Scene Graph-Enhanced Visual Question Answering for Context-Aware Image Understanding

Kim Juhyeong, Ryu Sangyeon

ISNLP, School of Computing, Gachon University

Abstract. Visual Question Answering (VQA) requires a deep understanding of both visual and textual modalities to accurately interpret and respond to questions about images. This task involves identifying and analyzing visual elements such as objects, their attributes, and interrelationships, while simultaneously comprehending the contextual nuances of the accompanying question. In this study, we leverage scene graphs—a structured knowledge representation capturing visual and semantic information in the form of triplets—to enhance VQA performance. First, we extract entities from the question that are present in the scene graph and append them to the question using special tokens, thereby enriching the textual input with relevant visual context. Second, we introduce Graph-based Relative Positional Encoding, a technique recently prominent in the NLP domain, to encode the structural and relational information of the scene graph within the model. To the best of our knowledge, this is the first application of graph-based relative positional encoding in computer vision tasks. Our experimental results demonstrate the effectiveness of scene graphs in integrating visual and semantic information for VQA and other computer vision tasks.

Keywords: Visual Question Answering, Knowledge Graph, Computer Vision, Natural Language Processing

1 Introduction

Visual Question Answering (VQA) is a task that necessitates a comprehensive understanding of both visual and textual modalities. It requires the ability to identify and interpret visual elements within an image, such as objects, attributes, and their relationships, while simultaneously integrating the contextual nuances presented by the accompanying question. VQA remains a key challenge in computer vision as it lies in its demand for deeper reasoning about visual content, which involves synthesizing diverse pieces of visual information and understanding their interconnections. This also require a more profound understanding of both visual and linguistic contexts.

Scene Graph is a knowledge graph containing visual/semantic information provided in a single scene image. (Example in Fig. 1). A scene graph adopts the format of a traditional knowledge graph, representing information in the form

Fig. 1. Example of Scene Graph from GQA Dataset[1]

of triplets. Each triplet consists of two entities—a subject and an object—and a relation, or predicate, that links them. For instance, the description "a white bowl is next to a green round apple" can be translated into the triplet subject: bowl, object: apple, relation: next to, with additional attributes such as "white" describing the bowl and "green" and "round" describing the apple. This structured representation captures both the relationships between entities and their specific attributes, providing a detailed and interpretable format for integrating visual and semantic information.

By leveraging scene graphs, it becomes possible to integrate visual and semantic information within a single scene, enabling a more comprehensive understanding of the visual content. Building on this approach, we combined the ViT [2] and T5 [3] model architecture and explored two distinct methods to enhance VQA performance using scene graphs. First, we extracted the entities mentioned in the question from the scene graph and appended them directly to the question using special tokens. Second, we applied Graph-based Relative Positional Encoding [4]-a technique that has recently garnered significant attention in the NLP community—to enhance Visual Question Answering (VQA). To the best of our knowledge, there have been no prior studies of utilizing graph-based relative positional encoding in computer vision tasks. We conducted experiments with aforementioned two methods and baseline and demonstrate the promising impact of integrating scene graph-based structured knowledge to computer vision tasks.

2 Scene Graph-Enhanced Visual Question Answering

2.1 Baseline

The primary objective of this baseline is to understand the inherent capabilities of the model when relying on direct visual and textual inputs. By processing only the essential components—question, answer, and image—the baseline serves as a control, enabling us to measure the influence and effectiveness of integrating scene graph information in subsequent methods. In this setup, the inputs consist of the question, the corresponding answer and the image. Initially, ViT [2] preprocess the image to obtain the input embedding for each image. The text model, T5 [3] pprerocesses inputs by tokenizing the question and answer texts and generating processed tokens Then, the preprocessed visual embedding and textual embeddings are concatenated and used as input of T5 [3] to generate predictions. This baseline allows us to quantify the improvements achieved through the incorporation of structured scene graph data in more advanced approaches. Overall architecture is shown in Fig. 2

