Can MLLMs Really Understand Knowledge Graph? A Comprehensive Evaluation

Anonymous ACL submission

Abstract

Although Multimodal Large Language Models (MLLMs) have achieved remarkable performance in various complex tasks, they still face challenges in understanding Knowledge Graphs (KGs), which are typical graphs with structured semantics. In this paper, we conduct a comprehensive evaluation to assess the capability of MLLMs in this aspect and investigate key factors influencing their performance in understanding and reasoning over KGs across different dimensions, with a particular focus on factors related to the triple recognition of KGs. Our study yields several key findings and insights that contribute to advancing this research domain. We find that MLLMs indeed have limitations in understanding complicated KGs, which is primarily attributed to the poor recognition ability of textual triples in KGs, particularly for graphs with special layouts or high density. On this basis, we propose a fine-tuning method to enhance the understanding capabilities of MLLMs on KGs, achieving an accuracy increase of 7.3% compared to baseline model.

1 Introduction

Recently, MLLMs have demonstrated remarkable capabilities in handling a wide range of complex tasks across multiple modalities (Bai et al., 2024; Fu et al., 2024), including visual question answering (He et al., 2024b), image captioning (Agarwal and Verma, 2024) and multimodal reasoning (Yan et al., 2024). Knowledge graphs, as an important form of structured data, not only store and represent complex relationships between entities but also contain extensive factual knowledge. Due to the inherent reasoning capabilities of the graph structure, visualized KGs are often easier for humans to understand, arousing our spatial and visual reasoning abilities. For MLLMs, recent work such as GITA (Wei et al., 2024) has shown that vision-only models can outperform LLM-based models in certain tasks without fine-tuning, underscoring

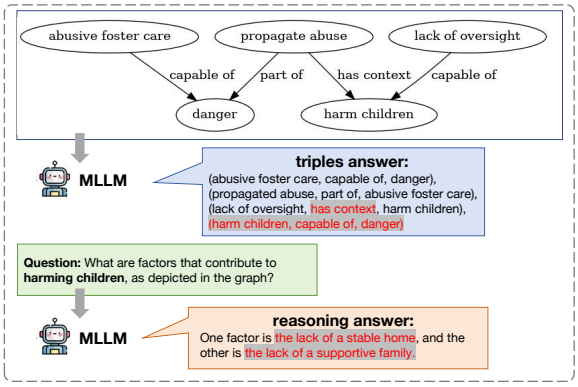


Figure 1: Examplle of knowledge graph understanding. The incorrect responses are marked as red.

the significant potential of MLLMs in addressing graph-related problems and promoting the generalization of traditional models.

However, unlike pixel-based image data well handled by MLLMs, the visualization of graphs emphasizes the relationships between nodes and edges associated with semantics, revealing the limited ability of MLLMs in graph understanding. (Rahmanzadehgervi et al., 2024). As shown in Figure 1, for the reasoning task on the triples of a KG, the generated answer is incorrect. It is derived from common knowledge rather than aligning with the edges in the graph. Therefore, at this critical junction in the development of MLLMs, we raise a significant research question: **Can MLLMs really understand and reason over KGs?**

In the meantime, some studies evaluate LLMs' capabilities in structural and semantic-related graph understanding tasks (Li et al., 2024b; Wang et al., 2023; Dai et al., 2024; Guo et al., 2023), and find that LLMs exhibit a preference for linearized triples over natural language texts (Dai et al., 2024). However, both formats require transforming the original graphs, which fail to fully leveraging the multimodal capabilites of MLLMs. To pursue a more intuitive and natural interaction, several emerging

studies have begun exploring the use of MLLMs to process graph images with the assistance of textual instruction prompts (Wei et al., 2024; Li et al., 2024e; Zhao et al., 2025), which mainly address graph reasoning tasks rooted in graph theory, without dealing with semantic information. On the other hand, (Ai et al., 2024; Zhang et al., 2024) construct datasets covering a wide range of graph-related tasks in daily scenarios for preliminary evaluation of semantic understanding, but they do not specifically aim at KGs with triple relationships, nor do they provide a deep analysis of the internal mechanisms of MLLMs to explain the evaluation results. Returning to the example above, we observe that the error can be clearly traced to the misrecognition of triples related to "harming children", as evidenced by the intermediate response. This draws forth another critical question: **Does the recognition of triples primarily and directly affects the reasoning of KGs, making it the bottleneck of KG understanding task?**

In this paper, we focus on a comprehensive evaluation of the current state-of-the-art MLLMs on the task of KG understanding and reasoning. Our goal is to highlight the strengths and weaknesses of MLLMs in solving such tasks. Meanwhile, we examine the impact of image presentation factors in triple recognition, such as layout and density. The evaluation is grounded on multi-dimension datasets constructed via adapting and extending existing ones, including generated graph images and corresponding question-answer pairs for both simple and complex reasoning tasks. Based on the key findings from our evaluation, we fine-tune the best-performing model Qwen2VL on triples-based question-answer pairs using the dot layout, and achieve improved performance across all layouts.

Key takeaways To the best of our knowledge, this is the first study to thoroughly explore MLLMs' capabilities on KG understanding tasks. The key findings and insights are summarized as follows:

- The accuracy of triple recognition directly determines the accuracy of KG understanding and reasoning. (1) MLLMs indeed have limitations in recognizing KG structures, particularly for complicated graph images, such as those with unreadable layouts or higher edge densities. (2) When textual triples are provided for reasoning, models with initially poor recognition abilities, such as LLaVA, a greater performance improvement. That implies the triple recognition ability sets the lower limit of KG understanding and reasoning.

- Layout plays a crucial role in the triple recognition ability, and the easiest layout for recognition is "dot" with a hierarchical structure. (Section 4.3).

- As the number of triples and the density of edges increase, the recognition ability declines, highlighting the necessity to dynamically adjust the layout to focus on complex or peripheral parts (Section 4.4).

- After fine-tuning on the triple recognition data of different layouts, the model's reasoning ability is enhanced. Among them, finetuning on the triple recognition data using the dot layout yields the best overall performance across all layouts, achieving an accuracy increase of 7.3% compared to zero-shot models (Section 4.5).

