RefLVQA: Referential Long-Form Visual Question Answering with Multimodal Documents

Anonymous ACL submission

Abstract

Long-form question answering (LFQA) aims to generate grounded paragraph-length answers by leveraging external documents. However, existing LFQA research has largely overlooked *multimodality*. We introduce RefLVQA as the first LFQA dataset featuring visual questions and multimodal documents. The dataset comprises 157K visual QA pairs, each annotated with sentence-level reference documents in the form of citations. To evaluate the model's ability to support its responses using external documents, we propose a citation-based evaluation approach, where models are required to append appropriate citations to back up their answers. Our key findings are threefold: (1) Naïve multimodal RAG methods face challenges due to an excessive reliance on textual documents and insufficient grounding capabilities in image-based documents. (2) We propose Two-step MultiRAG, which outperforms unimodal RAG approaches, demonstrating the benefits of leveraging multimodal documents over unimodal ones. (3) Our qualitative analysis reveals that models frequently generate responses ungrounded in the referenced image documents.

1 Introduction

In real-world question answering, people often seek not just simple factual answers but comprehensive, well-structured responses. For example, in response to a question like "I have a headache and feel dizzy," a factoid answering system simply provides short facts such as "tension headache", whereas a long-form answering system finds multiple explanations on symptom and diagnosis to help users fully understand their situation. Long-form question answering (LFQA) generates comprehensive explanations by synthesizing information from multiple documents, addressing open-ended questions that may have various valid answers (Fan et al., 2019; Bolotova-Baranova et al., 2023; Han et al., 2024).

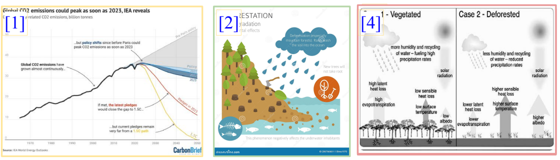
Question

What could be the potential consequences of this image?

Answer

Deforestation accelerates climate change by releasing stored CO_2, contributes to approximately 23% of global emissions[1] It causes soil degradation and erosion by removing stabilizing tree roots[2]. Deforestation also leads to biodiversity loss by destroying habitats that support over 80% of terrestrial species[3]. Furthermore, it disrupts water cycles by reducing transpiration, resulting in decreased rainfall and drier climates[4].

Multimodal Documents



[3] [...] forests are thought to provide habitat for 80% of amphibian species, 75% of bird species and 68% of mammal species [...]

Figure 1: A data point example of our RefLVQA benchmark. Each question is a pair of text and image. The answer is long-form; it contains multiple sentences, each of which may be associated with a multimodal supporting document.

Image documents, being inherently *compact yet rich*, can convey multiple layers of detailed and unique information that is often difficult to express in text. For instance, while text documents may describe high-level trends—such as "Global CO_2 emissions have grown almost continuously since the 1960s"—they often omit finer details like specific annual values or the magnitude of fluctuations, which are more effectively captured in images. Additionally, user-provided images play a critical role in understanding context and intent. Despite this, previous LFQA research has predominantly focused solely on the text modality—both for questions and reference documents.

In LFQA, providing a faithful answer requires not only generating the answer itself but also identifying supporting documents for each sentence (Han et al., 2024). Sentence-level annotation offers an advantage over coarse-grained document annotation—where only the overall answer and a set of supporting documents are provided—because LFQA answers typically consist of multiple sentences, making it difficult to determine which documents support which specific sentences.

We propose **RefLVQA** (*Referential Long-form Visual Question Answering*) as the first large-scale dataset for evaluating long-form answer generation ability of models with visual questions and multimodal reference documents. It is designed to evaluate how well the model generates comprehensive and well-grounded long-form answers, where each sentence is supported by external multimodal documents.

As illustrated in Figure 2, the task of our RefLVQA benchmark is performed in two stages: (1) Query generation and search: for a question with an image, the model crafts search queries by itself and retrieves Top-K documents for each query. (2) Referential answer generation: given a pool of retrieved documents, the model generates a final long-form answer, consisting of multiple sentences, each of which is associated with citation numbers (e.g., [1]) referencing documents. This explicit citation can directly identify utilized documents in sentence-level, facilitating a more precise assessment of the answer groundedness.

Due to the open-ended nature of LFQA, making binary judgments for the whole answer is inadequate for correct evaluation (Min et al., 2023). Hence, we evaluate model responses with three metrics: (1) Groundedness: how well each sentence in the model's answer is supported by the cited documents, (2) Completeness: how much the answer provides all necessary information to the question, and (3) Relevance: how well answer sentences are semantically aligned with the question. Our human evaluation results indicate that model-based evaluation correlates highly with human judgments, making it a scalable and reliable evaluation method.

We summarize our contributions: (1) We introduce RefLVQA as the first long-form question answering dataset with visual questions and multimodal reference documents. It contains 157K visual QA pairs, each with sentence-level citations to external documents. (2) We propose a citation-based evaluation method to assess the groundedness of model answers within a multimodal RAG framework. (3) We apply fine-grained evaluation metrics for long-form answers and confirmed that the scalable and efficient model-based evaluation correlates highly with human judgments.

2 Related Works

Long-Form Question Answering. LFQA aims to generate informative and coherent paragraph-level responses. ELI5 (Fan et al., 2019) uses questions from Reddit's "Explain Like I'm Five" forum, where users seek simple explanations for complex topics. HowSumm (Boni et al., 2021) leverages WikiHow articles to create query-focused summaries of procedural knowledge. They utilize reference-based metrics such as ROUGE (Lin, 2004) and BLEU (Papineni et al., 2002) for evaluation. To better address the open-ended nature of LFQA, recent work has proposed new evaluation strategies. LongFact (Wei et al., 2024b) introduces a fact-level framework that decomposes answers into atomic claims and verifies them via web search. RAG-QA Arena (Han et al., 2024) adopts a pairwise preference evaluation using model answers and gold answers. However, most LFQA benchmarks remain limited to text-only inputs, overlooking the importance of multimodality. On the other hand, our work focuses on general-purpose long-form visual QA, which requires grounding answers over both text and image documents.

Visual Question Answering. VQA has traditionally focused on generating concise, factoid-style answers. While recent benchmarks have broadened the scope to include diverse domains (Liu et al., 2024; Yue et al., 2024; Chen et al., 2024a), complex reasoning (Lu et al., 2023; Kembhavi et al., 2016), and external knowledge integration (Schwenk et al., 2022; Lu et al., 2022), the majority of VQA tasks still center around short-form answers. More recent efforts like VizWiz-LF (Huh et al., 2024) moves toward long-form visual QA by using open-ended questions from blind or low-vision users. However, it mainly evaluates answers based on what the model already knows, without using external information.