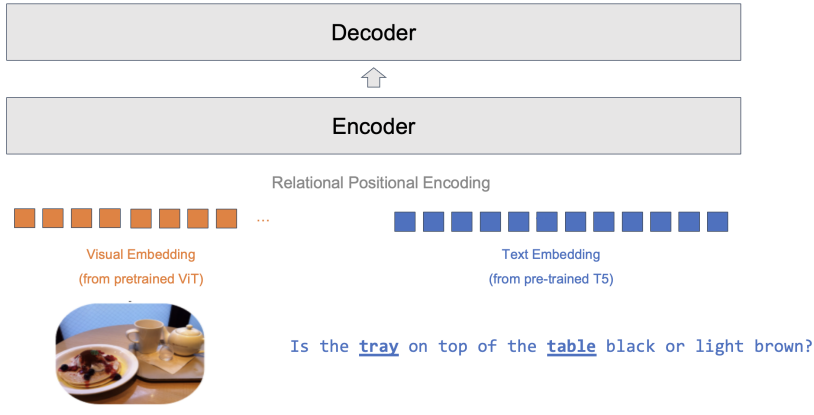

Fig. 2. Architecture of Baseline. Inputs are image embedding by ViT, question and answer embedding by T5.

2.2 Method A: Directly integrating Graph Information

In this method, we begin by extracting entities from the question that are present within the corresponding scene graph. These extracted entities are then appended to the original question using special tokens, thereby enriching the textual input with relevant visual context derived from the scene graph. This augmentation allows the model to leverage structured data, such as object attributes and their relationships, which provides a more comprehensive understanding of

the visual scene. By incorporating these additional details, the model gains the ability to interpret complex relationships and attributes within the image, leading to improved accuracy and more contextually relevant answers. Similar to baseline, visual embeddings and textual embeddings are obtained by ViT [2] and T5 [3] and injected as combined input to T5 encoder. Overall architecture is shown in Fig. 3

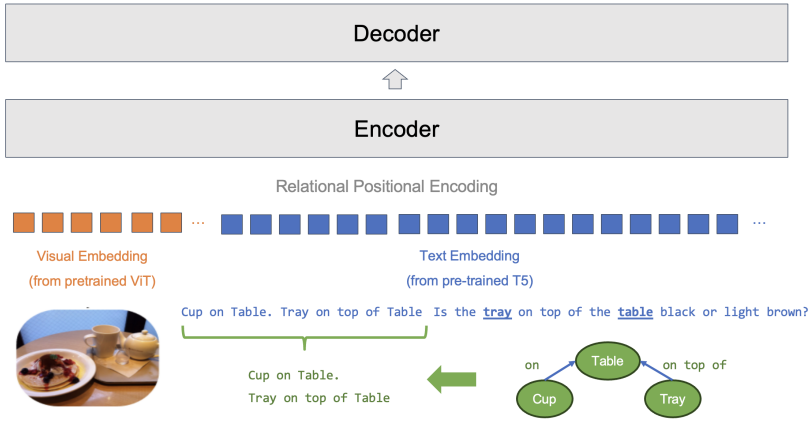

Fig. 3. Architecture of Method A: Entities are first extracted from the question and appended to it. The inputs include image embeddings from ViT and textual embeddings of the scene graph-enriched question and answer generated by T5.

2.3 Method B: Integrating Graph-based Relative Positional Encoding

The second approach introduces Graph-based Relative Positional Encoding [4], a technique that has recently gained significant attention in the natural language processing (NLP) community. This method involves encoding the structural and relational information of the scene graph into the model through a specialized positional encoding scheme.

First we convert each graph into Extended Levi-graph form, replacing each edge with a node that contains the relation name as text feature, and connect the new node to the head and

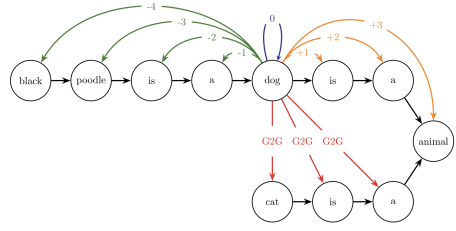

Fig. 4. Extended Levi graph for [5], [4] and our Method B. Nodes, edges and direction represent words, relations and direction of the relation predicate.

tail of the original edge via unlabeled edges. We follow node-node relation representation of [4] and [5]. This represent preserves the direction of the original edge and represent the relation between two nodes as the length of the shortest path between them.