2 Related Work

2.1 Multi-Modal Large Language Model

A typical MLLM comprises three modules: a pre-trained modality encoder, a pre-trained LLM, and a learnable projector connecting the modality encoder and the LLM. Recently, the "ViT-MLP-LLM" paradigm has been widely adopted in numerous MLLM studies (Liu et al., 2024a; Chen et al., 2024a,b; Zhu et al., 2023; Lu et al., 2024; Wang et al., 2024a; Li et al., 2024a; Wang et al., 2024b). The projector is trained to embed information from non-language modalities into the text semantic space that the LLM can understand. In LLaVA (Liu et al., 2024b), a single linear layer is used as a simple projector. To better extract information from non-language modalities, usually images or videos, Vision Transformer (ViT) (Dosovitskiy, 2020) has been improved in various ways. Qwen2VL (Wang et al., 2024a) can process images of any resolution by modifying ViT, removing the original absolute position embeddings, and introducing 2D-RoPE (Su et al., 2024; Heo et al., 2024) to capture the two-dimensional positional information of images. InternVL (Chen et al., 2024b) is the first to align a large-scale vision encoder with LLMs, scaling up the vision encoder to 6 billion parameters, resulting in the InternViT-6B model.

With their enhanced capability to comprehend visual inputs, contemporary MLLMs demonstrate

remarkable performance in various tasks, including visual question answering (VQA) (He et al., 2024b; Antol et al., 2015; Uppal et al., 2022), image captioning (Vinyals et al., 2015; Agarwal and Verma, 2024; Vaishnavi and Narmatha, 2024), and multimodal reasoning (Wang et al., 2024c; Yan and Lee, 2024; Yan et al., 2024). Despite these advancements, open-source MLLMs still lag behind commercial models like GPT-4o (Achiam et al., 2023) and Gemini-Pro (Team et al., 2024) in complex reasoning tasks (Liu et al., 2024c).

2.2 Graph Understanding and Reasoning

LLMs on graph-related problems Large Language Models (LLMs) have shown significant potentials across various domains, leading to their exploration in structured graph data. Researchers (Li et al., 2024d) summarizes LLMs to assist graph-related problems in three primary roles: enhancer, predictor, and aligner. As predictors, LLMs directly generate predictions through prompts including Flatten-based Prediction and GNN-based Prediction (Li et al., 2024c). Flatten-based prediction typically transform a graph structure into a sequence of nodes or tokens. GraphText(Zhao et al., 2023) leverages graph-syntax trees to convert a graph structure to a sequence of nodes. InstructGLM (Ye et al., 2023) designs a series of scalable prompts replacing traditional GNN predictors with LLMs, and MR-MKG (Lee et al., 2024) leverages multimodal knowledge graphs to enhance LLMs' reasoning capabilities.

There are also some evaluation works on LLMs in graph tasks. (Wang et al., 2023) shows that language models demonstrate preliminary graph reasoning abilities. (Li et al., 2024b) emphasizes that the capabilities of LLMs in handling structured data are still under-explored and demonstrates the effectiveness of LLM4Graph in enhancing LLMs' proficiency in graph analysis. (Dai et al., 2024) reveals that linearized triples are more effective than fluent natural language text in helping LLMs understand KG information and answer fact-intensive questions.GPT4Graph(Guo et al., 2023) assesses the proficiency of LLMs in comprehending graph data by employing a diverse range of structural and semantic-related tasks, indicating that there is still a long way for an LLM to understand graph data.

MLLMs on graph-related problems As the advent of MLLMs, we can directly analyze and understand representations of graph structures through visualization. GITA (Wei et al., 2024) and VisionGraph (Li et al., 2024e) introduced frameworks that leverage MLLMs for fundamental graph reasoning tasks by converting graphs into image representations, highlighting the advantages of MLLMs' visual intelligence, Beyond basic graph tasks, (Elhenawy et al., 2024) extended the use of MLLMs to combinatorial problems, such as solving the traveling salesman problem (TSP) using visual and textual information. (Zhao et al., 2025) reveals that MLLMs can tackle graph-structured challenges from combinatorial problems to sequential decision-making without the need for complex training or fine-tuning. For various graph-related tasks in daily scenarios, (Ai et al., 2024) introduces a benchmark for multimodal graph and leverage VLMs to encode the graph images with varying structures across different domains.

3 Methodology

The paper primarily focuses on knowledge graph understanding and reasoning tasks. The graph structure is denoted as $G = \{V, E\}$, where V and E represent the sets of nodes and edges, respectively. A graph visualizer is used to generate visual representations of the structural graph. The visual input I_G is given by $I_G = V(G, \Delta)$, where Δ represents the customizable graph-related image styles. The task requirement T includes specific operations or questions related to the graph. We also incorporate a prompt instruction P, resulting in the question text $Q_G^T = (T, P)$.

Different task dimensions require different instructions. Q_G^T includes two types of tasks: recognition and reasoning. Recognition refers to identifying the nodes, edges, and triples in the knowledge graph image, while reasoning refers to answering the reasoning questions constructed based on the triples of a knowledge graph. Our evaluation covers the following dimensions: step-by-step decomposition of triplet recognition, as well as recognition and reasoning performance under different layouts and densities. We feed both the visual input I_G and the textual input T into an MLLM to generate the target text $Y = f(I_G, Q_G)$. The correctness of the model's output Y is compared with the ground truth answer $\bar{Y}$ and calculating the accuracy.

Table 1: Statistics of datasets used in the evaluation

Dataset	ExplaGraphs	WebQSP-20	WebQSP-density
#Graphs	2766	4737	4737
Avg. #Nodes	5.17	32.55	8.34
Avg. #Edges	4.25	20	20

4 Experiment

4.1 Experiment setup

4.1.1 Dataset

We construct various datasets for evaluating knowledge graphs in different dimensions, providing text in triple format and generating knowledge graph images. We use the open-source tool Graphviz (Gansner and North, 2000) as the image formatting tool[1]. It can automatically design the layouts of visual graphs and is particularly suitable for building large-scale datasets. Table 1 summarizes the statistics of these datasets. An example of each dataset is presented in Appendix A.

ExplaGraphs A dataset for generative commonsense reasoning (Saha et al., 2021). We visualize the triplet-form data converted in (He et al., 2024a), including adjustment of six different visual layouts.

WebQSP A large-scale multi-hop knowledge graph QA dataset (Yih et al., 2016). We select the first 20 triples (WebQSP-20) and the top 20 triples with the highest density (WebQSP-density) from each knowledge graph for visualization, in order to assess the impact of triple count and density.

4.1.2 Models

The models we evaluate include both advanced open-source and closed-source models as follows.

