GraphVis: Boosting LLMs with Visual Knowledge Graph Integration

Yihe Deng* **Chenchen Ye** **Zijie Huang**
Mingyu Derek Ma **Yiwen Kou** **Wei Wang**

University of California, Los Angeles

Abstract

The rapid evolution of large language models (LLMs) has expanded their capabilities across various data modalities, extending from well-established image data to increasingly popular graph data. Given the limitation of LLMs in hallucinations and inaccuracies in recalling factual knowledge, Knowledge Graph (KG) has emerged as a crucial data modality to support more accurate reasoning by LLMs. However, integrating structured knowledge from KGs into LLMs remains challenging, as most KG-enhanced LLM methods directly convert the KG into linearized text triples, which is not as expressive as the original structured data. To address this, we introduce `GraphVis`, which conserves the intricate graph structure through the visual modality to enhance the comprehension of KGs with the aid of Large Vision Language Models (LVLMs). Our approach incorporates a unique curriculum fine-tuning scheme which first instructs LVLMs to recognize basic graphical features from the images, and subsequently incorporates reasoning on QA tasks with the visual graphs. This cross-modal methodology not only markedly enhances performance on standard textual QA but also shows improved zero-shot VQA performance by utilizing synthetic graph images to augment the data for VQA tasks. We present comprehensive evaluations across commonsense reasoning QA benchmarks, where `GraphVis` provides an average improvement of 11.1% over its base model and outperforms existing KG-enhanced LLM approaches. Across VQA benchmarks such as ScienceQA that share similar scientific diagram images, `GraphVis` provides a notable gain of 4.32%. Code is made available on GitHub.

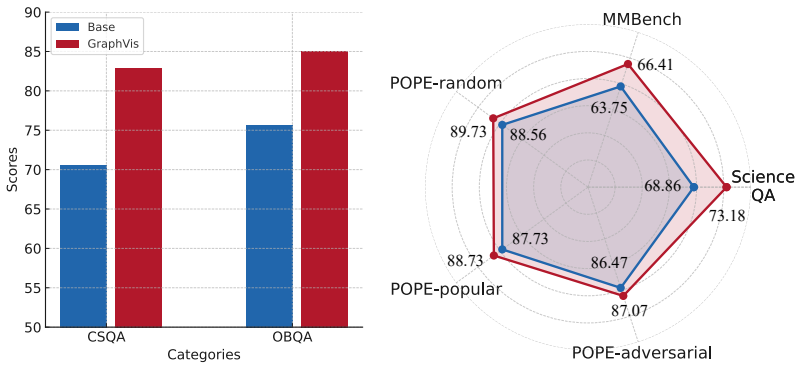


Figure 1: **Left**: Accuracy improvement of `GraphVis` compared to the base model's performance on commonsense reasoning tasks. **Right**: Improvement by `GraphVis` on multiple VQA benchmarks over its base LVLM model LLaVA-v1.6 (Liu et al., 2024).

*Corresponding to Yihe Deng <yihedeng@cs.ucla.edu>.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-2155

1 Introduction

The rapid evolution of large language models (LLMs) (Zhang et al., 2019; Brown et al., 2020; Touvron et al., 2023a; Chung et al., 2024) has unlocked new opportunities for interacting with multimodal data sources. Approaches that enable input from multi-modal data can expands the information that LLMs can take in for various downstream reasoning tasks across domains. The modalities in existing LLM-based models span across images (Zhu et al., 2023; Liu et al., 2023b), videos (Maaz et al., 2023; Li et al., 2023c), and audio (Zhang et al., 2023; Rubenstein et al., 2023). Most recently, researchers have also begun to build an unified architecture to encode diverse modalities jointly (Wu et al., 2023). Such unification holds considerable promise, and poses an interesting question on whether data from one modality could enhance the model performance in another.

Beyond the frequently explored modalities such as vision and audio, knowledge graphs (KGs) are also gaining attention. Given LLMs' limitations such as hallucinations (Li et al., 2023a), inaccuracies in recalling factual knowledge (Yang et al., 2023), and the costly updates of knowledge via training (Ding et al., 2023), researchers are exploring KGs as a robust source of structured and easy-to-update facts (Pan et al., 2023; Jin et al., 2023; Agrawal et al., 2023; Huang et al., 2023). The use of KGs to enhance language models began with smaller models like BERT (Kenton and Toutanova, 2019; Huang et al., 2022), incorporating KGs into the pre-training objectives (Zhang et al., 2019; Rosset et al., 2020; Wang et al., 2021) or integrating them through architectural modifications (Yasunaga et al., 2021; Zhang et al., 2022). However, the recent development of larger and more complex LLMs poses challenges in adapting these earlier methods. Current strategies for integrating KGs into LLMs fall into two categories: (1) verbalizing relevant KG triples and directly appending them to prompts as "(node a, edge, node b)" (Guo et al., 2023; Feng et al., 2023; Baek et al., 2023) or (2) employing a graph neural network (GNN) to generate embeddings for relevant KG subgraphs and projecting these into the LLM's token embedding space (Chai et al., 2023; Tian et al., 2024). Nonetheless, these approaches often yield results that are either weaker than or comparable to methods that fully fine-tune smaller LMs with integrated KG information, revealing an underutilization of the graph structure and rich relational context. Thus, effectively integrating the KG modality into LLMs and enabling them to comprehend graph concepts remains an unsolved challenge.

With the rapid advancement of LLMs across various modalities, a question arises: can multi-modal LLMs, trained in domains other than KGs, facilitate the understanding of graph structures? Large Vision Language Models (LVLMs) (Liu et al., 2023b), pre-trained on an extensive corpus of image-text pairs, demonstrate exceptional abilities in processing image inputs. In response to this potential, we introduce a novel methodology, `GraphVis`, that enhances graph comprehension by visualizing subgraphs and leveraging LVLMs for KG-enhanced question answering. This approach involves translating retrieved subgraphs into images, which are then processed by an LVLM to aid in answering questions. Recognizing that LVLMs typically lack proficiency with visual graphs, we design a unique curriculum fine-tuning scheme. Initially, the model is trained to interpret simple graphical features, such as node count, edge count, and node degree, through self-supervised learning. It then progresses to handling more complex queries by integrating textual question-answer data with relevant visualized subgraphs, thereby fine-tuning the LVLM to respond accurately to KG-based questions using these images. Our experiments demonstrate that this approach effectively improves the model's performance on downstream QA tasks, surpassing both current KG-enhanced LLMs and traditional fully fine-tuned KG language models.

While the vision modality significantly enhances the integration of KGs and improves the performance on textual QA tasks, our study extends this exploration to the benefits of KGs and textual QA data for LVLMs in zero-shot visual question-answering (VQA) tasks. Notably, images resembling graphs are prevalent in current VQA tasks (Lu et al., 2022; Yu et al., 2023a; Lu et al., 2024), yet similar training datasets are scarce. Our research demonstrates that the availability of extensive textual QA data and relevant KGs facilitates the generation of large synthetic datasets that feature graph images, effectively addressing this scarcity and supporting the training of LVLMs on such data. Evaluations across multiple VQA benchmarks reveal that our LVLM, fine-tuned with the `GraphVis` approach, also shows remarkable improvements in VQA performance.

Our contributions are summarized as follows:

- We introduce a novel method, `GraphVis`, that employs visual modality to enhance the understanding of KGs in LLMs, leveraging graph visualization to bridge the gap between structured KG data and multimodal LLM processing capabilities.