3 Dataset Creation

In real-world QA, users often require comprehensive and well-supported responses grounded in external knowledge. To fulfill these requirements, (1) detailed long-form answers are necessary to provide thorough explanations, (2) multimodality plays a pivotal role in offering richer external information and more precise question understanding, and (3) especially for long-form responses, sentence-level referencing is crucial to explicitly

	# of Instance			Document		Answer Length	Document Modality	Sentence-level Reference	Tasks
Dataset	Q	I	A	D_T/Q	D_I/Q				
ELI5 (2019)	272,000	0	272,000	1.0	0	130.6	Text	✗	Long-form QA
AquaMuse (2020)	5,519	0	5,519	6.0	0	105.9	Text	✗	Summarization
HowSumm (2021)	95,469	0	95,469	10.1	0	150.2	Text	✗	Summarization
WikihowQA (2023)	11,746	0	11,746	6.3	0	149.3	Text	✗	Long-form QA
LFRQA (2024)	26,907	0	26,907	3.0	0	76.3	Text	✓	Long-form QA & Text retrieval
LONGFACT (2024b)	2,280	0	0	0	0	-	-	✗	Long-form QA
VizWiz–LF (2024)	600	600	4,200	0	0	41.2	-	✗	Long-form VQA
RefLVQA (Ours)	81,173 (1,354)	67,140 (1,209)	157,586 (1,369)	5.9 (3.3)	3.2 (2.4)	76.5	Text, Image	✓	Referential long-form VQA & Multimodal retrieval

Table 1: Comparison of long-form question answering (LFQA) benchmarks. Q, I, and A denote the number of unique questions, images, and answers, respectively. D_T/Q and D_I/Q are the average number of text and image documents per question. Sentence-level reference indicates whether cited documents are available at the sentence level. The answer length is to the average word count of answers. The number in parentheses in our dataset indicates the size of the human-verified subset.

ground each claim to its corresponding reference.

As compared in Table 1, prior LFQA research has focused mainly on text modality, in both questions and reference documents. Also, except for LFRQA (Han et al., 2024), existing long-form generation benchmarks either lack supporting documents for answers or provide only coarse-grained document annotations, making it impossible to determine which sentence refers to which document.

To bridge this gap, we propose **Ref**erential **L**ong-form **V**isual **Q**uestion **A**nswering (RefVLQA). We use both automated and human annotation.

3.1 Data Filtering

From Reddit pushshift dumps from 2005-06 to 2023-12, we collect about 6M posts that contain both images and comments. We filter these raw data according to the following rules. First, the title should contain a question starting with a question word (e.g., what, when, which) ending with a question mark. We remove survey questions (e.g., do you, what is your). Second, posts should include long-form answers in comments containing more than 50 words and 3 sentences. Through this process, we obtain 432,817 posts with 182,567 images.

To ensure that the image is necessary to understand the user question, we filter out the posts that can be addressed without the user-given image. Some posts do not require image either because the image is irrelevant to the question (e.g., *Meme pictures*) or the question can be understood without the image (e.g., *What is the orange foil on the Apollo 11 moon lander, and what was it for?*).

Irrelevant image filtering. We further remove instances containing irrelevant images to the questions as follows. We first extract visual and textual features using the InternVL-2.5-38B (Chen et al., 2024b). To automatically determine relevance between the question text and image, we randomly sample 100 distractor images from other posts, and then find out top-10 most similar images to the question using cosine similarity. If the original image is not included, we remove the image, assuming that it is not relevant to the question.

Image-unnecessary question filtering. We further remove instances containing the questions that can be easily understood without the associated image by following the approach of Chen et al. (2024a). We instruct an LLM inspector to generate an answer based solely on the question (without access to the image). If more than one out of the four LLMs—GPT-4o (OpenAI, 2025a), LLaMa-3.3-70B (Meta, 2025), Mixtral 8x7B (Jiang et al., 2024), and Phi-4 14B (Abdin et al., 2024)—produces a plausible answer, we exclude the instance from the dataset. We use InternVL-2.5-38B to automatically decide whether the inspector's answer is correct enough to the dataset answer.

After filtering, 157,586 VQA pairs with 67,140 unique images remain. The prompts used for both inspector and evaluator models are provided in Appendix A.1.

3.2 Supporting Multimodal Documents

Fact-checkable sentence identification. We collect supporting multimodal documents for each

QA pair to provide information that VLMs can draw upon when generating answers. Since each answer is a long-form response that contains multiple pieces of information, we first decompose it into individual sentences using the NLTK sentence tokenizer[1]. Following Li et al. (2023), we then classify whether each sentence requires fact verification using GPT-4o (OpenAI, 2025a). More details on the identification of fact-checkable sentences are provided in Appendix A.2.

Relevant document finding. Each answer instance typically consists of 4.3 sentences requiring verification. For each sentence, we retrieve top-5 external images using Google Search[2] and top-5 relevant documents from Colossal Clean Crawled Corpus (C4) (Raffel et al., 2020), a dataset consisting of hundreds of gigabytes of clean English text scraped from Web.

Document filtering. On average, each QA instance contains 43 relevant documents (4.3 sentences × 5 documents × 2 modalities). Using all these as supporting document candidates poses a burden to human annotators, so we filter out irrelevant ones using entailment models (EMs), which predict whether a document supports each sentence. Specifically, we consider the document as a premise and the sentence as a hypothesis. To balance the trade-off between the volume of documents and the presence of supported sentences, we measure the F1 score of each EM. As shown in Table 11, Qwen3-8B (Qwen Team, 2025) performs well for text documents, while SkyworkVLReward-8B (Wang et al., 2025) performs best for image documents. Using these models, we compute the scores and perform filtering, whose details can be found in Appendix A.4.

Multimodal knowledge base. In the RAG framework, a large-scale knowledge base is crucial for models to retrieve relevant documents for generating their answers. In addition to the documents we collect, we supplement with the WebQA corpus (Chang et al., 2022), since it contains useful information from Wikipedia. The final collection comprises 2.5M documents, including 1.4M multimodal documents of our collection, 1.1M multimodal documents from WebQA (389K image documents and 787K text documents).