LLaVA uses language-only GPT-4 to generate multimodal language-image instruction-following data (Liu et al., 2024b), enabling the connection of a vision encoder and a language model via a simple linear layer for general-purpose applications.

LLaVA-OV inherits the minimalism design of LLaVA series (Li et al., 2024a), whose primary goals include effectively leveraging the pre-trained capabilities of both the LLM and visual model. The proposed Higher AnyRes strategy can serve as a flexible visual representation framework, adaptable for multi-image and video representations.

[1]https://graphviz.org/

Qwen2VL retains the Qwen-VL (Bai et al., 2023) framework, which integrates vision encoders and language models. To further enhance the model's ability of effectively perceiving and comprehending visual information in videos, it introduces several key upgrades including naive dynamic resolution and Multimodal Rotary Position Embedding (M-RoPE) (Wang et al., 2024a).

InternVL2 utilizes the same architecture as InternVL 1.5 (Chen et al., 2024a), specifically the ViT-MLP-LLM configuration referenced in various existing studies. To enhance the scalability for high resolution, a pixel shuffle (unshuffle) operation is employed to reduce the number of visual tokens to one-quarter of the original.

GPT-4o is a closed-source multimodal model (Achiam et al., 2023), which can accept text/audio/image/video inputs and generate text/audio/image outputs. It represents a step towards more natural human-computer interaction.

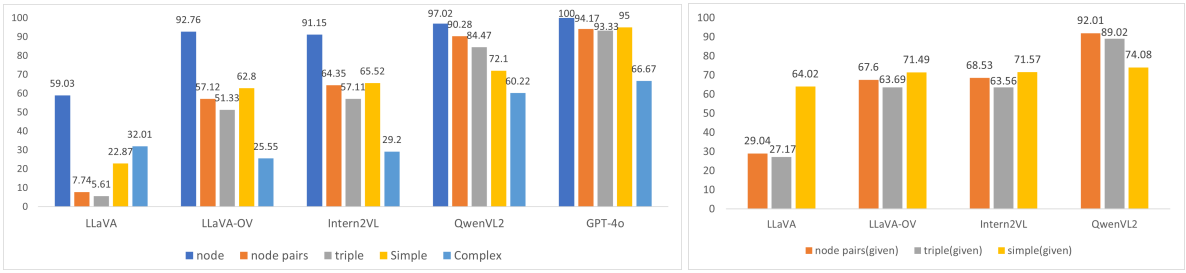
To ensure the consistency of parameter sizes with this closed-source model, the 7B model is used for LLaVA, LLaVA-OV, and Qwen2VL, while the 8B model is used for InternVL2.

4.1.3 Instruction Setting

For the recognition task, we directly ask the model about the elements of the triples involved in the KG in a zero-shot manner. For the reasoning task, our method for constructing reasoning questions follows previous work (Ai et al., 2024), employing an automatic annotation process generated by Gemini (Anil et al., 2023), and posing two levels of reasoning tasks: Simple (1 hop) and Complex (2 or more hops). For simple and one-hop questions, direct judgment can be made, while for more complex questions involving multiple triples, direct matching is infeasible. Since Gemini is used to generate the questions and its high accuracy in recognizing triples has been certified (Ai et al., 2024), we first use it to assess whether the answers provided for complex questions are correct. Then, we perform manual screening to determine the final accuracy. All the instruction prompts used in this paper are listed in Appendix B.

4.2 Evaluation with Task Decomposition

In order to thoroughly investigate the triple recognition process, as demonstrated in Figure 3, we decompose the recognition task for nodes, node pairs without relation type, and complete triples in



(a) Step-by-step Evaluation.

(b) Recognition and reasoning based on given nodes and edges.

Figure 2: Evaluation with task decomposition.

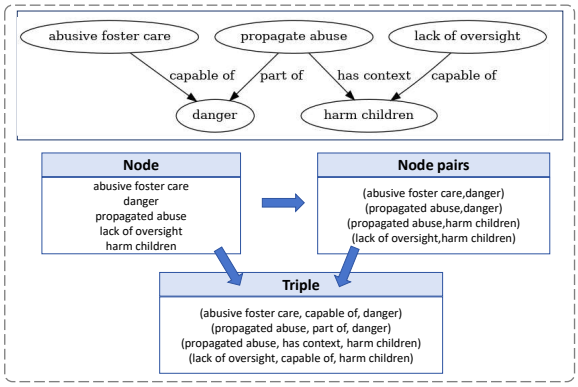


Figure 3: Decomposition of the recognition task

a sequential manner. The evaluation results in Figure 2(a) shows that the overall accuracy decreases from node recognition to node pair recognition and further to triple recognition, progressing from easier to harder tasks. That is intuitive and certifies the validity of our evaluation.

Across different models, the Pearson correlation coefficient between tasks "Triple" and "Simple" reaches 0.99, and the correlation with "Complex" is 0.96, which is the highest among all the triple recognition factors. Therefore, **there is a strong positive correlation between the accuracy of recognition tasks and the accuracy of reasoning task**. The closed-source model GPT-4o has shown good performance in both recognition and reasoning. The three models, LLaVA-OV, Qwen2VL, and InternVL2 exhibit consistent trends in recognizing nodes, node pairs, and triples, while LLAVA experiences a sharp performance drop in node pair and triple recognition tasks. Overall, the model performance is ranked as GPT-4o > Qwen2VL > InternVL2 > LLaVA-OV > LLaVA, so Qwen2VL is more suitable for processing images with graph structures in open-source MLLMs.

To further analyze MLLMs' performance on specific step of the KG recognition and focus on concrete problems within this step, we provide correct information from previous steps and conduct ad-hoc evaluations. Specifically, "Node pairs (given)" refers to providing node names in the prompt and then recognizing the node pairs, "triple (given)" refers to providing both node and edge names and then identifying the complete triples, and "Simple (given)" refers to answering simple reasoning based on the provided set of triples. As shown in Figure 2(b), after providing the basic structure names, the accuracy for node pairs and triples improves to some extent compared to direct recognition, but it is still relatively low for initially poor models like LLaVA. This can be explained that during the triple identification process, there are still issues such as confusion between edge and node names, missing triples, incorrect edges, nonexistent relationship caused by hallucination, incorrect recognition in the direction of directed edges and so on. However, providing all textual triples significantly enhances the reasoning accuracy across all models. **When textual triples are provided for reasoning, models with initially poor recognition abilities, such as LLaVA, a greater performance improvement.** Therefore, compared to the recognition of complete triples, node recognition and node pair recognition are not crucial steps, so in the following evaluation, we only pay attention to the evaluation accuracy of triple recognition.