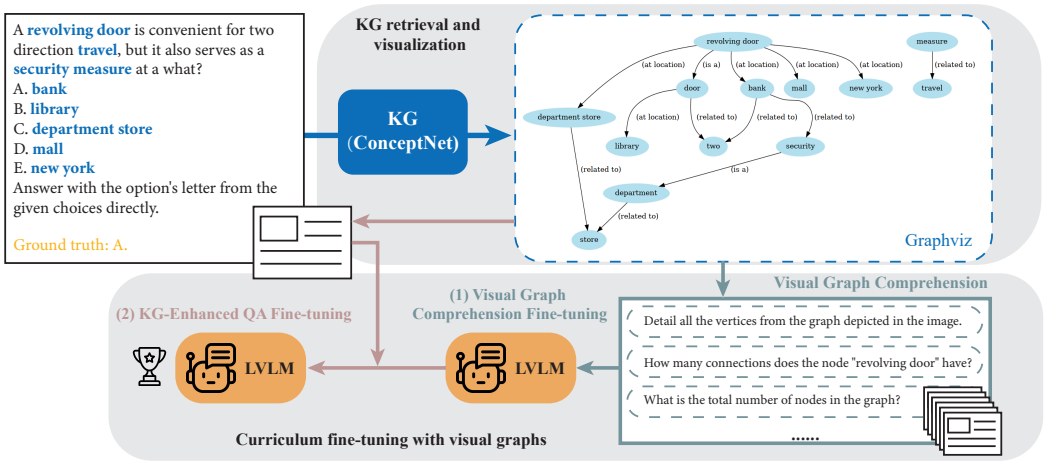


Figure 2: Overview of `GraphVis`. Given an input question and answer pair in the training data, we retrieve and visualize the relevant subgraph. With pre-defined questions on the basic features of the visual graphs such as numbers of nodes and node degree, we first construct data for visual graph comprehension fine-tuning. Subsequently, we incorporate the QA pair with the visual graph for KG-enhanced QA fine-tuning.

- We present a unique curriculum fine-tuning scheme tailored for LVLMs that sequentially trains on graph-derived images first to visual graph comprehension and then to apply this understanding in more complex QA contexts.
- We offer a new perspective on gathering fine-tuning data to enhance LVLMs. Specifically, we propose that pure textual data can be combined with relevant synthetic graph images derived from KGs to improve the LVLM's capability in image comprehension and reasoning.

2 Related Work

KG-Enhanced LLMs. Initial studies on KG-enhanced language models have shown that integrating KGs into the pre-training objectives can enrich the foundational knowledge of language models. This approach has been largely applied to encoder-only language models such as BERT (Kenton and Toutanova, 2019) with training objectives specifically tailored for these models such as masked word prediction (Zhang et al., 2019; Shen et al., 2020; Zhang et al., 2020; Rosset et al., 2020; Wang et al., 2021; Li et al., 2022; Kang et al., 2022; Baek et al., 2023). Another line of work also relies on the encoder architecture of language models and performs full-parameter fine-tuning on a KG encoder for the fusion of knowledge (Sun et al., 2021; Yasunaga et al., 2021; Zhang et al., 2022; Yujie et al., 2023). However, the advent of recent decoder-only LLM pre-trained on a significantly larger scale (e.g., the GPT series (Radford et al., 2019; Brown et al., 2020; OpenAI, 2023), LLaMA series (Touvron et al., 2023a,b) and Mistral (Jiang et al., 2023)), increases the difficulty and cost of adapting these KG-based pre-training and fine-tuning methods to current LLMs. Consequently, with KGs as a distinct modality, researchers have been exploring various methods for integrating the information into LLMs. One straightforward and most commonly used approach involves verbalizing relevant knowledge graphs and appending them to the prompts (Guo et al., 2023; Feng et al., 2023; Fatemi et al., 2023; Sun et al., 2023; Luo et al., 2023). For a notable example, KAPING (Baek et al., 2023) retrieves the top k most relevant knowledge triples to the prompt and appends it in the form of textual triples to the original prompt. Meanwhile, such an approach linearizes the originally structured information and does not maintain a natural language form. Another research direction therefore employs GNNs to generate embeddings for retrieved subgraphs, subsequently projecting these into the LLM's token embedding space as soft prompts to preserve structured graph information (Yasunaga et al., 2021; Hu et al., 2022; Zhang et al., 2022; Chai et al., 2023; Tian et al., 2024). Nevertheless, GNN-based approaches require task-specific fine-tuning and may struggle with generalization across new tasks.

Multimodal LLMs. Other than incorporating knowledge graphs, popular investigations on multi-modal inputs to LLMs include image (Zhu et al., 2023; Liu et al., 2023b), video (Maaz et al., 2023; Li et al., 2023c), audio (Zhang et al., 2023; Rubenstein et al., 2023) and temporal data (Yu et al., 2023b; Chang et al., 2023). Significantly, advances in pre-trained vision-language models (Radford et al., 2021; Jia et al., 2021; Alayrac et al., 2022), which align the visual and textual embedding spaces on web-scale image-caption data, have facilitated substantial progress in the development of Large

https://doi.org/10.52202/079017-2155

Vision Language Models (LVLMs) (Liu et al., 2023a,b; Zhu et al., 2023; Chen et al., 2023; Ye et al., 2023; Dai et al., 2023; Gao et al., 2023; Bai et al., 2023; Peng et al., 2023). These models, with vision encoders trained on extensive collections of web images, exhibit robust visual reasoning capabilities across a range of tasks (Gao et al., 2015; Lu et al., 2022; Xu et al., 2023; Lu et al., 2024). However, the incorporation of image data with graph structures into both pre-training and benchmark datasets remains limited, primarily appearing as scientific diagrams within visual question answering datasets for mathematics and science (Lu et al., 2022, 2024). This paper also sheds light on an interesting potential to acquire a large volume of graph images through text-based QA datasets to enhance the capabilities of LVLMs.

3 Problem Setting and Preliminaries

Notation. We use lower case letters to denote scalars and lower case bold face letters to denote vectors. We denote an input sequence, or prompt, as $\mathbf{x} = [x_1, \ldots, x_n]$, where x_i represents a token in the LLM's vocabulary. Then, we use the symbol $p(\cdot|\mathbf{x})$ to represent the conditional probability of LLM's response given the prompt $\mathbf{x}$. Lastly, we denote the sequence of tokens generated before the t-th token as $\mathbf{y}_{<t} = [y_1, \ldots, y_{t-1}]$ for $t > 1$.

Generative Language Models. Let $p_{\boldsymbol{\theta}}$ denotes an LLM parameterized by $\boldsymbol{\theta}$. We consider a sequence $\mathbf{x} = [x_1, \ldots, x_n]$ as the input prompt, for which each x_i is a token from the LLM's vocabulary. The LLM then generates the response sequence $\mathbf{y} = [y_1, \ldots, y_m]$ by sampling from the conditional probability distribution $p_{\boldsymbol{\theta}}(\cdot|\mathbf{x})$, where y_t denotes individual token for $1 \leq t \leq m$. The conditional distribution $p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})$ can therefore be expressed as a Markov process $p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x}) = \prod_{t=1}^{m} p_{\boldsymbol{\theta}}(y_t|\mathbf{x}, \mathbf{y}_{<t})$. Given a supervised fine-tuning dataset, $S = \{(\mathbf{x}, \mathbf{y})\}_{i=1}^{n}$, the training objective is therefore to maximize the model's likelihood of generating $\mathbf{y}$ given $\mathbf{x}$, resulting in the following loss function:

$$L(\boldsymbol{\theta}) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim S}\Big[-\log p_{\boldsymbol{\theta}}\big(\mathbf{y}|\mathbf{x}\big)\Big]. \tag{3.1}$$

Given an LLM, an LVLM additionally contains two more components, including a vision encoder $f_v(\cdot)$ and a projection network $f_p(\cdot)$. The model processes an additional image input $\mathbf{e}$, which is converted into visual tokens within the language token space by the vision encoder and the projection network, producing $\mathbf{v} = [v_1, \ldots, v_k] = f_v \circ f_p(\mathbf{e})$. The conditional probability distribution $p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{v}, \mathbf{x})$ is thus decomposed as

$$p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{v}, \mathbf{x}) = \prod_{j=1}^{m} p_{\boldsymbol{\theta}}(y_j|\mathbf{v}, \mathbf{x}, \mathbf{y}_{<j}). \tag{3.2}$$