Type	Model	F1 Score	
		Text	Image
Textual	NLIDeBERTaV3-184M (2023)	41.4	-
	FlanT5Verifier-11B (2024)	40.7	-
	Qwen3-8B (2025)	**53.7**	-
Visual	OFA-VE-470M (2022)	-	23.1
	SkyworkVLReward-8B (2025)	-	**50.1**
Multi	Qwen2VL-7B (2024)	45.0	49.6

Table 2: Model performances in verifying the groundedness of each sentence on the retrieved documents across different document modalities. Due to computational constraints, we use about 11B open-source models.

Criterion	%
Statements requiring verification	85.7
Image docs. supporting statements	66.1
Text docs. supporting statements	67.6
Supported statements	87.3

Table 3: Results of human annotation on a subset of RefLVQA to assess groundedness of each statement.

3.3 Human Annotation

We conduct human annotation on a subset of the automatically generated dataset. Annotators label randomly sampled QA instances based on three criteria: (1) Do the fact-checkable sentences genuinely require fact verification? (2) Do the supporting documents really support the sentences? (3) Are the sentences accurately grounded in the supporting document(s)?

Table 3 shows that 85.7% of the fact-checkable sentences require verification. Among the supporting documents, 66.1% of image documents and 67.6% of text documents support the corresponding sentences. Although these rates may seem low, it could not be a critical issue in our benchmark as non-supporting documents can act as distractors that the model would avoid retrieving for answering. Overall, 87.3% of the sentences are supported by their corresponding supporting document(s).

Finally, we collect 1,369 QA instances containing 3,382 human-verified fact-checkable sentences supported by 7,825 multimodal documents. We utilize this human-annotated subset to evaluate the model's performance in §5.1 and §5.2. Further details on the annotation procedure and inter-annotator agreement are provided in Appendix A.3.

---

[1] `https://www.nltk.org/api/nltk.tokenize.sent_tokenize.html`

[2] `https://programmablesearchengine.google.com`

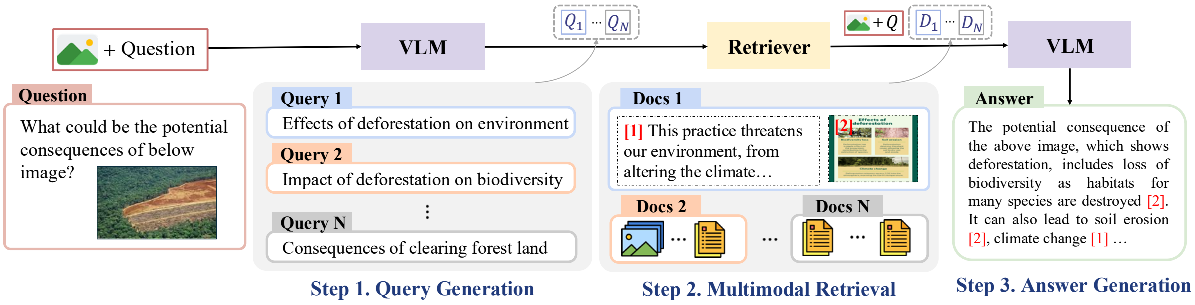

+ Question
VLM
Q_1 ··· Q_N
Retriever
+ Q
D_1 ··· D_N
VLM
Question
What could be the potential consequences of below image?
Query 1
Effects of deforestation on environment
Query 2
Impact of deforestation on biodiversity
Query N
Consequences of clearing forest land
Docs 1
[1] This practice threatens our environment, from altering the climate…
[2]
Docs 2
Docs N
Answer
The potential consequence of the above image, which shows deforestation, includes loss of biodiversity as habitats for many species are destroyed [2]. It can also lead to soil erosion [2], climate change [1] …
Step 1. Query Generation
Step 2. Multimodal Retrieval
Step 3. Answer Generation

Figure 2: Illustration of retrieval-augmented generation (RAG) in long-form visual question answering.

4 RefLVQA Benchmark

The key task of our RefLVQA benchmark is long-form response generation, where the model is required to produce well-grounded answer within a multimodal RAG framework. As shown in Figure 2, the task is addressed in two stages: (1) Query generation and multimodal retrieval: for a question Q with an image I, the VLM $\mathcal{M}$ generates N search queries and lets the retriever fetch the Top-K documents per query. (2) Referential answer generation: Given $N \cdot K$ multimodal documents retrieved, the VLM generates a final long-form answer. See Appendix C.1 for more detail in prompt templates.

4.1 Citation-based Evaluation

Inspired by citation accuracy in the LFRQA (Han et al., 2024), we propose a citation-based evaluation method in which models directly predict citation numbers (e.g., [1]) referring to the retrieved multimodal documents within their answers. Unlike LFRQA, where LLM is instruct to create coherent long-form answers from gold short-form answers, our approach introduces three key differences.

First, we evaluate the model's generative ability within the RAG framework to support its responses. Second, we allow the model to autonomously craft search queries to retrieve relevant documents instead of relying on a predefined static document set. Third, our method extends the scope from text-only to multimodal documents, requiring the model to ground its responses in both text and images.

4.2 Evaluation Metrics

Since each long-form answer includes many sentences, making binary judgments for the answer is inadequate for correct evaluation (Min et al., 2023). Therefore, we propose three types of fine-grained evaluation metrics as follows.

1. **Groundedness**: Groundedness assesses how well each sentence a_i in the model's answer A is supported by the cited document(s) D. First, we split A into sentences $\{a_1, \dots a_n\}$ and identify D via citation number(s) for each a_i. Then, evaluators rate groundedness of each pair $g(a_i, D)$ as fully supported (1), partially supported (0.5), or not supported (0). Fully supported means that D provides sufficient evidence to verify the factuality of a_i. Partially supported means D offers some but insufficient evidence. Not supported means there is no evidence or no cited documents. The final groundedness score for each A is calculated as the averaged $g(a_i, D)$ over all sentences in A.
2. **Completeness**: Completeness measures how well the model's answer A addresses all necessary information to the question. Specifically, we assesses the degree to which A covers the fact-checkable sentences $\{r_1, \dots, r_m\}$ in the dataset answer R, as annotated in §3.2 and 3.3. Evaluators rate the completeness of each pair $c(r_j, A)$ as fully addressed (1), partially addressed (0.5), or not addressed (0). Fully addressed means that A considers r_j directly and clearly. Partially addressed means that A mentions or implies r_j but does not cover it fully or clearly. Not addressed means that A does not mention or consider r_j at all. The final completeness score for each A is calculated by averaging $c(r_j, A)$ across all fact-checkable sentences in R annotated in our dataset.
3. **Relevance**: Relevance evaluates how well the model's answer A aligns with the question Q. Specifically, we assess whether A contains only helpful information without any unnecessary content. Evaluators rate the relevance of each pair $r(Q, A)$ on a 1–7 Likert scale, where 7 indicates a helpful answer without unnecessary information, and 1 indicates an answer that fails to provide any relevant information. Evaluators penalize answers that