4.3 Evaluation of layout

Layout refers to various algorithms for projecting abstract graphs into a space for visualization. According to the previous study (Wei et al., 2024), layout variations of visual graphs play a crucial role in mitigating the visual confusion caused by the spatial arrangement within a visual graph. However, that paper lacks a separate analysis of each

Table 2: Evaluation of recognition and reasoning for different layouts

layout	Triple					Simple					Simple(given)
	LLaVA	LLaVA-OV	InternVL2	Qwen2VL	GPT-4o	LLaVA	LLaVA-OV	InternVL2	Qwen2VL	GPT-4o	Qwen2VL
dot	5.61	51.33	57.11	84.47	93.33	22.87	55.34	65.52	72.10	95.00	74.08
circo	1.09	19.35	41.26	76.70	90.00	7.62	42.38	58.08	72.26	95.00	74.09
twopi	1.25	5.41	9.03	11.38	29.27	15.09	41.01	51.62	58.99	89.31	72.41
neato	1.67	11.18	13.80	22.09	31.25	19.36	48.32	59.98	60.92	92.21	70.50
fdp	1.10	12.62	15.54	28.48	23.91	18.45	40.55	55.52	60.37	86.67	70.88
sfdp	0.17	15.0	31.15	53.44	68.25	3.20	37.35	64.32	71.34	94.12	73.47

layout. In this section, we evaluate the recognition performance of different layouts, fixing other factors such as node shapes, node outline styles and edge thickness. The characteristics of each layout are summarized as follows:

- **dot** adopts a hierarchical structure, where nodes are arranged by level, and edges are arranged in the same direction (top to bottom, or left to right) to avoid edge crossings.
- **circo** uses a circular structure, placing nodes in a circle or ellipse, making it suitable for displaying cyclic structures or graphs with strong symmetry.
- **twopi** is a radial layout, arranging nodes around a central node in concentric circles. However, for non-radial structures, it may lead to edge overlaps.
- **neato** is based on a force-directed algorithm, using a spring-repulsion model (Kamada-Kawai algorithm). Nodes are compactly distributed in steps, and edge lengths are kept as consistent as possible. It performs well for small to medium-sized graphs, but short edge lengths may cause relationships to overlap.
- **fdp** is also force-directed, but with a simpler spring model. Nodes are distributed more loosely to cater for medium-sized graphs.
- **sfdp** is a multilevel and force-directed algorithm that efficiently layouts large graphs. he layout is more spread out with longer edges.

We use the ExplaGraphs dataset, altering the layout style of each KG while keeping all other settings fixed. At first, we perform manual filtering to remove images unclear to humans, ensuring that no characters are obstructed, and the triples for each layout are consistent, with only the visual format differing. Table 2 shows that there are significant discrepancies in the recognition performance for different layouts. For LLaVA-OV, InternVL2 and Qwen2VL, **the ranking is consistent: dot > circo > sfdp > fdp > neato > twopi**. As to GPT-4o, which has strong recognition capabilities, has a relatively low recognition accuracy on certain layouts such as twopi, neato and fdp. LLaVA has low triple recognition accuracy, with only about 1% accuracy for layouts other than dot. Therefore, a well-designed layout not only enhances readability by reducing cognitive load, but also helps improve the model's ability to recognize and process structural and semantic relationships within the graph.

Based on the summary of the characteristics of each layout, the dot layout's highest recognition accuracy suggests that all the models perform better in recognizing hierarchically structured triples. This may be attributed to the training dataset which includes flowchart data employing hierarchical layout, while lacks exposure to other layout formats. An example of different answers to the same question for different layouts is presented in Figure 4 and Appendix C. It can be observed that if the triples involved in the question are recognized correctly, the reasoning problem can also be answered correctly. To address this, we will further investigate the impact of incorporating data from other layouts for fine-tuning on recognition and reasoning tasks in Section 4. This aims to enhance the models' adaptability to diverse graph structures and improve their overall reasoning capabilities.

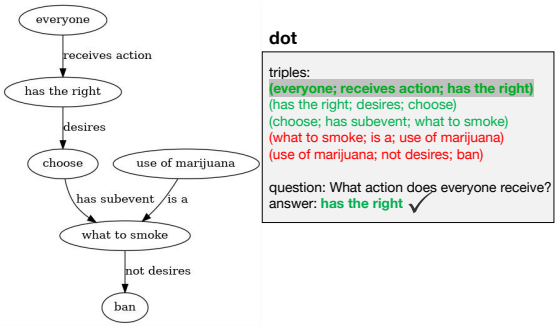
According to the results on the "Simple (given)" task for the Qwen2VL model, it is found that all layouts show improvements compared to the original reasoning without providing any triples. Furthermore, the inference accuracy of each layout reaches 70%, with less difference among them. This again emphasizes the role of triple recognition in the whole reasoning task.

4.4 Evaluation of Density

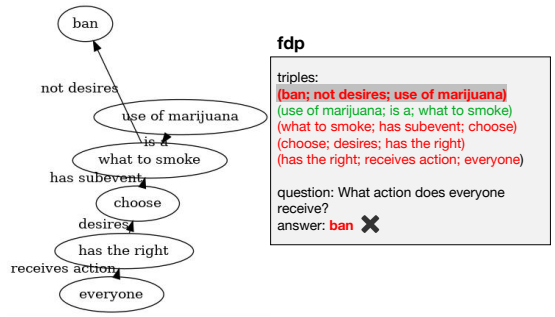
Since the construction of real-world KGs is more diverse and complex for the representation of different relationships, we also inspect the knowledge

Table 3: Triple recognition of complexity and density of knowledge graphs (Semantic: KG contains semantic Information, Non-Semantic: KG does not contain semantic information)

Dataset	ExplaGraphs		WebQSP-20		WebQSP-density	
Model	Semantic	Non-Semantic	Semantic	Non-Semantic	Semantic	Non-Semantic
LLaVA-OV	51.33	70.83	0.0	6.68	0.0	6.90
InternVL2	57.11	65.89	11.83	16.01	0.82	7.19
Qwen2VL	84.89	96.30	22.27	25.26	13.06	21.67
GPT-4o	93.33	95.68	20.25	34.31	13.26	38.33



(a) Layout of dot



(b) Layout of fdp

Figure 4: Example of different responses to the same question caused by different layouts. We color the correct and wrong responses in green and red.

graph with different edge densities. In order to better reflect the differences, we exclude the LLaVA model here, which performs poorly even on simple graphs. For the dataset, we use WebQSP, which expands the number of triples to 20 compared to the average value of 4.25 for the ExplaGraphs dataset. In addition, we create a variant named WebQSP-density, in which the number of triples is kept fixed, and the nodes with the highest sum of in-degree and out-degree in the graph are selected, sorting the top 20 edges in descending order. The results in Table 3 show that **as the number of triples and the density increase, the recognition rate of triples decreases for all models**. Interestingly, for high-density KGs, the LLaVA-OV model responds that the graph is a complex network diagram with various nodes and edges, making it almost impossible to recognize.