KG-enhanced LLMs. A knowledge graph, denoted as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, consists of a set of vertices $\mathcal{V}$ and their connections, or edges, $\mathcal{E}$. Considering an input question $\mathbf{x} = [x_1, \ldots, x_n]$ with its corresponding ground truth answer $\mathbf{y}^*$, we define $\mathcal{V}_{\mathbf{x}} = \{\mathbf{v}_i\}_{i\in\mathcal{I}_{\mathbf{x}}} \subseteq \mathcal{V}$ as the vertices mentioned in $\mathbf{x}$, where $\mathcal{I}_{\mathbf{x}}$ is the index set of vertices associated with the tokens in the question. The objective of KG-enhanced LLM can be decomposed into two steps: (1) relevant subgraph retrieval and (2) effective knowledge projection to the language embedding space. Subgraph retrieval involves designing a function f that generates a subgraph most relevant to the input prompt and containing the mentioned vertices $\mathcal{V}_{\mathbf{x}}$ and connected via the edges $\mathcal{E}_{\mathbf{x}}$:

$$f(\mathbf{x}, \mathcal{V}_{\mathbf{x}}, \mathcal{G}) = \{\mathcal{V}_{\mathbf{x}}, \mathcal{E}_{\mathbf{x}}\} = \mathcal{G}_{\mathbf{x}} \subset \mathcal{G}.$$

The function f could be pre-defined or trained. In this work, we consider the same approach as previous works (Feng et al., 2020), where $\mathcal{E}_{\mathbf{x}}$ is obtained from all k-hop paths connecting two nodes in $\mathcal{V}_{\mathbf{x}}$. Given a relevant subgraph $\mathcal{G}_{\mathbf{x}}$, the target of effectively leveraging the information is to construct a function g that generates informative tokens such that

$$p_{\boldsymbol{\theta}}(\mathbf{y}^*|\mathbf{x}^g) = \max_{\mathbf{x}} p_{\boldsymbol{\theta}}(\mathbf{y}^*|\mathbf{x}),$$

where $\mathbf{x}^g = [g(\mathcal{G}_{\mathbf{x}}), \mathbf{x}]$ is the KG-augmented prompt. In essence, the function g finds a way of leveraging the knowledge graph to enhance the language model's capacity for answering questions. The current methods therefore fall into the framework as

- Linearize. The linearization process is to represent the KG as a list of triples: $g(\mathcal{G}_{\mathbf{x}}) = [(\mathbf{v}_1, \mathbf{e}_1, \mathbf{u}_1), (\mathbf{v}_2, \mathbf{e}_2, \mathbf{u}_2), \cdots]$ where edge $\mathbf{e}_i \in \mathcal{E}_{\mathbf{x}}$ and $\mathbf{v}_i, \mathbf{u}_i$ are two endpoints of $\mathbf{e}_i$.
- GNN-based. The GNN-based methods leverage a GNN model for the additional information: $\mathbf{x}^g = \big[g_{\text{GNN}.}(\mathcal{G}_{\mathbf{x}}), \mathbf{x}\big]$.

4 Method

In this section, we formally introduce GraphVis, a technique that employs LVLMs to enhance the integration of KG information, thereby improving performance in downstream textual QA tasks. Reversely, GraphVis also enhances the performance of LVLMs in visual QA tasks by utilizing extensive data from both textual and KG modalities. The methodology of GraphVis is outlined in Algorithm 1 and demonstrated in Figure 2. We further elaborate the details of the method below.

GraphVis consists of two major components: (1) a novel integration of the retrieved subgraph for KG-enhanced QA via visualization of the graph, and (2) a progressive fine-tuning approach that starts by understanding graphical features and subsequently leverages them for reasoning. The primary objective of GraphVis is to improve the incorporation of KG information into LVLMs rather than enhancing retrieval techniques. Therefore, we adopt the same subgraph retrieval approach as previous studies (Lin et al., 2019; Feng et al., 2020; Yasunaga et al., 2021), which involves retrieving k-hop paths between entities mentioned in the input prompts from the entire KG. For visualization, we utilize the Graphviz tool (Gansner and North, 2000) to generate visual representations for each retrieved KG subgraph.

Most importantly, GraphVis employs a unique curriculum fine-tuning approach specifically designed for visual graph comprehension. While current LVLMs are fine-tuned on human-labeled vision-language instruction data, images of complex graph structures are much more scarce compared to the many natural images. The reasoning tasks designed for complex graph images are also very limited. GraphVis highlights the potential to leverage textual data and KG images to improve the LVLM's capability in reasoning with graph images. To address this, we initiate the fine-tuning process with simple, self-constructed questions about the structural and relational information in the graph, paired with automatically derived answers, training the model to thoroughly understand visual graphs before progressing to more complex reasoning tasks. The loss objective remains the same as SFT objective (3.1). These questions include,

- **Node description**: name all nodes appeared in the image.
- **Node degree detection**: answer with the degree of a named node in the image.
- **Highest node degree detection**: answer with name(s) of the node(s) that has the highest degree in the image.
- **Node number detection**: answer with the total number of nodes appeared in the image.
- **Edge number detection**: answer with the total number of edges appeared in the image.
- **Triple listing**: describe the image by listing all triples that appeared in the image.

For each of the question types, we draw a prompt from a pool of five pre-defined prompts of the task to add variance to the data. After the model fully understands the features of a visual graph, we proceed to further fine-tune its ability to reason with the visual graph, enhancing its capability to respond to related queries. The original question from the textual QA training dataset is then augmented with the visual subgraph as the following,

```
<visual subgraph>
The image represents a knowledge graph relevant to the question,
which may or may not be useful.  Question:  <original question>
```

The ground truth answers remain unchanged from the textual QA training data. This KG-enhanced QA fine-tuning subsequently starts from the model weights learned in the previous visual graph comprehension fine-tuning phase.

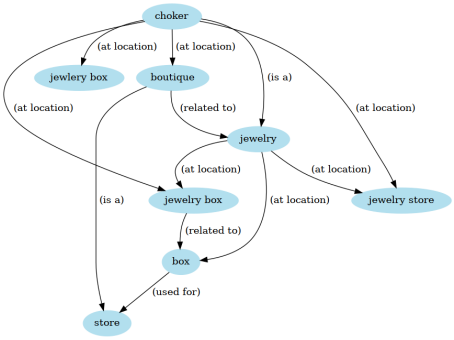
5 Experiments

In this section, we present experiment results of GraphVis on enhancing commonsense reasoning tasks with retrieved KG subgraphs from ConceptNet, as well as improving the zero-shot VQA capability of the LVLM by leveraging the data from the textual and KG modality. Across several benchmark datasets, we demonstrate the effectiveness of GraphVis.

5.1 Experiment Setup

Model and Datasets. In experiments, we consider `llava-v1.6-mistral-7b` (Liu et al., 2023a) as our base VLM model. We consider ConceptNet (Speer et al., 2017), a commonsense knowledge graph, as the KG used in our experiments. There are 799,273 nodes and 2,487,810 edges in total existing in the KG, and there are 42 specific different types of relations, merged into 17 relations (Feng et al., 2020). In both fine-tuning stage and inference stage, we consider retrieving 2-hop subgraphs for the conciseness of the images while preserving important information. We then

https://doi.org/10.52202/079017-2155



Detail all the vertices from the graph depicted in the image.
The image depicts the following nodes: store, box, choker, boutique, jewelry store, jewelry box, jewlery box, jewelry.

Identify one of the node that has the most connections in the graph and specify its degree.
One node with the highest degree is "jewelry" with a degree of 5.

What is the degree of the node with the name "jewlery box"?
The degree of the node "jewlery box" is 1.

How many nodes are there in the graph?
There are 8 nodes in the graph.

How many connections are there in the graph?
There are 12 edges in the graph.

Enumerate all the triples in the graph.
The triples in the graph are listed as: (choker, at location, jewelry box), (jewelry, at location, box), (boutique, is a, store), (choker, at location, boutique), (box, used for, store), (choker, is a, jewelry), (choker, at location, jewlery box), ...

Figure 3: Example of the visual graph comprehension question and answer pairs.