Retriever	NDCG@10	Recall@100
Fine-tuned on WebQA (Chang et al., 2022)		
CLIP-DPR	0.1567	0.4355
UniVL-DR	0.1136	0.3244
MARVEL-DPR	0.1292	0.4188
MARVEL-ANCE	0.1322	0.3948
Fine-tuned on ClueWeb (Overwijk et al., 2022)		
MARVEL-DPR	0.1098	0.4357
MARVEL-ANCE	0.1460	0.4398
Fine-tuned on M-BEIR (Wei et al., 2024a)		
MM-Embed		
+ text-seeking query	0.2216	0.5909
+ image-seeking query	<u>0.2217</u>	<u>0.6074</u>
+ averaged query embedding	**0.2565**	**0.6977**

Table 4: Multimodal retrieval performance on the human annotated test set.

include unnecessary information.

We utilize human evaluators and GPT-4.1 (OpenAI, 2025b) as evaluators. Further details on the evaluation instructions are provided in Appendix B.

5 Experiments

We evaluate state-of-the-art VLMs with multimodal retrievers in the RefLVQA benchmark, using the evaluation metrics described in §4.2.

5.1 Model Details

Multimodal Retrievers. We explore various dense retrievers for the RAG framework, such as CLIP-DPR (Liu et al., 2022), UniVL-DR (Liu et al., 2022), MARVEL (Zhou et al., 2023), CLIP-SF (Wei et al., 2024a), and MM-Embed (Lin et al., 2024). MM-Embed is a modality-aware retriever where the retrieval modality should be chosen in advance. As shown in Table 4, MM-EMBED (with averaged query embedding of text and images) achieves the highest performance in both NDCG@10 and Recall@100. Thus, we choose MM-EMBED as the default retriever.

Multimodal Rerankers. To find Top-K documents, we take a re-ranking approach; after the retriever finds out the top-100 documents, from which the reranker selects the top-K documents. We use JINA-RERANKER-M0[3] as our multimodal reranker. As Figure 3 shows Hit@K sharply increases up to $K = 5$, we set $K = 5$ as the cutoff, balancing the supportedness of statements (almost 53%) with the input context length constraints of VLMs.

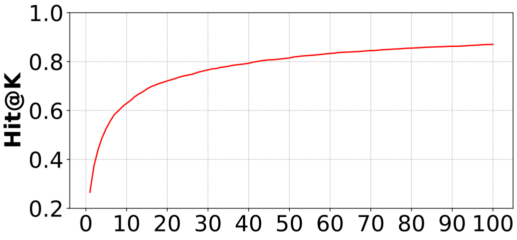


Figure 3: Hit@K for MM-Embed (Lin et al., 2024) with multimodal reranking.

Vision-Language Models. We select four contemporary VLMs as answer generators. For proprietary models, we use (1) GPT-4O-240806 (OpenAI, 2025a), and (2) CLAUDE-3.5-SONNET-20241022 (Anthropic, 2024). For public models, we use (3) INTERNVL2.5-78B (Chen et al., 2024b), and (4) QWENVL-72B (Wang et al., 2024). See Appendix C.2 for more implementation details.

RAG Baselines. We evaluate three types of RAG settings: Text-RAG, Image-RAG, and Multimodal-RAG. In the uni-modal RAG settings, the model retrieves documents from only one modality. To investigate the impact of retrieved document diversity, we compare a single query retrieval ($N = 1$) with a multiple query retrieval ($N = 4$). For each query, we retrieve Top-5 ($K = 5$) documents.

5.2 Automatic Evaluation Results

Table 5 reports the results of automatic evaluation. We first compare the unimodal and multimodal RAG baselines in the single query retrieval setting.

Document Groundedness and Utilization. Results show that vision-language models (VLMs) face challenges in effectively utilizing image documents compared to text documents. Specifically, the groundedness of generated answers is significantly lower when relying on image inputs; for instance, GPT-4o with ImageRAG produces only 22.5% fully grounded sentences versus 65.2% for TextRAG. Similarly, the average groundedness score for image documents is 44.5%, notably less than the 73.2% for text documents. This discrepancy reflects a tendency of models to generate less accurate or less verifiable content from images. Furthermore, document utilization in MultiRAG settings is imbalanced, with image documents being underutilized — only 25.1% (0.39 out of 1.55) of image documents are used, compared to 42.0% (3.04 out of 5) in the image-only setting. This suggests an over-reliance on textual informa-

---

[3] `https://huggingface.co/jinaai/jina-reranker-m0`

	Evaluation Metrics							Statistics			
	Groundedness			Completeness			Relevance	Retrieved Docs.		Used Docs.	
	Mean	@1.0	@0.5	Mean	@1.0	@0.5	(1-7)	Text	Image	Text	Image
InternVL2.5											
+ ImageRAG	0.372	0.213	0.531	0.368	0.149	0.587	4.878	0.00	5.00	0.00	2.25
+ TextRAG	**0.635**	**0.552**	**0.718**	**0.380**	0.149	**0.611**	**5.977**	5.00	0.00	3.24	0.00
Qwen2.5VL											
+ ImageRAG	0.442	0.345	0.538	0.333	0.130	0.536	4.329	0.00	5.00	0.00	2.49
+ TextRAG	0.765	**0.661**	0.868	0.377	0.158	0.595	6.062	5.00	0.00	2.83	0.00
+ MultiRAG	0.693	0.581	0.804	0.378	0.158	0.598	5.918	3.48	1.51	2.02	0.58
+ Two-step MultiRAG	**0.776**	0.654	**0.899**	**0.476**	**0.248**	**0.704**	**6.306**	3.48	1.51	2.22	0.89
GPT-4o											
+ ImageRAG	0.445	0.225	0.530	0.359	0.141	0.573	4.357	0.00	5.00	0.00	2.10
+ TextRAG	**0.732**	0.652	**0.859**	0.360	0.143	0.577	6.060	5.00	0.00	2.50	0.00
+ MultiRAG	0.731	**0.656**	0.858	0.367	0.128	0.606	6.072	3.44	1.55	1.97	0.39
+ Two-step MultiRAG	0.706	0.600	0.812	0.473	0.221	0.723	6.441	3.44	1.55	1.98	0.87
+ Two-step MultiRAG ($N = 4$)	0.685	0.558	0.812	**0.493**	**0.240**	**0.746**	**6.676**	14.27	5.72	3.56	1.52
Claude-3.5-Sonnet											
+ ImageRAG	0.521	0.410	0.630	0.372	0.161	0.587	5.318	0.00	5.00	0.00	3.04
+ TextRAG	**0.783**	**0.663**	**0.903**	0.375	0.166	0.583	5.977	5.00	0.00	3.56	0.00
+ MultiRAG	0.760	0.654	0.864	0.372	0.161	0.584	5.932	3.46	1.53	2.59	0.60
+ Two-step MultiRAG	0.779	0.662	0.896	0.515	**0.307**	0.723	6.077	3.46	1.53	2.57	1.10
+ Two-step MultiRAG ($N = 4$)	0.708	0.589	0.828	**0.517**	0.290	**0.743**	**6.485**	14.35	5.63	5.45	2.25