Moreover, to avoid obstruction caused by complex semantics in node and edge names in high-density triple images, we replaced the nodes and edges with simple characters for recognition. The issues encountered in complex graph recognition by Intern2VL and QwenVL2 include ignoring triples that are located far from the center of the image, and incorrectly identifying non-existent edges. This indicates that the visual encoders perform less effectively in handling edges in non-central regions when the graph's complexity is high.

4.5 Fine-tuning

In previous section, we demonstrate that graph recognition significantly affects reasoning capabilities. We aim to further utilize this finding to improve the model's performance in reasoning tasks by strengthening its recognition ability.

Data Preparation we constructed a dataset focusing on enhancing the recognition of graph triples, which contains six different layout types and 16.5k samples. Each layout consists of over 2.7k recognition samples. We created the instruction training set by using the prompt "Identify and list all the triples in the image" and pairing it with the triple recognition results from ExplaGraphs dataset, which isillustrated in Figure 5. For the fine-tuning baseline, we selected the Qwen2VL model, which performed the best in earlier evaluations. The training set for fine-tuning was constructed using the recognition-enhanced dataset, while the test set consists of simple reasoning problems.

Training Details The base model for fine-tuning is Qwen2-VL-7B-Instruct and the model was trained for 3 epochs with a LoRA rank of 8. Following the Stage 3 SFT setting in Qwen2VL, we locked the ViT parameters and performed exclusive fine-tuning of the LLM. The effective batch size was set to 8.

Table 4: Fine-tuned reasoning results with different fine-tuning strategies

	Qwen2VL	Qwen2VL_sft_self	Qwen2VL_sft_all	Qwen2VL_sft_dot
dot	71.19	**76.62**	74.79	**76.62**
circo	72.26	72.66	72.85	**77.73**
twopi	58.99	65.57	63.68	**69.02**
neato	60.92	**71.13**	66.56	70.81
fdp	60.37	**69.22**	65.73	67.48
sfdp	71.34	75.09	71.08	**76.21**

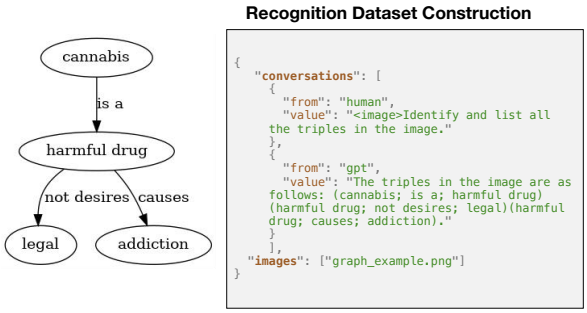


Figure 5: An example from the graph recognition enhancement dataset.

Graph Recognition Enhancement To fully leverage the constructed dataset and explore the fine-tuned model performance on each layout dataset, we implemented two distinct training strategies: layout-specific training and layout-mixed training.

1. **Layout-specific training**: The model is fine-tuned on each layout's specific recognition dataset respectively. Specifically, "Qwen2VL_sft_self" refers to fine-tuning on the same layout with test data, while "Qwen2VL_sft_dot" refers to fine-tuning on the layout of dot.

2. **Layout-mixed training**: The model is fine-tuned using a combination of all layout datasets, named as "Qwen2VL_sft_all".

As shown in Table 4, the results indicate that fine-tuning using the dot method across all layouts consistently yields even better performance than fine-tuning with the combined datasets. Even when the test data comes from other layouts (circo, twopi, or sfdp), the model fine-tuned with the dot layout still achieves optimal performance. This suggests that the hierarchical features of the dot layout are generalizable and can be transferred to other layouts. For some layouts, using the dot to fine-tune even outperforms directly providing the triples for reasoning. A case study of different fine-tune strategies is presented in the Appendix D. Additionally, fine-tuning the model on its own layout's dedicated dataset results in the highest performance for neato and fdp layout. This approach ensures that the model can adapt more specifically to the characteristics of each layout, enhancing its ability to recognize graph triples effectively in diverse scenarios. In contrast, fine-tuning with all layouts data does not achieve the best results on any single layout. The interference effect of mixed training, caused by feature dilution and conflicts, forces the model to learn visual patterns from multiple layouts simultaneously. The significant structural differences between layouts, such as the hierarchy of dot, the circular structure of circo, and the force-directed nature of neato, make it difficult for the model to focus on core semantics. For KGs in real-world scenarios, there may exist different layout features that cannot be mapped to a standardized layout. Therefore, the use of layout-specific dot is suitable and can achieve optimal results in open scenarios.

5 Conclusion

In this work, we analyze the ability of MLLMs to understand KG data in multiple dimensions. Our findings indicate that the current MLLMs have limitations in graph understanding, strongly attributed to their poor recognition ability, and the layout and density of the images play a crucial role in recognition. Through fine-tuning experiments, we confirm that fine-tuning on triple question-answer pairs with different layouts can improve their reasoning performance. In the future work, we will advance this work from the following two directions: (1) Designing layouts that are more easily to understand by MLLMs for large-scale and complex graphs, or designing dedicated image encoders to capture the pixels related to nodes; (2) Exploring how to incorporate internal knowledge within the model to reduce hallucinations and erroneous information based on graph understanding. By improving the recognition ability of MLLMs, there will be greater potential for advancements in graph understanding in the future.

6 Limitation

MLLMs that we evaluate are not complete In our experiments, we focus on evaluating the graph understanding abilities of the following models: LLaVA, LLaVA-OV, InternVL2, Qwen2VL, and GPT-4o. These models are selected based on their advanced capabilities in multimodal tasks and their potential to handle graph-based data. However, we acknowledge that there are several other open-source MLLMs that could also contribute valuable insights into graph understanding. We leave it to future work on evaluating the graph reasoning abilities of other models.