Algorithm 1 `GraphVis`

Input: Training data from the textual QA dataset: $\{\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\}_{i\in[N]}$. LVLM parameterized by $\boldsymbol{\theta}$: $p_{\boldsymbol{\theta}}$. The relevant KG: $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. Relevent subgraph retrieval method f. Self-supervised graph question set $P = \{\mathbf{p}^{(i)}\}_{i\in[M]}$.
Let graphical feature training dataset $D_g = \{\}$ and the graph VQA dataset $D_v = \{\}$.
for $i = 1, \ldots N$ **do**
  Retrieve the relevant KG subgraph $\mathcal{G}_i = f(\mathbf{x}^{(i)}, \mathcal{V}_{\mathbf{x}^{(i)}}, \mathcal{G})$.
  Plot the KG subgraph to obtain the image for visualized KG $\mathbf{v}^{(i)}$.
  for $j = 1, \ldots M$ **do**
    Given $\mathbf{p}^{(j)}$ and $\mathcal{G}_i$, automatically get answer $\mathbf{a}^{(j)}$.
    Add $\left(\mathbf{v}^{(i)}, \mathbf{p}^{(j)}, \mathbf{a}^{(j)}\right)$ to D_g.
  end for
  Add $\left(\mathbf{v}^{(i)}, \mathbf{x}^{(i)}, \mathbf{y}^{(i)}\right)$ to D_v.
end for
Graph understanding fine-tuning: update $\widehat{\boldsymbol{\theta}} = \text{argmin}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \sum_{(\mathbf{v},\mathbf{x},\mathbf{y})\in D_g} \Big(-\log p_{\boldsymbol{\theta}}\big(\mathbf{y}|\mathbf{v},\mathbf{x}\big)\Big)$.
KG-enhanced QA fine-tuning: update $\widehat{\boldsymbol{\theta}} = \text{argmin}_{\widehat{\boldsymbol{\theta}}\in\boldsymbol{\Theta}} \sum_{(\mathbf{v},\mathbf{x},\mathbf{y})\in D_v} \Big(-\log p_{\widehat{\boldsymbol{\theta}}}\big(\mathbf{y}|\mathbf{v},\mathbf{x}\big)\Big)$.
Output: $\widehat{\boldsymbol{\theta}}$.

consider Commonsense QA (CSQA) (Talmor et al., 2019) and OpenBook QA (OBQA) (Mihaylov et al., 2018) as the commonsense reasoning tasks that can be improved via relevant subgraphs in ConceptNet. For the zero-shot VQA tasks, we consider ScienceQA (Lu et al., 2022), MMBench (Liu et al., 2023c) and POPE (Li et al., 2023b) that share similar images or tasks as our synthetic data from textual QA with visual KG subgraphs. Specifically, ScienceQA focuses on scientific question answering and contains scientific diagrams. MMBench is a recent multi-modal benchmark that comprehensively evaluates a model's capabilities in a wide range of tasks and evaluation criteria. POPE evaluates the extent of object hallucinations for LVLMs, formulating a binary classification task by prompting the model with questions such as "Is there an <object> in this image?". For VQA benchmarks, we use the evaluation scripts provided by LLaVA (Liu et al., 2023a) to obtain the results for both our base model and after using `GraphVis` to ensure a fair comparison. In Figure 4 and 5, we demonstrate the statistics of the synthetic visual knowledge graphs in CSQA.

Baselines. We consider the previous KG-enhanced methods that fine-tune language models on the training data with ConceptNet as one category of the baselines, including the popular *QA-GNN* (Yasunaga et al., 2021) and *GreaseLM* (Zhang et al., 2022). We further include the performance of current LLMs without KG or fine-tuning, including *FLAN-T5-xxlarge* (11B) (Chung et al., 2024), which is the base LLM used for many KG-enhanced methods, and *GPT-4*. Lastly, we include the reported values of methods on KG-enhanced LLMs including *KAPING* (Baek et al., 2023), *KSL* (Feng et al., 2023) and Graph Neural Promping (*GNP*) (Tian et al., 2024), which all share the same setting of using ConceptNet for enhancement on commonsense reasoning tasks. In particular, KSL and GNP are fine-tuning approaches and we report their best performances (e.g. for GNP, we consider the results from both fine-tuning GNN and projection network and Low-Rank Adaptation (LoRA) (Hu

et al., 2021) fine-tuning on LLM). Lastly, we note that these methods have not open-sourced their codes and models, and therefore we consider our re-implementation of KAPING based on the same VLM as a reference.

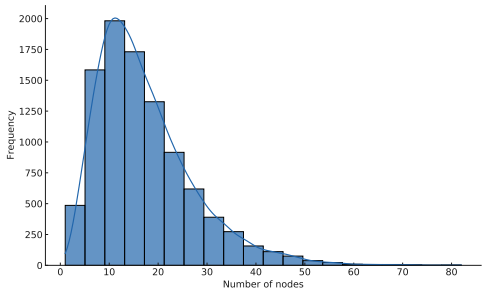


Figure 4: Distribution of node number in CSQA.

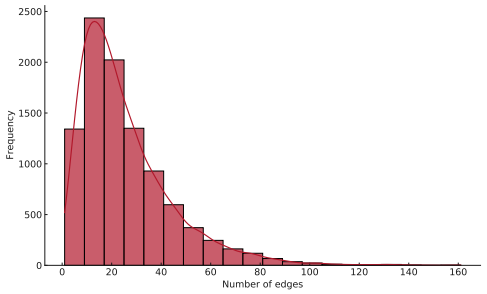


Figure 5: Distribution of edge number in CSQA.

5.2 Main Results

In Table 1, we present the main results of `GraphVis` on KG-enhanced question answering. `GraphVis` demonstrates a significant improvement in accuracy over the base model, with an increase of 12.3% on CSQA and 9.9% on OBQA. We include the full fine-tuning methods of KG-enhanced LMs as strong baselines, which include well-designed model architectures based on small-scale language models to better integrate KG information. Although the current methods proposed for KG-enhanced LLMs are not open-sourced at the time of this manuscript, we incorporate the reported values of the baselines, including KSL, KAPING, and GNP. We observe that, due to the strong performance of the base LLMs, prompting methods like KAPING can actually harm performance by causing notably longer contexts with information not in natural language form. Conversely, fine-tuning methods like KSL and GNP offer greater improvements, even though mostly under-performing or matching the performance of full-parameter fine-tuned LMs that have intrinsic architectural changes to adapt the KG information. Meanwhile, the scale of LLMs is unprecedented, causing difficulty in both modifying the architecture or fully fine-tuning all parameters. While `GraphVis` similarly employs LoRA fine-tuning to only update a small amount of parameters similar to KSL and GNP, we observe a much more significant improvement that suggests a better incorporation of the information. On CSQA, `GraphVis` with a 7B LLM surpasses the second-best result, KSL with GPT-3.5 (>100B), by a substantial margin of 3.2%. On OBQA, `GraphVis` remains the top-performing method, outperforming fine-tuning methods like GNP with an 11B LLM by 5.7%.

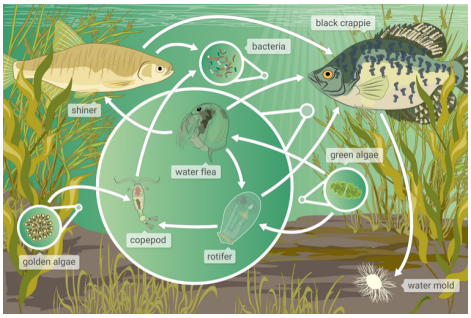
Table 1: Performance of `GraphVis` compared with the original VLM model across benchmarks and VQA tasks. As current baselines on LLMs are not open-sourced yet, we include the results directly reported from their papers (Zhang et al., 2022; Feng et al., 2023; Tian et al., 2024). We use FT to indicate if a method involves fine-tuning. The **bold** numbers indicate the best results among all methods and underscored numbers represent the second best.

