reasonable; e.g., red is a reasonable color of an apple reasonable, blue is not), as proposed in (Hudson & Manning, 2019b).

The results in Table 1 show that our method achieves human level performance with respect to most metrics. Figure 3 shows three examples of scene graph traversals, which produced the correct answer. An advantage of our approach is that the sequential reasoning process makes the model output transparent and easier to debug in cases of failures.

Although the results in Table 1 are very encouraging, the performance numbers are not directly comparable, since the underlying data sets are different and since we operated on manually curated scene graphs. As part of ongoing work, we are exploring different methods for extracting the scene graph automatically from the images. This step is not really the focus of this work but turns out to be the weak part of our overall VQA approach. By using the scene graph generation procedure proposed in (Yang et al., 2018), we found that in the cases where the answer to an open question was contained in the scene graph, our method was able to achieve 53.24% accuracy on this subset of the data (which could be compared to the 50.22% accuracy of TRRNet). We are currently working on improving the scene graph generation framework by integrating recent advancements in object detection such as (Tan et al., 2019) or in scene graph generation (Zellers et al., 2018; Zhang et al., 2019; Koner et al., 2020). We hope that these methods lead to more accurate scene graphs so that our method is able to retain close to human performance as presented in this paper.

4. Conclusion

We have proposed a novel method for visual question answering based on multi-hop sequential reasoning and deep reinforcement learning. Concretely, an agent is trained to extract conclusive reasoning paths from scene graphs. To analyze the reasoning abilities of our method in a controlled setting, we conducted a preliminary experimental study on manually curated scene graphs and concluded that our method reaches human performance. In future works, we plan to incorporate state-of-the-art scene graph generation into our method to cover the complete VQA pipeline.

[1] https://evalai.cloudcv.org/web/challenges/challenge-page/225/leaderboard/733#leaderboardrank-5

References

Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., and Zhang, L. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6077–6086, 2018.

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., and Parikh, D. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

Das, R., Dhuliawala, S., Zaheer, M., Vilnis, L., Durugkar, I., Krishnamurthy, A., Smola, A., and McCallum, A. Go for a walk and arrive at the answer: Reasoning over paths in knowledge bases using reinforcement learning. In *ICLR*, 2018.

Hildebrandt, M., Serna, J. A. Q., Ma, Y., Ringsquandl, M., Joblin, M., and Tresp, V. Reasoning on knowledge graphs with debate dynamics. *arXiv preprint arXiv:2001.00461*, 2020.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Hudson, D. and Manning, C. D. Learning by abstraction: The neural state machine. In *Advances in Neural Information Processing Systems*, pp. 5901–5914, 2019a.

Hudson, D. A. and Manning, C. D. Gqa: A new dataset for real-world visual reasoning and compositional question answering. *arXiv preprint arXiv:1902.09506*, 2019b.

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

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., and Girshick, R. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2901–2910, 2017.

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Koner, R., Sinhamahapatra, P., and Tresp, V. Relation transformer network. *arXiv preprint arXiv:2004.06193*, 2020.

Pennington, J., Socher, R., and Manning, C. D. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014. URL http://www.aclweb.org/anthology/D14-1162.

Shi, J., Zhang, H., and Li, J. Explainable and explicit visual reasoning over scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8376–8384, 2019.

Tan, M., Pang, R., and Le, Q. V. Efficientdet: Scalable and efficient object detection. *arXiv preprint arXiv:1911.09070*, 2019.

Teney, D., Liu, L., and van Den Hengel, A. Graph-structured representations for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, 2017.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., and Bengio, Y. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

Yang, Z., He, X., Gao, J., Deng, L., and Smola, A. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 21–29, 2016.

Yang, Z., Qin, Z., Yu, J., and Hu, Y. Scene graph reasoning with prior visual relationship for visual question answering. *arXiv preprint arXiv:1812.09681*, 2018.

Zellers, R., Yatskar, M., Thomson, S., and Choi, Y. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5831–5840, 2018.

Zhang, J., Shih, K. J., Elgammal, A., Tao, A., and Catanzaro, B. Graphical contrastive losses for scene graph parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11535–11543, 2019.