Methods for improving graph reasoning abilities In this paper, we only provide a fine-tuning solution for the layout, which works on easy graph reasoning problems. However, there is still much room for improvement when it comes to more complex graph models and reasoning tasks, where adjustments to the visual encoder could be made.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Lakshita Agarwal and Bindu Verma. 2024. From methods to datasets: A survey on image-caption generators. *Multim. Tools Appl.*, 83(9):28077–28123.

Qihang Ai, Jiafan Li, Jincheng Dai, Jianwu Zhou, Lemao Liu, Haiyun Jiang, and Shuming Shi. 2024. Advancement in graph understanding: A multimodal benchmark and fine-tuning of vision-language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 7485–7501. Association for Computational Linguistics.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul Ronald Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, and et al. 2023. Gemini: A family of highly capable multimodal models. *CoRR*, abs/2312.11805.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Tianyi Bai, Hao Liang, Binwang Wan, Ling Yang, Bozhou Li, Yifan Wang, Bin Cui, Conghui He, Binhang Yuan, and Wentao Zhang. 2024. A survey of multimodal large language model from A data-centric perspective. *CoRR*, abs/2405.16640.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. 2024a. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *Science China Information Sciences*, 67(12):220101.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. 2024b. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Xinbang Dai, Yuncheng Hua, Tongtong Wu, Yang Sheng, and Guilin Qi. 2024. Counter-intuitive: Large language models can better understand knowledge graphs than we thought. *CoRR*, abs/2402.11541.

Alexey Dosovitskiy. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Mohammed Elhenawy, Ahmed Abdelhay, Taqwa I. Alhadidi, Huthaifa I Ashqar, Shadi Jaradat, Ahmed Jaber, Sebastien Glaser, and Andry Rakotonirainy. 2024. Eyeballing combinatorial problems: A case study of using multimodal large language models to solve traveling salesman problems. *Preprint*, arXiv:2406.06865.

Chaoyou Fu, Yifan Zhang, Shukang Yin, Bo Li, Xinyu Fang, Sirui Zhao, Haodong Duan, Xing Sun, Ziwei Liu, Liang Wang, Caifeng Shan, and Ran He. 2024. Mme-survey: A comprehensive survey on evaluation of multimodal llms. *CoRR*, abs/2411.15296.

Emden R. Gansner and Stephen C. North. 2000. An open graph visualization system and its applications to software engineering. *Softw. Pract. Exp.*, 30(11):1203–1233.

Jiayan Guo, Lun Du, Hengyu Liu, Mengyu Zhou, Xinyi He, and Shi Han. 2023. Gpt4graph: Can large language models understand graph structured data ? an empirical evaluation and benchmarking. *Preprint*, arXiv:2305.15066.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. 2024a. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zheqi He, Xinya Wu, Pengfei Zhou, Richeng Xuan, Guang Liu, Xi Yang, Qiannan Zhu, and Hua Huang. 2024b. CMMU: A benchmark for chinese multi-modal multi-type question understanding and reasoning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*, pages 830–838. ijcai.org.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. 2024. Rotary position embedding for vision transformer. In *European Conference on Computer Vision*, pages 289–305. Springer.

Junlin Lee, Yequan Wang, Jing Li, and Min Zhang. 2024. Multimodal reasoning with multimodal knowledge graph. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10767–10782, Bangkok, Thailand. Association for Computational Linguistics.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. 2024a. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

Xin Li, Weize Chen, Qizhi Chu, Haopeng Li, Zhaojun Sun, Ran Li, Chen Qian, Yiwei Wei, Chuan Shi, Zhiyuan Liu, Maosong Sun, and Cheng Yang. 2024b. Can large language models analyze graphs like professionals? a benchmark, datasets and models. In *Advances in Neural Information Processing Systems*, volume 37, pages 141045–141070. Curran Associates, Inc.

Yuhan Li, Zhixun Li, Peisong Wang, Jia Li, Xiangguo Sun, Hong Cheng, and Jeffrey Xu Yu. 2024c. A survey of graph meets large language model: Progress and future directions. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*, pages 8123–8131. ijcai.org.

Yuhan Li, Peisong Wang, Xiao Zhu, Aochuan Chen, Haiyun Jiang, Deng Cai, Wai Kin (Victor) Chan, and Jia Li. 2024d. Glbench: A comprehensive benchmark for graph with large language models. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Yunxin Li, Baotian Hu, Haoyuan Shi, Wei Wang, Longyue Wang, and Min Zhang. 2024e. Visiongraph: Leveraging large multimodal models for graph theory problems in visual context. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024b. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. 2024c. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. 2024. Deepseek-vl: Towards real-world vision-language understanding. *CoRR*, abs/2403.05525.

Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. 2024. Vision language models are blind[inline-graphic not available: see fulltext]. In *Computer Vision - ACCV 2024 - 17th Asian Conference on Computer Vision, Hanoi, Vietnam, December 8-12, 2024, Proceedings, Part V*, volume 15476 of *Lecture Notes in Computer Science*, pages 293–309. Springer.

Swarnadeep Saha, Prateek Yadav, Lisa Bauer, and Mohit Bansal. 2021. Explagraphs: An explanation graph generation task for structured commonsense reasoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 7716–7740. Association for Computational Linguistics.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Shagun Uppal, Sarthak Bhagat, Devamanyu Hazarika, Navonil Majumder, Soujanya Poria, Roger Zimmermann, and Amir Zadeh. 2022. Multimodal research in vision and language: A review of current and emerging trends. *Information Fusion*, 77:149–171.

J Vaishnavi and V Narmatha. 2024. Video captioning–a survey. *Multimedia Tools and Applications*, pages 1–32.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

Heng Wang, Shangbin Feng, Tianxing He, Zhaoxuan Tan, Xiaochuang Han, and Yulia Tsvetkov. 2023. Can language models solve graph problems in natural language? In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024a. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. 2024b. Internvideo2: Scaling foundation models for multimodal video understanding. In *European Conference on Computer Vision*, pages 396–416. Springer.

Yiqi Wang, Wentao Chen, Xiaotian Han, Xudong Lin, Haiteng Zhao, Yongfei Liu, Bohan Zhai, Jianbo Yuan, Quanzeng You, and Hongxia Yang. 2024c. Exploring the reasoning abilities of multimodal large language models (mllms): A comprehensive survey on emerging trends in multimodal reasoning. *arXiv preprint arXiv:2401.06805*.