Category	Method	Base Model	FT	CSQA	OBQA
LM	QA-GNN	AristoRoBERTa (355M)	✓	76.1	82.8
	GreaseLM	AristoRoBERTa (355M)	✓	78.5	84.8
LLM	Base LLM	GPT-3.5 (>100B)	✗	72.9	74.8
	KSL	GPT-3.5 (>100B)	✗	79.6	81.6
	Base LLM	LLaMA (7B)	✗	38.0	29.8
	KSL	LLaMA (7B)	✓	47.4	45.8
	Base LLM	FLAN-T5-xxlarge (11B)	✗	–	76.8
	KAPING	FLAN-T5-xxlarge (11B)	✗	–	60.0
	GNP	FLAN-T5-xxlarge (11B)	✓	–	79.8
LVLM	Base LVLM	LLaVA-v1.6-Mistral (7B)	✗	70.5	75.6
	KAPING	LLaVA-v1.6-Mistral (7B)	✗	67.7	71.2
	`GraphVis`	LLaVA-v1.6-Mistral (7B)	✓	**82.8** (+12.3)	**85.5** (+9.9)

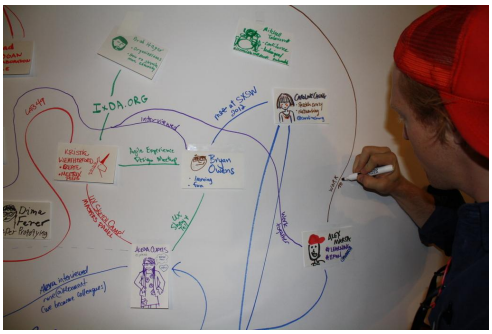
https://doi.org/10.52202/079017-2155

5.2.1 Leveraging KG and Textual Data to Enhance LVLM

Furthermore, we investigate the benefit of `GraphVis` in the reverse direction, by leveraging textual QA dataset and KG to improve the LVLM's zero-shot performance on VQA tasks. We begin with the observation that many prevalent VQA benchmarks, such as ScienceQA Lu et al. (2022), feature images structured as directed graphs. For instance, ScienceQA contains a category of image exists as the food web images with questions to identify the decomposers or the producers in the web, as the example shown in Figure 6a. Similarly, MMBench (Liu et al., 2023c) includes a notable portion of images that comprise charts and diagrams, illustrated in Figure 6b. While current LVLMs are pre-trained and fine-tuned on large corpus of vision-language instruction data, images of graph structures are much more scarce compared to the many natural images, in addition to the scarcity of reasoning tasks designed specifically for graphs. The presence of structured graphical images within VQA benchmarks highlights the potential of `GraphVis` to leverage textual data and KG images to improve the LVLM's capability in reasoning with such type of images.



(a) Example image from ScienceQA (Lu et al., 2022). Question: Which of the following organisms is the decomposer in this food web?



(b) Example image from MM-Bench (Liu et al., 2023c). Question: who is at the center of all of this?

Figure 6: Example images from VQA tasks that share resemblance to the visualized KG subgraphs.

In Table 2, we present the performance of our base LVLM (`llava-v1.6-mistral-7b`) and its comparison after applying `GraphVis`. Notably, `GraphVis` employs synthetic images with textual QAs and does not require human-curated VQA training data, yet it robustly generalizes the visual graph comprehension capabilities to diagrams in current VQA benchmarks. We can observe a remarkable improvement of 4.32% on ScienceQA and 2.66% on MMBench. Furthermore, by leveraging the node description and number detection tasks in our graph comprehension fine-tuning, we explore the impact of `GraphVis` on object hallucinations. Through evaluations using POPE across its three scenarios (random, popular, and adversarial) we find that `GraphVis` effectively reduces object hallucinations in the LVLM, enhancing both the accuracy and the F1 score in determining whether an object is present in an image. This results in an average improvement of 1.09%. Additionally, we note that differences exist between our visualized knowledge graphs and the graph images in these VQA benchmarks in terms of visual clarity, graph layout, information density, and domain knowledge. Despite these disparities, the model consistently shows improvements across various distinct benchmarks and demonstrates robust generalization capabilities, transitioning effectively from abstract graph structures to real-world images.

Table 2: VQA performance of `GraphVis` based on `llava-v1.6-mistral-7b`.

Model	ScienceQA	MMBench	POPE-ran		POPE-pop		POPE-adv	
	Img-Acc	Overall	Acc	F1	Acc	F1	Acc	F1
Base LVLM	68.86	63.75	88.56	87.65	87.73	86.53	86.47	85.37
`GraphVis`	**73.18**$_{(+4.32)}$	**66.41**$_{(+2.66)}$	**89.73**	**89.12**	**88.73**	**87.89**	**87.07**	**86.32**

6 Ablation Study

In this section, we conduct further ablation studies to explore the different variants of `GraphVis` to illustrate the significance of the components within our method design.

Curriculum fine-tuning. `GraphVis` emphasizes a curriculum fine-tuning scheme, initially training the model on fundamental visual graph concepts, such as node number and node degrees. Only after mastering these basic comprehension tasks does the model advance to train on the more complex

reasoning tasks which require leveraging visual graphs to answer relevant questions. Here, we evaluate the impact of task sequencing in fine-tuning by comparing the standard `GraphVis` with a variant that jointly fine-tunes across the mixed data, encompassing both image comprehension and reasoning tasks. Additionally, we categorize image comprehension tasks into two distinct groups for more fine-grained curriculum fine-tuning:

- **OCR** tasks, including node description and triple listing.
- **Graph** tasks, including node degree detection, highest node degree detection, and node/edge number detection.

Figure 7 presents the CSQA performance results for each fine-tuning strategy of `GraphVis`. Although `GraphVis` generally enhances performance across the different schemes, improvements are notably less significant when tasks are jointly trained. The curriculum-based approach yields an additional gain of 4.51% over joint fine-tuning. However, the benefits of more detailed fine-tuning appear minimal. Initiating fine-tuning with OCR tasks, followed by graph tasks and subsequent QA reasoning, leads to a marginal increase of 0.24%. Conversely, reversing the order of these detailed tasks results in a performance decline of 1.56%. These findings indicate that optimal fine-tuning involves separating initial image comprehension stages from subsequent reasoning tasks.

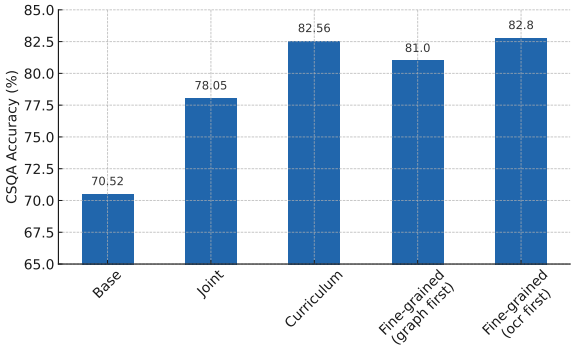


Figure 7: Comparison between different fine-tuning schemes with `GraphVis` on CSQA.

Combination with prompting. As `GraphVis` is intrinsically compatible with prompting methods like KAPING, we explore the integration of the two methods, as demonstrated in Table 3. Firstly, due to the strong performance of current LLMs, incorporating KAPING into the base model results in a performance decline of 2.87%, a trend consistent with results reported in the previous study (Tian et al., 2024). Such degradation can be attributed to both the prolonged context and the fact that the original model was not adept at understanding graph structures. Meanwhile, for `GraphVis` with joint fine-tuning that understands the graph structure as well as the triplet format through the task of triple listing, an improvement of 0.66% is observed. However, for `GraphVis` with curriculum fine-tuning that has more effectively learned the visual graph, the addition of KAPING prompts appears to be redundant and causes a minor degradation of 0.82%.

Table 3: Performance of `GraphVis` based on `llava-v1.6-mistral-7b` with or without the prompting from KAPING.

	Original	w/ KAPING
Base LVLM	70.52	67.65$_{(-2.87)}$
`GraphVis` (Joint)	78.05	78.71$_{(+0.66)}$
`GraphVis`	82.56	81.74$_{(-0.82)}$

Performance on graph comprehension task. In Table 4, we further evaluate the LVLM on the graph comprehension tasks we defined, both before and after training on the synthetic tasks. To ensure a fair comparison, we utilized synthetic images from the test data of CSQA to construct a test set. The accuracy for each individual task is reported. We implement exact matching in determining answer accuracy, which, while strict, provides insight into performance gains and error sources. We observed that graph comprehension tasks are essentially difficult for the LVLM, as such images and tasks are scarce in its pre-training and fine-tuning data. On tasks such as triple listing, it almost cannot fulfill the task. For an output example: "*Based on the image provided, the graph appears to represent a network or a system with nodes (blue circles) and edges (black lines) connecting them. To list all the triples in the graph, I'll describe each triple as a sequence of three nodes in the*

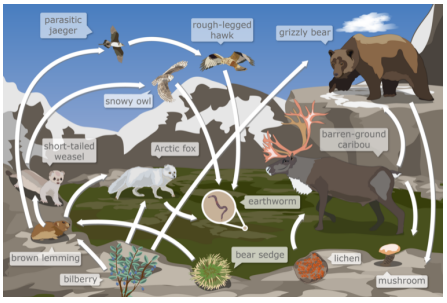
https://doi.org/10.52202/079017-2155

graph, which are connected by edges. Here are the triples in the graph: 1. (node1, node2, node3) 2. (node2, node3, node4)..." Since these preliminary tasks were considered a warm start for the model to learn grounding its reasoning on graph images, we only fine-tuned on these tasks for one epoch. Nevertheless, we observed a notable gain across all tasks after just one epoch of fine-tuning.