Table 5: Automatic evaluation under the single retrieval setting ($N = 1, K = 5$) and multiple retrieval setting ($N = 4, K = 5$). If N is not specified, the single retrieval setting ($N = 1$) is assumed. Bold numbers indicate the best performance, and underlined numbers indicate the second-best. The @K columns represent the proportion of scores higher than K, while the Mean column shows the average score for each metric. In statistics, Retrieved Docs. and Used Docs. denote the number of retrieved and used documents for each baseline.

tion when multimodal data is concatenated naïvely. These findings highlight the dual challenges of low groundedness and poor utilization of image documents, which together limit the effectiveness of multimodal retrieval-augmented generation.

Two-step MultiRAG To address these challenges, we propose *Two-step MultiRAG*, which first generates answers separately from text and image documents and then combines them within the model. Additionally, we employ image captioning to improve groundedness by providing captions alongside image documents. Detailed instructions are provided in Appendix C.1.

Through the Two-step MultiRAG approach, VLMs utilize image documents more frequently than before. Two-step MultiRAG outperforms unimodal RAG baselines in terms of answer completeness and relevance, highlighting the advantages of leveraging multimodal documents over relying exclusively on a single modality. However, the current Two-step MultiRAG method has several limitations: (1) it does not simultaneously consider both modalities when generating answers, and (2) it introduces computational inefficiencies. Addressing these limitations remains an important area for future research.

Multiple Queries Retrieval We compare single query retrieval and multiple queries retrieval baselines within Two-step MultiRAG. As demonstrated in the main results, multiple queries retrieval baselines consistently outperform single query retrieval in terms of completeness and relevance. However, performance in groundedness decreases compared to single query retrieval, suggesting that an increased number of documents may reduce the model's grounding ability. These findings suggest that while multifaceted retrieval improves overall answer quality, it may come at the cost of grounding performance.

5.3 Human Evaluation Results

We performed a human evaluation to compare model-based evaluations against human judgments. Qwen2.5-VL-72B and GPT-4o each generated 100 answers from three RAG frameworks (ImageRAG, TextRAG, MultiRAG), resulting in a total of 600 answers. Human raters, consisting of three participants as detailed in Appendix B.2, were asked to rate the answers based on groundedness, completeness, and relevance. As shown in Table 6, all Pearson correlation coefficients for groundedness, completeness, and relevance are above 0.733.

		Evaluator		Pearson
Metric	Answer	Human	GPT-4.1	Corr.
Grd	ImageRAG	0.491	0.443	0.773
	TextRAG	**0.691**	**0.687**	
	MultiRAG	0.672	0.672	
Com	ImageRAG	0.258	0.342	0.733
	TextRAG	**0.296**	0.368	
	MultiRAG	0.284	**0.377**	
Rel	ImageRAG	5.200	4.222	0.855
	TextRAG	**5.993**	**6.022**	
	MultiRAG	5.662	5.860	

Table 6: Results of human and model-based evaluation on RefLVQA. Grd, Com, and Rel indicate mean scores of groundedness, completeness, and relevance, respectively. We report Pearson correlation between human evaluators and GPT-4.1 for each metric.

These strong correlations demonstrates the reliability of automatic long-form answer evaluation using VLMs.

6 Analysis

6.1 Image Grounding Errors

As shown in §5.2 and §5.3, deficient image grounding ability is one of the biggest hurdles for multimodal RAG. To better understand the limitations of VLMs in image grounding, we manually analyzed 230 image grounding errors made by GPT-4o and Qwen2.5VL. These errors were labeled as either partially grounded or not grounded in §5.3. We carefully defined the three most frequent categories of errors as follows: (1) No evidence: The model generate answer for which there is no evidence in the image, (2): Ommision the model fails to recognize information that is actually present in the image, and (3) Overgeneralization: the model overgeneralizes from specific cases to draw general conclusions. As shown in Figure 4, the models frequently generate content that is not present in the image (55.75%) rather than omit (23.08%) or over-generalize visual content (16.90%).

6.2 Image Grounding Improvement

To enhance the image grounding ability of VLMs, we explore three types of inference-time scaling methods: zero-shot CoT (Kojima et al., 2022), self-refine (Madaan et al., 2023), and image captioning. We adjusted the prompting for each method to make it robust against common image grounding error cases discussed in §6.1. See Appendix C.1 for a more detailed explanation of each method. As shown in Table 7, the error-robust image captioning method outperforms the other two prompting methods and ImageRAG in terms of groundedness and relevance, while also demonstrating completeness comparable to that of CoT.

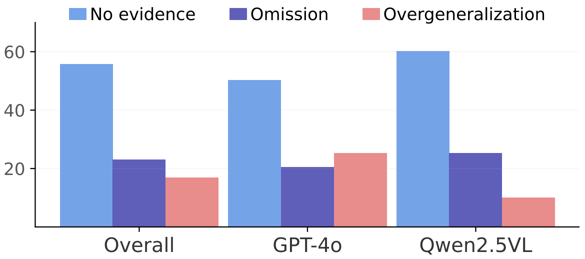


Figure 4: Distribution of the three most frequent image grounding error types made by GPT-4o and Qwen2.5VL. Examples of each error type are presented in Table 19.