Yanbin Wei, Shuai Fu, Weisen Jiang, Zejian Zhang, Zhixiong Zeng, Qi Wu, James T. Kwok, and Yu Zhang. 2024. GITA: graph to visual and textual integration for vision-language graph reasoning. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Yibo Yan and Joey Lee. 2024. Georeasoner: Reasoning on geospatially grounded context for natural language understanding. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 4163–4167.

Yibo Yan, Jiamin Su, Jianxiang He, Fangteng Fu, Xu Zheng, Yuanhuiyi Lyu, Kun Wang, Shen Wang, Qingsong Wen, and Xuming Hu. 2024. A survey of mathematical reasoning in the era of multimodal large language model: Benchmark, method & challenges. *CoRR*, abs/2412.11936.

Ruosong Ye, Caiqi Zhang, Runhui Wang, Shuyuan Xu, and Yongfeng Zhang. 2023. Natural language is all a graph needs. *Preprint*, arXiv:2308.07134.

Wen-tau Yih, Matthew Richardson, Christopher Meek, Ming-Wei Chang, and Jina Suh. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. The Association for Computer Linguistics.

Wenqi Zhang, Zhenglin Cheng, Yuanyu He, Mengna Wang, Yongliang Shen, Zeqi Tan, Guiyang Hou, Mingqian He, Yanna Ma, Weiming Lu, and Yueting Zhuang. 2024. Multimodal self-instruct: Synthetic abstract image and visual reasoning instruction using language model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 19228–19252. Association for Computational Linguistics.

Jianan Zhao, Le Zhuo, Yikang Shen, Meng Qu, Kai Liu, Michael M. Bronstein, Zhaocheng Zhu, and Jian Tang. 2023. Graphtext: Graph reasoning in text space. *CoRR*, abs/2310.01089.

Jie Zhao, Kang Hao Cheong, and Witold Pedrycz. 2025. Bridging visualization and optimization: Multimodal large language models on graph-structured combinatorial optimization. *Preprint*, arXiv:2501.11968.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

A Dataset Example

ExplaGraphs (Figure 6) is used in Section 4.2, WebQSP and its variations (Figure 7, 8, 9, 10) are used in Section 4.4.