Table 4: Performance of LLaVA-v1.6 before and after fine-tuning on each graph comprehension task. N. denotes node and E. denotes Edge. For node description and triple listing, we consider the average accuracy of each test example. We use exact matching to determine the accuracy, which may be a stricter evaluation.

Model	N. description	N. degree	Highest N. degree	N. number	E. number	Triple listing
Original	1.4	15.3	3.3	16.7	9.7	0.6
After fine-tuning	12.8$_{(+11.4)}$	27.0$_{(+11.7)}$	11.6$_{(+8.3)}$	27.5$_{(+10.8)}$	16.2$_{(+9.7)}$	8.2$_{(+7.6)}$

Qualitative example. In Figure 8, we provide a specific example of the model generations for the VQA task ScienceQA. The displayed question fundamentally requires the model to traverse through the food web from a starting point, following the directed arrows, and match the target node names with the provided options. The original model failed to complete this task successfully. However, with `GraphVis`, the model's ability to handle such image data improved significantly, resulting in a correct answer.



Query: Context: Below is a food web from a tundra ecosystem in Nunavut, a territory in Northern Canada. A food web models how the matter eaten by organisms moves through an ecosystem. The arrows in a food web represent how matter moves between organisms in an ecosystem.
Which of these organisms contains matter that was once part of the lichen?
A. mushroom B. short-tailed weasel C. brown lemming
D. rough-legged hawk E. bilberry
Answer with the option's letter from the given choices directly.

Base (LLaVA-v1.6 7B): C **GraphVis (LLaVA-v1.6 7B)**: A

Figure 8: Example of model output on ScienceQA (VQA task). Note that after fine-tuning the base LVLM with GraphVis on CSQA with synthetic KG images, the model can successfully traverse the graph to locate the correct answer.

7 Conclusion

In conclusion, we proposed `GraphVis`, a new approach to integrate structured knowledge from KGs with LLMs through the visual modality. By preserving the intricate graph structure and employing a curriculum fine-tuning scheme, our method not only enhanced LLMs' ability to comprehend and reason over KG data to enhance its response to textual QAs but also significantly improves performance across several VQA benchmarks. `GraphVis` leveraged the strengths of both textual, visual and KG data, reducing factual inaccuracies and hallucinations typical in LLM outputs. The promising results achieved on multiple benchmarks underscore the potential of `GraphVis` to set a new approach of utilizing data from the KG modality and enhancing a model's performance in the cross-modal fashion.

Limitations and future work. Firstly, we acknowledge the limitation induced by compute resources that our experiments are done on 7B models with LoRA fine-tuning. If compute resource permits, it is interesting to scale up the experiments with larger models and full fine-tuning. Another limitation is the size of the retrieved subgraph, for which we considerd a 2-hop subgraph to ensure that the visualization is not too complicated for the vision model to recognize. Extending from our current method, interesting future work includes exploring how different visualizations may influence the effectiveness of `GraphVis`. Additionally, instead of following the previous retrieval methods, it would be valuable to investigate better subgraph retrieval techniques and integrate them into the learning process. Lastly, while we used ConceptNet as an example KG to enhance commonsense reasoning, there are numerous other KGs available. It is crucial to explore the generalizability of `GraphVis` to adapt to new KGs. Furthermore, it is possible to obtain multiple relevant subgraphs for a given question from different KG sources. An open problem remains on how to leverage multiple KG subgraphs for enhanced reasoning in LLMs.

Acknowledgments

We sincerely thank the anonymous reviewers for their helpful comments. The work is partially supported by DARPA HR0011-24-9-0370, NSF 2200274, 2106859, 2312501, NIH U54HG012517, U24DK097771, and Optum AI. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

References

AGRAWAL, G., KUMARAGE, T., ALGHAMI, Z. and LIU, H. (2023). Can knowledge graphs reduce hallucinations in llms?: A survey. *arXiv preprint arXiv:2311.07914* .

ALAYRAC, J.-B., DONAHUE, J., LUC, P., MIECH, A., BARR, I., HASSON, Y., LENC, K., MENSCH, A., MILLICAN, K., REYNOLDS, M. ET AL. (2022). Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems* **35** 23716–23736.

BAEK, J., AJI, A. F. and SAFFARI, A. (2023). Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. *arXiv preprint arXiv:2306.04136* .

BAI, J., BAI, S., YANG, S., WANG, S., TAN, S., WANG, P., LIN, J., ZHOU, C. and ZHOU, J. (2023). Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966* .

BROWN, T., MANN, B., RYDER, N., SUBBIAH, M., KAPLAN, J. D., DHARIWAL, P., NEELAKANTAN, A., SHYAM, P., SASTRY, G., ASKELL, A. ET AL. (2020). Language models are few-shot learners. *Advances in neural information processing systems* **33** 1877–1901.

CHAI, Z., ZHANG, T., WU, L., HAN, K., HU, X., HUANG, X. and YANG, Y. (2023). Graphllm: Boosting graph reasoning ability of large language model. *arXiv preprint arXiv:2310.05845* .

CHANG, C., PENG, W.-C. and CHEN, T.-F. (2023). Llm4ts: Two-stage fine-tuning for time-series forecasting with pre-trained llms. *arXiv preprint arXiv:2308.08469* .

CHEN, J., ZHU, D., SHEN, X., LI, X., LIU, Z., ZHANG, P., KRISHNAMOORTHI, R., CHANDRA, V., XIONG, Y. and ELHOSEINY, M. (2023). Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478* .

CHUNG, H. W., HOU, L., LONGPRE, S., ZOPH, B., TAY, Y., FEDUS, W., LI, Y., WANG, X., DEHGHANI, M., BRAHMA, S. ET AL. (2024). Scaling instruction-finetuned language models. *Journal of Machine Learning Research* **25** 1–53.

DAI, W., LI, J., LI, D., TIONG, A. M. H., ZHAO, J., WANG, W., LI, B., FUNG, P. and HOI, S. (2023). Instructblip: Towards general-purpose vision-language models with instruction tuning.

DING, N., QIN, Y., YANG, G., WEI, F., YANG, Z., SU, Y., HU, S., CHEN, Y., CHAN, C.-M., CHEN, W. ET AL. (2023). Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence* **5** 220–235.

FATEMI, B., HALCROW, J. and PEROZZI, B. (2023). Talk like a graph: Encoding graphs for large language models. *arXiv preprint arXiv:2310.04560* .

FENG, C., ZHANG, X. and FEI, Z. (2023). Knowledge solver: Teaching llms to search for domain knowledge from knowledge graphs. *arXiv preprint arXiv:2309.03118* .

FENG, Y., CHEN, X., LIN, B. Y., WANG, P., YAN, J. and REN, X. (2020). Scalable multi-hop relational reasoning for knowledge-aware question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

GANSNER, E. R. and NORTH, S. C. (2000). An open graph visualization system and its applications to software engineering. *Software: practice and experience* **30** 1203–1233.

GAO, H., MAO, J., ZHOU, J., HUANG, Z., WANG, L. and XU, W. (2015). Are you talking to a machine? dataset and methods for multilingual image question. *Advances in neural information processing systems* **28**.

 https://doi.org/10.52202/079017-2155

GAO, P., HAN, J., ZHANG, R., LIN, Z., GENG, S., ZHOU, A., ZHANG, W., LU, P., HE, C., YUE, X., LI, H. and QIAO, Y. (2023). Llama-adapter v2: Parameter-efficient visual instruction model.

GUO, J., DU, L. and LIU, H. (2023). Gpt4graph: Can large language models understand graph structured data? an empirical evaluation and benchmarking. *arXiv preprint arXiv:2305.15066* .

HU, E. J., WALLIS, P., ALLEN-ZHU, Z., LI, Y., WANG, S., WANG, L., CHEN, W. ET AL. (2021). Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

HU, Z., XU, Y., YU, W., WANG, S., YANG, Z., ZHU, C., CHANG, K.-W. and SUN, Y. (2022). Empowering language models with knowledge graph reasoning for question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.