	Grd	Com	Rel	# IMG
ImageRAG	0.445	0.359	4.357	2.10
+ CoT	0.412	**0.399**	4.817	1.42
+ Self-Refine	0.437	0.394	5.963	2.19
+ Captioning	**0.470**	0.391	**6.302**	2.12

Table 7: Automatic evaluation under the single retrieval setting ($N = 1, K = 5$) using GPT-4o. Grd, Com, and Rel denote the mean scores for groundedness, completeness, and relevance, respectively. # IMG indicates the number of utilized image documents out of 5.

7 Conclusion

In conclusion, we introduce RefLVQA, the first large-scale dataset designed to evaluate the long-form answer generation capabilities of large vision-language models using visual questions and multimodal documents. RefLVQA contains 157K visual question-answering instances, each supported by sentence-level annotations within multimodal documents. To assess model performance, we propose a citation-based evaluation framework that requires models to provide citation numbers referencing the supporting documents. Our findings indicate that (1) multimodal RAG methods face challenges due to over-reliance on textual documents and limited image grounding ability; (2) our proposed method, Two-step MultiRAG, outperforms unimodal approaches, demonstrating the advantage of utilizing multimodal documents for generating grounded answers; and (3) error-robust image captioning of image documents leads to enhanced image grounding ability.

Limitations

We acknowledge a few potential limitations of our research. (1) In this study, we did not cover frameworks that generate responses by simultaneously considering multimodal documents, as mentioned in §5.2. To address the challenges of naïve multimodal retrieval-augmented generation (RAG), we employed a framework that generates answers for each modality separately and then integrates them. Future work could explore frameworks that jointly consider multiple modalities when generating responses. (2) Our study primarily focused on text and image documents; therefore, the application and evaluation of our approach on other types of multimodal external documents, such as video and audio, remain unexplored. (3) In §6.2, we only explored inference-time scaling methods, which incur high computational costs. Future research could investigate more efficient methods to improve image grounding ability. (4) Our data generation pipeline automatically collects external documents. However, in our framework, if the external document retrieval fails, we do not attempt to re-collect them to maintain efficiency.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Anthropic. 2024. Introducing claude 3.5 sonnet.

Valeriia Bolotova-Baranova, Vladislav Blinov, Sofya Filippova, Falk Scholer, and Mark Sanderson. 2023. Wikihowqa: A comprehensive benchmark for multi-document non-factoid question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5291–5314.

Odellia Boni, Guy Feigenblat, Guy Lev, Michal Shmueli-Scheuer, Benjamin Sznajder, and David Konopnicki. 2021. Howsumm: A multi-document summarization dataset derived from wikihow articles. *arXiv preprint arXiv:2110.03179*.

Yingshan Chang, Mridu Narang, Hisami Suzuki, Guihong Cao, Jianfeng Gao, and Yonatan Bisk. 2022. Webqa: Multihop and multimodal qa. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16495–16504.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. 2024a. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. 2024b. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. Eli5: Long form question answering. *arXiv preprint arXiv:1907.09190*.

Rujun Han, Yuhao Zhang, Peng Qi, Yumo Xu, Jenyuan Wang, Lan Liu, William Yang Wang, Bonan Min, and Vittorio Castelli. 2024. Rag-qa arena: Evaluating domain robustness for long-form retrieval augmented question answering. *arXiv preprint arXiv:2407.13998*.

Mina Huh, Fangyuan Xu, Yi-Hao Peng, Chongyan Chen, Hansika Murugu, Danna Gurari, Eunsol Choi, and Amy Pavel. 2024. Long-form answers to visual questions from blind and low vision people. *arXiv preprint arXiv:2408.06303*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. 2016. A diagram is worth a dozen images. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 235–251. Springer.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Sayali Kulkarni, Sheide Chammas, Wan Zhu, Fei Sha, and Eugene Ie. 2020. Aquamuse: Automatically generating datasets for query-based multi-document summarization. *arXiv preprint arXiv:2010.12694*.

Moritz Laurer, Wouter van Atteveldt, Andreu Casas, and Kasper Welbers. 2023. Building Efficient Universal Classifiers with Natural Language Inference. ArXiv:2312.17543 [cs].

Miaoran Li, Baolin Peng, Michel Galley, Jianfeng Gao, and Zhu Zhang. 2023. Self-checker: Plug-and-play modules for fact-checking with large language models. *arXiv preprint arXiv:2305.14623*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Sheng-Chieh Lin, Chankyu Lee, Mohammad Shoeybi, Jimmy Lin, Bryan Catanzaro, and Wei Ping. 2024. Mm-embed: Universal multimodal retrieval with multimodal llms. *arXiv preprint arXiv:2411.02571*.

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

Zhenghao Liu, Chenyan Xiong, Yuanhuiyi Lv, Zhiyuan Liu, and Ge Yu. 2022. Universal vision-language dense retrieval: Learning a unified representation space for multi-modal retrieval. *arXiv preprint arXiv:2209.00179*.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2023. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594.

Meta. 2025. Llama. `https://www.llama.com//`.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*.

OpenAI. 2025a. Hello gpt-4o. `https://openai.com/index/hello-gpt-4o/`.

OpenAI. 2025b. Introducing gpt-4.1 in the api.

Arnold Overwijk, Chenyan Xiong, Xiao Liu, Cameron VandenBerg, and Jamie Callan. 2022. Clueweb22: 10 billion web documents with visual and semantic information. *arXiv preprint arXiv:2211.15848*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Qwen Team. 2025. Qwen3: Think deeper, act faster.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Soumya Sanyal, Tianyi Xiao, Jiacheng Liu, Wenya Wang, and Xiang Ren. 2024. Are machines better at complex reasoning? unveiling human-machine inference gaps in entailment verification. *arXiv preprint arXiv:2402.03686*.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. 2022. A-okvqa: A benchmark for visual question answering using world knowledge. In *European conference on computer vision*, pages 146–162. Springer.

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

Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. 2022. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International conference on machine learning*, pages 23318–23340. PMLR.

Xiaokun Wang, Chris, Jiangbo Pei, Yunzhuo Hao, Weijie Qiu, Ai Jian, Tianyidan Xie, Xuchen Song, Yang Liu, and Yahui Zhou. 2025. Skywork-vl reward: An effective reward model for multimodal understanding and reasoning.