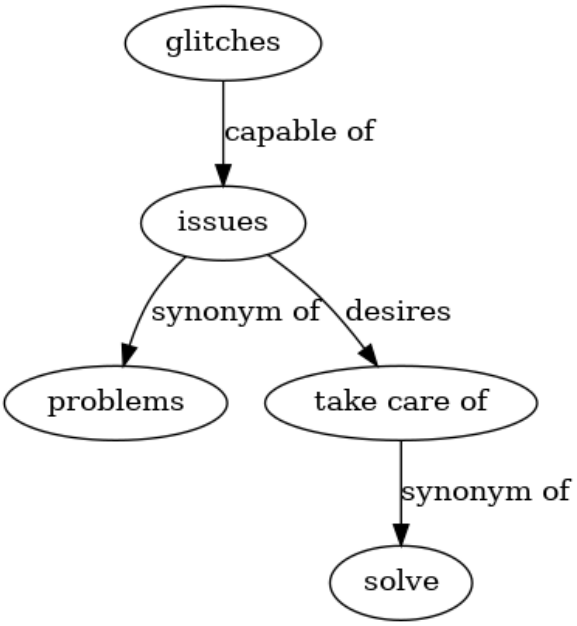


Figure 6: An example of ExplaGraphs

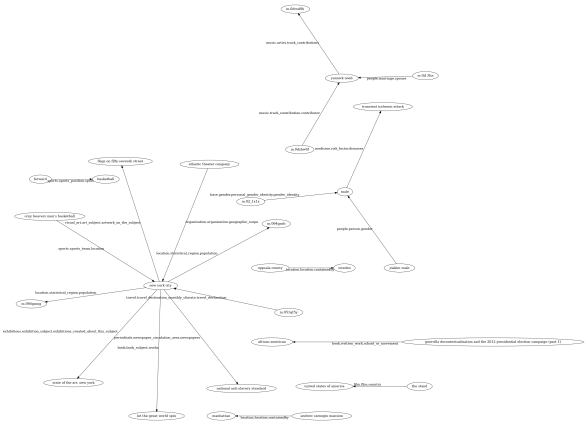


Figure 7: An example of WebQSP-20

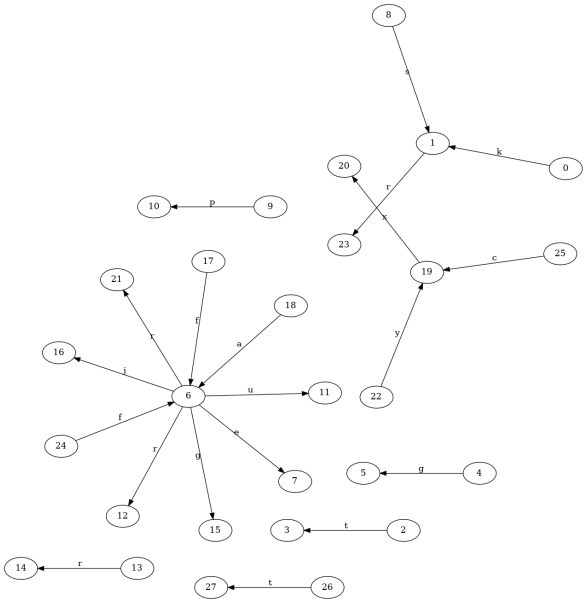


Figure 8: An example of WebQSP-20 without Semantic

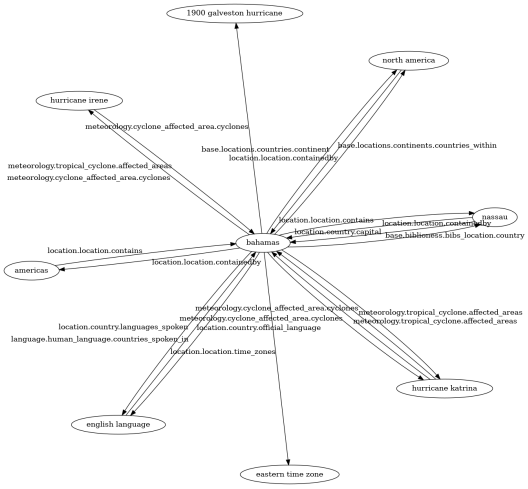


Figure 9: An example of WebQSP-density

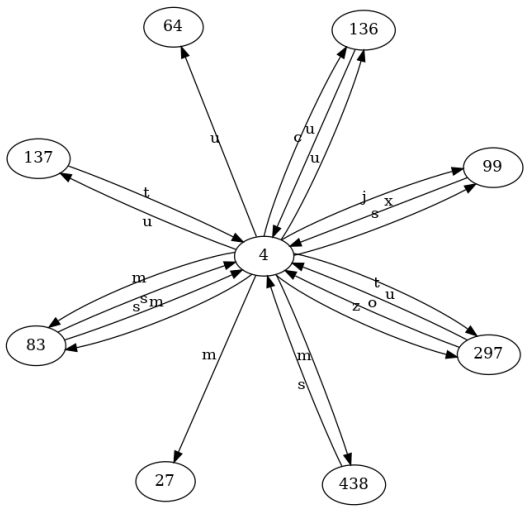


Figure 10: An example of WebQSP-density without Semantic

B Prompt List

All the prompt inputs used in this paper are listed in the Table 5.

Table 5: Prompt input for different tasks

Task	Evaluation	Example prompt
Step-by-step Recognition Evaluation	node	Please analyze the image and list all the entities (nodes) represented in the diagram.
	node pairs	Please list the triples without edges based on the knowledge graph structure in the picture, in the format (node1, node2),(node3, node4). Ignore the edge relationships.
	node pairs(given)	The nodes in the picture is cannabis,legal,marijuana,good thing,marijuana,more available,good thing,legal. Please list the node pairs without edges based on the knowledge graph structure in the picture, in the format (node1, node2) (node3, node4). Ignore the edge relationships.
	triple	Identify and extract all nodes and edges from the image in the form of triples (node1; relation; node2).
	triple(given)	The nodes in the picture are cannabis,legal,marijuana,good thing,marijuana,more available,good thing,legal,the relations in the picture are synonym of, causes, capable of, desires. Please output all triples from the image based on the provided nodes and relations in the format (node1; relation; node2).
Reasoning	simple	What property does cannabis have?
	complex	What is the relationship between cannabis and medicinal purposes?
	simple(given)	The triples in the image is:(cannabis; synonym of; marijuana)(legal; causes; more available)(marijuana; capable of; good thing)(good thing; desires; legal),Please answer the questions based on the triples:What is a synonym of cannabis?

C Layout Example

Different layouts provide different answers to the same question, "What action does everyone receive?" As shown in the Figure 4(a),4(b),11,12,13,14, we demonstrate how various layouts perform in terms of triple recognition and reasoning. Green highlights the correct triple and Red highlights the wrong triple. Gray and Bold represents the target triplet. It can be observed that if a layout correctly recognizes the triples involved in the question, it also provides an accurate answer. However, layouts of neato fail to recognize all triples while fdp incorrectly identifies the triples related to the question, leading to erroneous responses.

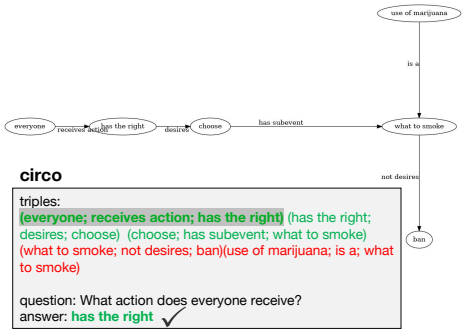


Figure 11: An example of circo layout

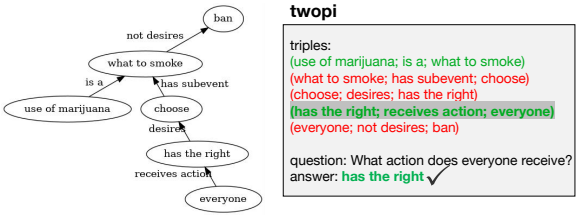


Figure 12: An example of twopi layout

D Case Study

This is an example of a graph image of neato layout. All fine-tuned models can correctly identify the triples. According to the response, sft_all and sft_neato models confuse the edges and nodes, leading to incorrect results. On the other hand, the sft_dot model has a better understanding of the relationships within the triples and provides correct answers.

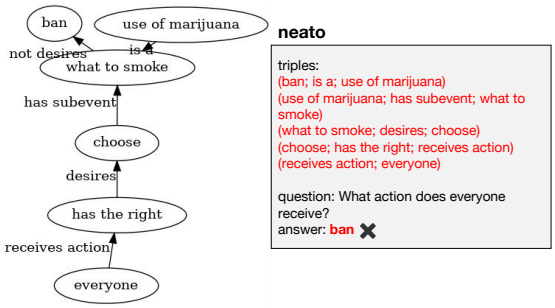


Figure 13: An example of neato layout

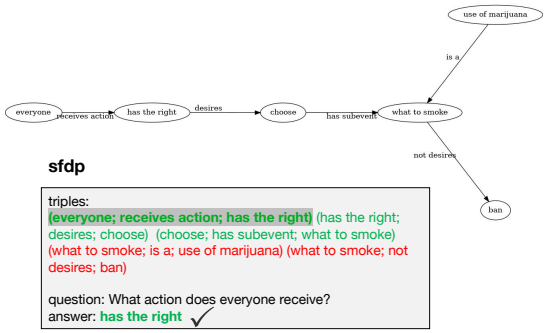


Figure 14: An example of sfdp layout

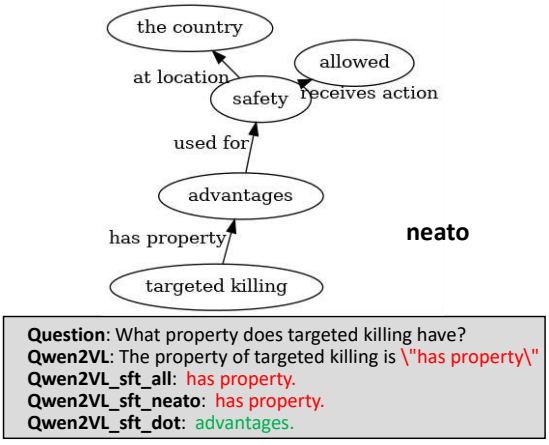


Figure 15: A case study of fine-tune model