HUANG, Z., LI, Z., JIANG, H., CAO, T., LU, H., YIN, B., SUBBIAN, K., SUN, Y. and WANG, W. (2022). Multilingual knowledge graph completion with self-supervised adaptive graph alignment. In *Annual Meeting of the Association for Computational Linguistics (ACL)*.

HUANG, Z., WANG, D., HUANG, B., ZHANG, C., SHANG, J., LIANG, Y., WANG, Z., LI, X., FALOUTSOS, C., SUN, Y. and WANG, W. (2023). Concept2Box: Joint geometric embeddings for learning two-view knowledge graphs. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics, Toronto, Canada.

JIA, C., YANG, Y., XIA, Y., CHEN, Y.-T., PAREKH, Z., PHAM, H., LE, Q., SUNG, Y.-H., LI, Z. and DUERIG, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*. PMLR.

JIANG, A. Q., SABLAYROLLES, A., MENSCH, A., BAMFORD, C., CHAPLOT, D. S., CASAS, D. D. L., BRESSAND, F., LENGYEL, G., LAMPLE, G., SAULNIER, L. ET AL. (2023). Mistral 7b. *arXiv preprint arXiv:2310.06825* .

JIN, B., LIU, G., HAN, C., JIANG, M., JI, H. and HAN, J. (2023). Large language models on graphs: A comprehensive survey. *arXiv preprint arXiv:2312.02783* .

KANG, M., BAEK, J. and HWANG, S. J. (2022). Kala: Knowledge-augmented language model adaptation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

KENTON, J. D. M.-W. C. and TOUTANOVA, L. K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, vol. 1.

LI, J., CHENG, X., ZHAO, W. X., NIE, J.-Y. and WEN, J.-R. (2023a). Halueval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

LI, S., LI, X., SHANG, L., SUN, C.-J., LIU, B., JI, Z., JIANG, X. and LIU, Q. (2022). Pre-training language models with deterministic factual knowledge. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.

LI, Y., DU, Y., ZHOU, K., WANG, J., ZHAO, W. X. and WEN, J.-R. (2023b). Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355* .

LI, Y., WANG, C. and JIA, J. (2023c). Llama-vid: An image is worth 2 tokens in large language models. *arXiv preprint arXiv:2311.17043* .

LIN, B. Y., CHEN, X., CHEN, J. and REN, X. (2019). Kagnet: Knowledge-aware graph networks for commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

LIU, H., LI, C., LI, Y. and LEE, Y. J. (2023a). Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744* .

LIU, H., LI, C., LI, Y., LI, B., ZHANG, Y., SHEN, S. and LEE, Y. J. (2024). Llava-next: Improved reasoning, ocr, and world knowledge.

LIU, H., LI, C., WU, Q. and LEE, Y. J. (2023b). Visual instruction tuning. In *NeurIPS*.

LIU, Y., DUAN, H., ZHANG, Y., LI, B., ZHANG, S., ZHAO, W., YUAN, Y., WANG, J., HE, C., LIU, Z. ET AL. (2023c). Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281* .

LU, P., BANSAL, H., XIA, T., LIU, J., LI, C., HAJISHIRZI, H., CHENG, H., CHANG, K.-W., GALLEY, M. and GAO, J. (2024). Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts.

LU, P., MISHRA, S., XIA, T., QIU, L., CHANG, K.-W., ZHU, S.-C., TAFJORD, O., CLARK, P. and KALYAN, A. (2022). Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems* **35** 2507–2521.

LUO, L., LI, Y.-F., HAF, R. and PAN, S. (2023). Reasoning on graphs: Faithful and interpretable large language model reasoning. In *The Twelfth International Conference on Learning Representations*.

MAAZ, M., RASHEED, H., KHAN, S. and KHAN, F. S. (2023). Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424* .

MIHAYLOV, T., CLARK, P., KHOT, T. and SABHARWAL, A. (2018). Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*.

OPENAI (2023). Gpt-4 technical report.

PAN, S., LUO, L., WANG, Y., CHEN, C., WANG, J. and WU, X. (2023). Unifying large language models and knowledge graphs: A roadmap. *arXiv preprint arXiv:2306.08302* .

PENG, Z., WANG, W., DONG, L., HAO, Y., HUANG, S., MA, S. and WEI, F. (2023). Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824* .

RADFORD, A., KIM, J. W., HALLACY, C., RAMESH, A., GOH, G., AGARWAL, S., SASTRY, G., ASKELL, A., MISHKIN, P., CLARK, J. ET AL. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR.

RADFORD, A., WU, J., CHILD, R., LUAN, D., AMODEI, D., SUTSKEVER, I. ET AL. (2019). Language models are unsupervised multitask learners. *OpenAI blog* **1** 9.

ROSSET, C., XIONG, C., PHAN, M., SONG, X., BENNETT, P. and TIWARY, S. (2020). Knowledge-aware language model pretraining. *arXiv preprint arXiv:2007.00655* .

RUBENSTEIN, P. K., ASAWAROENGCHAI, C., NGUYEN, D. D., BAPNA, A., BORSOS, Z., QUITRY, F. D. C., CHEN, P., BADAWY, D. E., HAN, W., KHARITONOV, E. ET AL. (2023). Audiopalm: A large language model that can speak and listen. *arXiv preprint arXiv:2306.12925* .

SHEN, T., MAO, Y., HE, P., LONG, G., TRISCHLER, A. and CHEN, W. (2020). Exploiting structured knowledge in text via graph-guided representation learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

SPEER, R., CHIN, J. and HAVASI, C. (2017). Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence*, vol. 31.

SUN, J., XU, C., TANG, L., WANG, S., LIN, C., GONG, Y., SHUM, H.-Y. and GUO, J. (2023). Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph. *arXiv preprint arXiv:2307.07697* .

SUN, Y., SHI, Q., QI, L. and ZHANG, Y. (2021). Jointlk: Joint reasoning with language models and knowledge graphs for commonsense question answering. In *North American Chapter of the Association for Computational Linguistics*.

TALMOR, A., HERZIG, J., LOURIE, N. and BERANT, J. (2019). Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.

 https://doi.org/10.52202/079017-2155

TIAN, Y., SONG, H., WANG, Z., WANG, H., HU, Z., WANG, F., CHAWLA, N. V. and XU, P. (2024). Graph neural prompting with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38.

TOUVRON, H., LAVRIL, T., IZACARD, G., MARTINET, X., LACHAUX, M.-A., LACROIX, T., ROZIÈRE, B., GOYAL, N., HAMBRO, E., AZHAR, F. ET AL. (2023a). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* .

TOUVRON, H., MARTIN, L., STONE, K., ALBERT, P., ALMAHAIRI, A., BABAEI, Y., BASHLYKOV, N., BATRA, S., BHARGAVA, P., BHOSALE, S. ET AL. (2023b). Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* .

WANG, X., GAO, T., ZHU, Z., ZHANG, Z., LIU, Z., LI, J. and TANG, J. (2021). Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics* **9** 176–194.

WU, S., FEI, H., QU, L., JI, W. and CHUA, T.-S. (2023). Next-gpt: Any-to-any multimodal llm. *arXiv preprint arXiv:2309.05519* .

XU, P., SHAO, W., ZHANG, K., GAO, P., LIU, S., LEI, M., MENG, F., HUANG, S., QIAO, Y. and LUO, P. (2023). Lvlm-ehub: A comprehensive evaluation benchmark for large vision-language models. *arXiv preprint arXiv:2306.09265* .

YANG, L., CHEN, H., LI, Z., DING, X. and WU, X. (2023). Chatgpt is not enough: Enhancing large language models with knowledge graphs for fact-aware language modeling. *arXiv preprint arXiv:2306.11489* .

YASUNAGA, M., REN, H., BOSSELUT, A., LIANG, P. and LESKOVEC, J. (2021). Qa-gnn: Reasoning with language models and knowledge graphs for question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

YE, Q., XU, H., XU, G., YE, J., YAN, M., ZHOU, Y., WANG, J., HU, A., SHI, P., SHI, Y. ET AL. (2023). mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178* .

YU, W., YANG, Z., LI, L., WANG, J., LIN, K., LIU, Z., WANG, X. and WANG, L. (2023a). Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490* .