Next, we compute a relative positions matrix of size $N \times N$ for the graph tokens, where N is the length of the graph token sequence. This computation is based on the integer distance values that carry directional signs (positive or negative) in the extended Levi-graph form, representing the relations between nodes. Specifically, each entry in the matrix corresponds to the shortest path distance between a pair of nodes, with the sign indicating the directionality of the relationship. Positive values denote one direction, while negative values indicate the opposite direction.

These relative position values are then incorporated into the attention mechanism of the transformer model. By adding the relative positions matrix to the attention weights, the model effectively encodes the structural information of the graph. This integration allows the self-attention layers to consider not only the content of the nodes but also their positional and relational contexts within the graph. Consequently, the model can better capture both the positional and structural dependencies among the nodes, enhancing its ability to reason about complex relationships and interactions present in the visual scene.

For the model input, we concatenate the baseline input with graph tokens. Note that we concatenate the baseline relative positions for baseline input with Graph-based relative position for graph tokens. Overall architecture is shown in Fig. 5

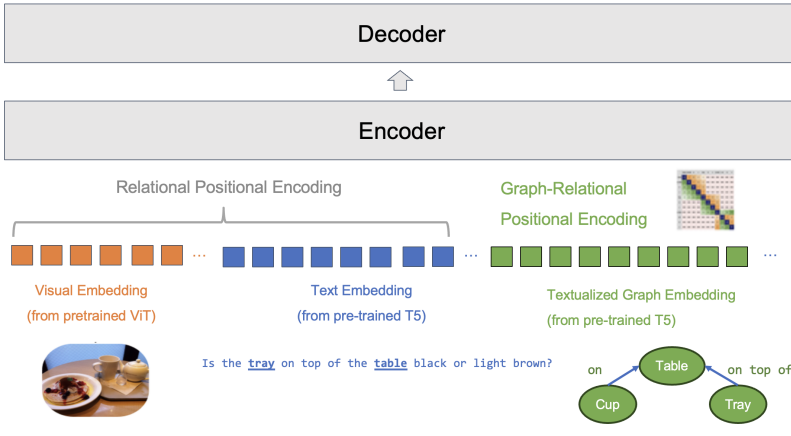

Fig. 5. Architecture of Method B: Entities are first extracted from the question and appended to it. The inputs consist of image embeddings from ViT, textual embeddings of the question and answer generated by T5, and graph token embeddings from T5 enriched with Graph-Relational Positional Encoding.

3 Experiments

3.1 Dataset

The GQA dataset [1] is a large-scale visual question answering dataset with real images from the Visual Genome dataset and balanced question-answer pairs. Each training and validation image is also associated with scene graph annotations describing the classes and attributes of those objects in the scene, and their pairwise relations. Along with the images and question-answer pairs, the GQA dataset [1] provides two types of pre-extracted visual features for each image – convolutional grid features of size 7x7x2048 extracted from a ResNet-101 network trained on ImageNet, and object detection features of size Ndetx2048 (where Ndet is the number of detected objects in each image with a maximum of 100 per image) from a Faster R-CNN detector.

3.2 Settings

We employed the T5-base model [3] as our text encoder and the ViT [2] base model with a patch size of 16 and pre-trained on ImageNet-21k [2] from HuggingFace library. The experiments were conducted on Baseline, Method A, and Method B, where Method B was evaluated under conditions ranging from 1 to 4 hops. As the size of GQA [1] is enormous, we randomly sampled 50k and 10k samples from each training and evaluation dataset and conducted experiments over 4 epochs. AdamW optimizer was used and the learning rate was set to $5e-5$. We evaluated experiment results with Accuracy (Exact Match), F1, Precision and Recall. All experiments were conducted using NVIDIA 3090 GPUs.

3.3 Results

Table 1. Results of our experiments. EM refers to Accuracy (Exact-Match). The highest performance for each metric is indicated in bold and second highest performance is indicated in underlined text.

Method	Hop	EM	F1	Precision	Recall
Baseline	-	35.19	35.77	35.81	35.86
A	-	**35.74**	**36.49**	**36.21**	**36.28**
B	1 hop	34.53	34.92	35.07	34.86
B	2 hop	<u>35.41</u>	<u>35.85</u>	<u>36.01</u>	<u>35.77</u>
B	3 hop	34.72	35.13	35.29	35.06
B	4 hop	34.98	35.37	35.50	35.31

The experimental results, summarized in Table 1, demonstrate the effectiveness of scene graph enhancing Visual Question Answering (VQA) performance.