Cong Wei, Yang Chen, Haonan Chen, Hexiang Hu, Ge Zhang, Jie Fu, Alan Ritter, and Wenhu Chen. 2024a. Uniir: Training and benchmarking universal multimodal information retrievers. In *European Conference on Computer Vision*, pages 387–404. Springer.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, et al. 2024b. Long-form factuality in large language models. *arXiv preprint arXiv:2403.18802*.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567.

Tianshuo Zhou, Sen Mei, Xinze Li, Zhenghao Liu, Chenyan Xiong, Zhiyuan Liu, Yu Gu, and Ge Yu. 2023. Marvel: unlocking the multi-modal capability of dense retrieval via visual module plugin. *arXiv preprint arXiv:2310.14037*.

A Data Collection Details

A.1 Instruction for Inspector and Evaluator

The instructions for the inspectors and the evaluator are shown in Table 8 and Table 9.

A.2 Identification of Fact-checkable Sentences

We utilize LLM-based identification of fact-checkable sentences. Using the prompt described in Table 10, we input each sentence individually. If a sentence contains more than one distinct claim, we consider it a fact-checkable sentence.

A.3 Human Annotation

We hired data annotators via Amazon Mechanical Turk (MTurk). Five annotators were selected based on their performance in a qualification task designed to assess their ability to determine whether statements are accurately supported by the given documents. We required annotators to be from English-speaking countries (AU, CA, NZ, US, GB), have completed more than 10,000 HITs, and maintain a HIT approval rate above 98%. The qualification task consisted of 10 examples (30 questions in total) and paid $5.00 per qualification task. Each qualification task included three questions as illustrated in Figures 5, 6, 7, and 8.

1. **Is verification genuinely required?** Determine whether the statement is self-evident or based on common sense and thus does not require verification, to avoid unnecessary validation.

2. **Do the supporting documents actually support the statements?** Assess whether the documents retrieved and filtered by the automated system genuinely support the given statements.

3. **Are the statements accurately grounded in the supporting documents?** Verify if each statement is precisely grounded by referencing one or more external documents.

We randomly extracted 2,000 QA instance pairs, consisting of almost 4,000 sentences (instances without supporting documents were removed). Annotators labeled each pair using the three-question format described above. If annotators labeled question (1) as *false* (verification not required), they skipped the remaining two questions for that pair. For question (3), up to three documents were provided for each sentence.

We measured inter-annotator agreement on a subset of 100 pairs in advance. Fleiss' κ scores for binary classification were 0.75 for question (1), 0.65 for question (2), and 0.80 for question (3).

A.4 Entailment Model

To identify the groundedness of each statement with respect to the corresponding document, we treat the document as the premise and the statement as the hypothesis. If an entailment model outputs the label "entailment," we consider the statement to be grounded. We use different entailment models depending on the modality of the document.

Textual Entailment Models. We consider the following textual entailment models: NLIDeBERTaV3-184M(Laurer et al., 2023), FlanT5Verifier-11B(Sanyal et al., 2024), and Qwen3-8B (Qwen Team, 2025).

For NLIDeBERTaV3-184M, we use the `text-classification` pipeline from the `transformers` library[4]. The model classifies input into one of three labels: `entailment`, `neutral`, or `contradiction`.

For FlanT5Verifier-11B, we use the following prompt template:

```
Premise: {premise} Hypothesis:
{hypothesis} Given the premise,
is the hypothesis correct?
Answer:
```

We then compute token probabilities for "`Yes`" and "`No`". If "`Yes`" has a higher probability, we classify the pair as `entailment`; otherwise, we classify it as `not entailment`.

For Qwen3-8B, we use a similar prompt:

```
Premise: {premise} Hypothesis:
{hypothesis} Given the premise,
is the hypothesis correct?
Respond in yes or no. Answer:
```

If the model outputs "yes", we treat the pair as `entailment`; otherwise, as `not entailment`.

Visual Entailment Models. We consider the following visual entailment models: OFA-VE-470M (Wang et al., 2022) and SkyworkVLReward-8B (Wang et al., 2025).

For OFA-VE-470M, we use the visual entailment pipeline from the ModelScope library[5]. The model is prompted with:

---

[4] `https://huggingface.co/docs/transformers/en/tasks/sequence_classification`

[5] `https://github.com/modelscope/modelscope`

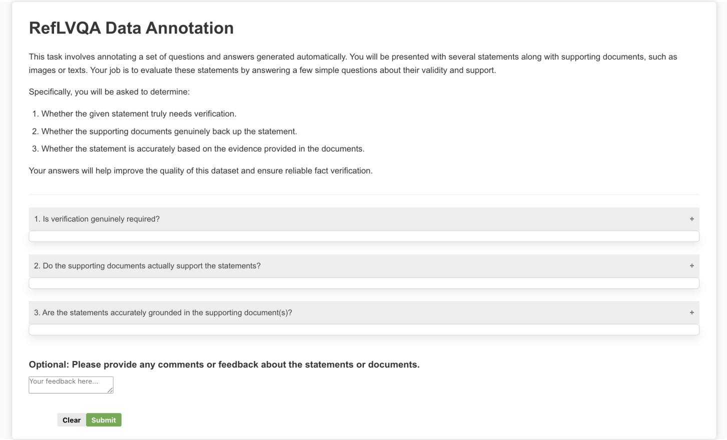
Instruction
Instruction: 1. Given a question, your task is to generate an answer. 2. Even if describing the image seems impossible without viewing it, you should predict the situation and describe it accordingly. 3. Only generate answer. Question: {question}

Table 8: Instruction for inspector.

Instruction

Instructions:
1. Given an image, a question, a gold answer, and a model response, your task is to evaluate whether the model response is “right” or “wrong”.
2. Even if the model response differs from the gold answer, if the model appears to have correctly understood the image, label the response as “right”.
 	
Question: <image>{question}
Gold answer: {gold_answer}
Model response: {model_response}

Table 9: Instruction for evaluator.



RefLVQA Data Annotation

This task involves annotating a set of questions and answers generated automatically. You will be presented with several statements along with supporting documents, such as images or texts. Your job is to evaluate these statements by answering a few simple questions about their validity and support.

Specifically, you will be asked to determine:

1. Whether the given statement truly needs verification.
2. Whether the supporting documents genuinely back up the statement.
3. Whether the statement is accurately based on the evidence provided in the documents.

Your answers will help improve the quality of this dataset and ensure reliable fact verification.