YU, X., CHEN, Z., LING, Y., DONG, S., LIU, Z. and LU, Y. (2023b). Temporal data meets llm–explainable financial time series forecasting. *arXiv preprint arXiv:2306.11025* .

YUJIE, W., HU, Z., JIYE, L. and RU, L. (2023). Dynamic heterogeneous-graph reasoning with language models and knowledge representation learning for commonsense question answering. In *Association for Computational Linguistics (ACL)*.

ZHANG, D., LI, S., ZHANG, X., ZHAN, J., WANG, P., ZHOU, Y. and QIU, X. (2023). Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

ZHANG, D., YUAN, Z., LIU, Y., ZHUANG, F., CHEN, H. and XIONG, H. (2020). E-bert: A phrase and product knowledge enhanced language model for e-commerce. *arXiv preprint arXiv:2009.02835* .

ZHANG, X., BOSSELUT, A., YASUNAGA, M., REN, H., LIANG, P., MANNING, C. and LESKOVEC, J. (2022). Greaselm: Graph reasoning enhanced language models for question answering. In *International Conference on Representation Learning (ICLR)*.

ZHANG, Z., HAN, X., LIU, Z., JIANG, X., SUN, M. and LIU, Q. (2019). Ernie: Enhanced language representation with informative entities. *arXiv preprint arXiv:1905.07129* .

ZHU, D., CHEN, J., SHEN, X., LI, X. and ELHOSEINY, M. (2023). Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592* .

A Experiment Details

Visual graph comprehension data. We created a pool of five prompts for each of the task in visual graph comprehension, where the answers can be automatically extracted. For node description, we have

- "List all nodes of the graph shown in the image."
- "Provide the names of all nodes displayed in the graph image."
- "Can you name all the nodes shown in the graph image?"
- "Identify all the vertices in the diagram of the graph provided."
- "Detail all the vertices from the graph depicted in the image."

For highest node degree detection, we have

- "Name one of the node with the highest degree in the graph. And what is its degree?"
- "Identify one of the node that has the most connections in the graph and specify its degree."
- "Can you tell me which node (name one) has the highest degree in this graph and what that degree is?"
- "Provide the name and degree of the node with the most connections in the graph."
- "Which node in the graph has the greatest number of connections, and what is that total?"

For node degree detection, we have

- "What is the degree of the node with the name "node"?",
- "What is the degree of the node labeled "node"?",
- "Can you tell me the degree of the node named "node"?"
- "What is the total number of connections that the node "node" has?"
- "How many connections does the node "node" have?"

For node number detection, we have

- "How many nodes are there in the graph?"
- "What is the total number of nodes in the graph?"
- "Can you tell me how many nodes are in the graph?"
- "What is the total number of vertices in the graph?"
- "How many vertices are there in the graph?"

For edge number detection, we have

- "How many edges are there in the graph?"
- "What is the total number of edges in the graph?"
- "Can you tell me how many edges are in the graph?"
- "What is the total number of connections in the graph?"
- "How many connections are there in the graph?"

For triple listing, we have

- "List all the triples in the graph."
- "Provide all the triples in the graph."
- "Can you list all the triples in the graph?"
- "Detail all the triples in the graph."
- "Enumerate all the triples in the graph."

Fine-tuning. We train 1 epoch for both part of the fine-tuning process. We present the fine-tuning hyperparameters of `GraphVis` in Table 5.

https://doi.org/10.52202/079017-2155

Table 5: Fine-tuning hyperparameters.

lora_r	128
lora_alpha	256
lora_target	all
Learning rate	1e-7
Optimizer	AdamW
Global batch size	4
gradient_accumulation_steps	1
weight_decay	0
warmup_ratio	0.03
lr_scheduler_type	cosine
image_aspect_ratio	pad
group_by_modality_length	True
model_max_length	2048
mm_projector_lr	2e-5
mm_projector_type	mlp2x_gelu

Evaluation. We use the same evaluation scripts provided by LLaVA (Liu et al., 2023a) for all evaluations performed in this paper. We note that the new evaluation scripts (prompts) used to report the newest results of LLaVA-v1.6 are not released yet, which may cause minor differences in evaluation results of the original model compared to their reported values. Nevertheless, we use the same evaluation scripts throughout the paper to ensure fairness in comparison.

Compute resources. Experiments of this paper were all conducted on NVIDIA RTX A6000 GPU clusters. The fine-tuning of LLaVA v1.5 (7B) on the visualized subgraphs takes approximately 3 hours on 4 GPUs. The time span for evaluations on the different benchmarks range from 0.5 to 8 hours using 1 GPU, depending on the varying size of the dataset.

Additional Experiment Results In Table 6, we include the additional results on ScienceQA as one of the VQA tasks from either doing a curriculum fine-tuning or simply joint fine-tuning on the curated synthetic data. As indicated by the results, curriculum learning transfers to these VQA tasks as well. In Figure 9, we investigate the influence of image quality for the synthetic visual graphs used for training. It is generally observed in VQA tasks that images with lower resolution can lead to degraded performance, as these images are considered as "corrupted" and often leads to object hallucinations. For the QA tasks that we considered in our evaluation, we conducted additional experiments using graph images with smaller sizes and consequently lower resolutions (50x50).

Table 6: Performance of LLaVA-v1.6 on ScienceQA compared with GraphVis and GraphVis (joint fine-tuning).

	ScienceQA (%)
Base LVLM	68.86
GraphVis (Joint)	71.94
GraphVis	**73.18**

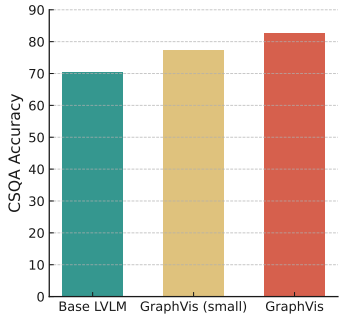


Figure 9: Effect of lower resolution graph images.

B Broader Impact

By leveraging factual information from KGs, `GraphVis` aims to mitigate inaccuracies in the reasoning process and effectively reduced hallucinations in model outputs. This approach aims for a more accurate and reliable model, contributing positively in social impact by providing a more trustworthy and accountable AI model. The improved accuracy and reliability of `GraphVis` can enhance user trust in AI applications, especially in critical areas such as healthcare, education, and legal advice.

Meanwhile, there are potential negative societal impacts of enhanced LVLMs capabilities. As `GraphVis` increases the effectiveness of these models, there is a risk of misuse in ways that could harm privacy and fairness. For instance, more advanced LVLMs could be exploited to generate misleading or deceptive content or amplify biases present in the underlying data, leading to unfair outcomes. To address these concerns, it is crucial to ensure transparency in how the models are trained and used, incorporating bias detection and mitigation strategies.

https://doi.org/10.52202/079017-2155

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and precede the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS paper checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims made in abstract and introduction of effectively leveraging KG via the vision modality and reversely improving LVLM's VQA capability are well-supported with our experiment results in Section 5.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations of our work are discussed in the conclusion section of this paper (Section 7).

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: This paper focuses on methodology and empirical analysis.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We thoroughly provide the algorithm and method pipeline in Section 4 and hyperparameters used in our experiments in Appendix A to ensure reproducibility. Furthermore, code and model weights will be released and maintained.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

https://doi.org/10.52202/079017-2155

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Codes and scripts are provided in the supplemental mateal.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Fine-tuning and evaluation details of our experiments are explained in Appendix A. The supplemental material also contains the codes for our work.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We used greedy-decoding during evaluation, which makes the process deterministic and affected by randomness in evaluation. Meanwhile, the fine-tuning process of a large vision language model is computationally expensive, and running the fine-tuning process for multiple times is prohibitively expensive. We admit the limitation caused by computational cost, and ensure that our experiment results are robust and reproducible.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provided the information of the required compute resources in Section A.

   Guidelines:

https://doi.org/10.52202/079017-2155

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Yes, the paper conform with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: In Appendix B, we discussed the potential impacts of this work.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

Justification: Our method does not focus on releasing a specific data or model such as pretrained language models.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All models and datasets used in this work are properly cited.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: This work does not aim to introduce new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

https://doi.org/10.52202/079017-2155

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This work did not conduct any crowdsourcing experiments.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The question is not applicable as this work does not involve study participants.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.