The Baseline model, which relies solely on the question, answer, and image data, achieves an Exact-Match (EM) accuracy of 35.19%, with corresponding F1, Precision, and Recall scores of 35.77%, 35.81%, and 35.86%, respectively. In contrast, Method A, which integrates scene graph information by appending extracted entities to the question, significantly outperforms the baseline across all metrics, attaining an EM accuracy of 35.74%, F1 of 36.49%, Precision of 36.21%, and Recall of 36.28%. This improvement underscores the value of incorporating structured visual and semantic context from scene graphs to enrich the model's understanding and reasoning capabilities.

Additionally, Method B, which employs Graph-based Relative Positional Encoding with varying hop levels, reveals nuanced performance trends. At 2 hops, Method B achieves the second-highest performance with an EM accuracy of 35.41%, F1 of 35.85%, Precision of 36.01%, and Recall of 35.77%, all of which are underlined to indicate their status as the second-best results. However, increasing the number of hops beyond two leads to a decline in performance. These findings suggest that while incorporating relational information through graph-based positional encoding can enhance VQA performance, there is an optimal level of depth beyond which the benefits diminish. Overall, the results highlight the potential of the integration of scene graph-based structured knowledge.

4 Analysis

Although we demonstrated the effectiveness of the scene graph, there is a gap between our expectation that Method B would outperform Method A and the actual experimental results. In this section we discuss the analysis of the probable causes.

Firstly, we discuss the ambiguity caused by the repeated appearance of objects with the same name across multiple images, which complicates directly referencing the relations within a single image. In Method A, all objects relevant to the question were retrieved from the scene graph and concatenated to form the input. In our implementation of Method B, however, all objects in the scene graph were first retrieved, but only one object was randomly selected, and multiple hops were then performed to extract related triplets. Due to the nature of the images, where multiple redundant objects often coexist, Task B's approach failed to select the appropriate object. This resulted in noisy and irrelevant data being included in the input, which we hypothesize caused Method B to perform worse than Method A.

Secondly, We assume multi-hop extraction of our Method B implementation does not provide sufficient triplets for model input. Since we extracted multi-hop triplets linearly, only a limited number of triplets corresponding to the number of hops were included as graph tokens, which failed to provide sufficient structural knowledge from the scene graph.

5 Conclusions

In this study, we explored two distinct approaches to enhance VQA performance using scene graphs, a structured representation of visual and semantic information. The first approach enriches the textual input by appending question-relevant entities from the scene graph as special tokens, providing additional visual context. The second introduces Graph-based Relative Positional Encoding to capture the structural and relational information of the scene graph. Our study emphasize the potential of integrating scene graph-based structured knowledge as a solution to addressing current challenges in computer vision tasks.

For future work, we aim to improve Method B by refining the object selection algorithm to ensure that extracted triplets are contextually aligned with the question, avoiding irrelevant or random selections. Additionally, we plan to enhance multi-hop reasoning by transitioning from a linear extraction to a branching strategy. For instance, at 1-hop, two objects connected to the starting node will be extracted, while at 2-hop, two objects will be selected for each node from 1-hop, resulting in four contextually meaningful triplets. This approach is expected to capture more comprehensive and relevant information from the scene graph.

References

1. Hudson, D.A., Manning, C.D.: Gqa: A new dataset for real-world visual reasoning and compositional question answering (2019)
2. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale (2021)
3. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer (2023)
4. Plenz, M., Frank, A.: Graph language models. In Ku, L.W., Martins, A., Srikumar, V., eds.: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Bangkok, Thailand, Association for Computational Linguistics (August 2024) 4477–4494
5. Schmitt, M., Ribeiro, L.F.R., Dufter, P., Gurevych, I., Schütze, H.: Modeling graph structure via relative position for text generation from knowledge graphs. In Panchenko, A., Malliaros, F.D., Logacheva, V., Jana, A., Ustalov, D., Jansen, P., eds.: Proceedings of the Fifteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-15), Mexico City, Mexico, Association for Computational Linguistics (June 2021) 10–21