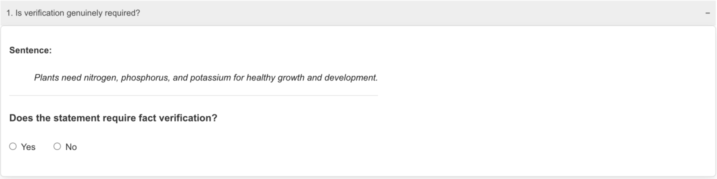
1. Is verification genuinely required? +

2. Do the supporting documents actually support the statements? +

3. Are the statements accurately grounded in the supporting document(s)? +

Optional: Please provide any comments or feedback about the statements or documents.

Figure 5: Instructions provided for human evaluators to obtain.



1. Is verification genuinely required? –

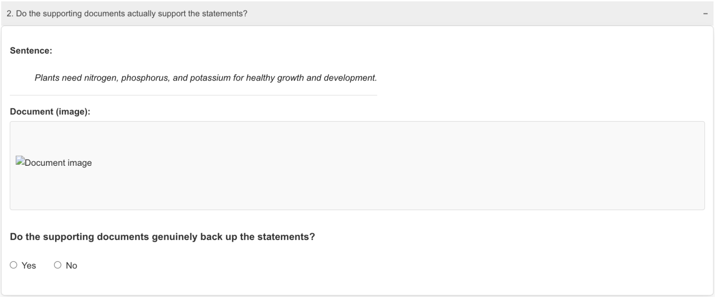
Sentence:

Plants need nitrogen, phosphorus, and potassium for healthy growth and development.

Does the statement require fact verification?

○ Yes ○ No

Figure 6: Instructions template provided for human evaluators to obtain labels for verification requirement.



2. Do the supporting documents actually support the statements? –

Sentence:

Plants need nitrogen, phosphorus, and potassium for healthy growth and development.

Document (image):

Document image

Do the supporting documents genuinely back up the statements?

○ Yes ○ No

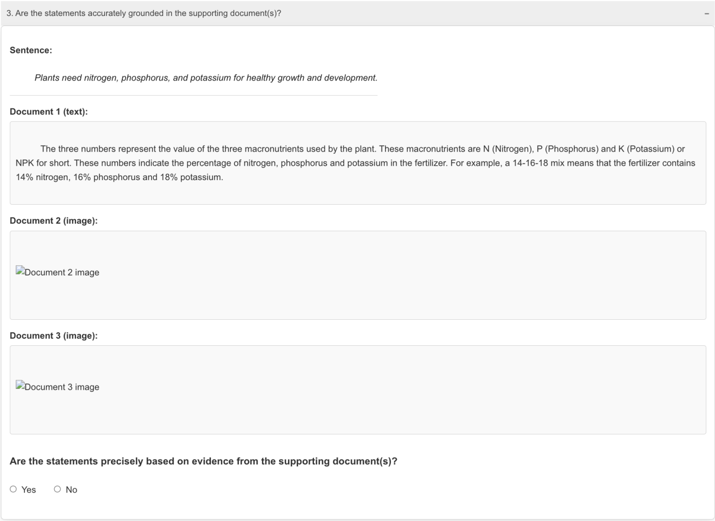
Figure 7: Instructions provided for human evaluators to obtain labels for document supportedness.

`Statement: {statement} Is this statement right according to the image? Please answer yes or no.`

We classify the image-statement pair as `entailment` if the model outputs “yes”, and `not entailment` otherwise.

Instruction
You and your partners are on a mission to fact-check a claim that may contain multiple subclaims that need to be verified. A sentence that needs to be verified is any statement or assertion that requires evidence or proof to support its accuracy or truthfulness. For example, “Titanic was first released in 1997” necessitates verification of the accuracy of its release date, whereas a claim like "Water is wet" does not warrant verification. Each subclaim is a simple, complete sentence with single point to be verified. Imagine yourself as an expert in processing complex paragraphs and extracting subclaims. Your task is to extract clear, unambiguous subclaims to check from the input paragraph, avoiding vague references like ’he,’ ’she,’ ’it,’ or ’this,’ and using complete names. To illustrate the task, here are some examples: {in-context examples} Now, let’s return to your task. You are given the following input paragraph, please extract all subclaims that need to be checked. Input: {input} Subclaims: {extracted claims}.

Table 10: Instruction for claim processor from Li et al. (2023).



3. Are the statements accurately grounded in the supporting document(s)?

Sentence:

Plants need nitrogen, phosphorus, and potassium for healthy growth and development.

Document 1 (text):

The three numbers represent the value of the three macronutrients used by the plant. These macronutrients are N (Nitrogen), P (Phosphorus) and K (Potassium) or NPK for short. These numbers indicate the percentage of nitrogen, phosphorus and potassium in the fertilizer. For example, a 14-16-18 mix means that the fertilizer contains 14% nitrogen, 16% phosphorus and 18% potassium.

Document 2 (image):

Document 2 image

Document 3 (image):

Document 3 image

Are the statements precisely based on evidence from the supporting document(s)?

○ Yes ○ No

Figure 8: Instructions provided for human evaluators to obtain labels for sentence groundedness.

For SkyworkVLReward-8B, we adopt a reward-based scoring approach. Given a premise image and a textual hypothesis, we prompt the model with:

```
Determine whether the
image entails the statement
"{statement}". A. Yes. B. No.
```

We compute separate reward scores for the completions A. Yes. and B. No.” The option with the higher reward score determines the final prediction.

Multimodal Entailment Model. We use Qwen2VL-7B (Wang et al., 2024) as a multimodal entailment model. It is prompted as follows:

```
Premise: {premise} Hypothesis:
{hypothesis} Given the premise,
is the hypothesis correct?
Respond in yes or no. Answer:
```

If the model outputs “yes”, we classify the image-hypothesis pair as `entailment`; otherwise, as `not entailment`.

B Evaluation Details

B.1 Model-based Evaluation

In this section, we explain our instruction templates for automatic evaluation using GPT-4.1 (OpenAI, 2025b). For groundedness, see Table 11. For completeness, see Table 12. For relevance, see Table 13.

B.2 Human Evaluation

To verify the quality of the model-based automatic evaluation used in §5.2, we conducted a human evaluation with three graduate students selected through a qualification task. This task involved rating 10 model-generated answers based on groundedness, completeness, and relevance. On average, participants spent about 10 minutes per answer and were compensated $15.00 for completing the qualification. Instructions for the human evaluation example is shown in Figure 9.

Following the qualification, each human evaluator assessed 200 model answers from the perspectives of groundedness, completeness, and relevance. Each answer was evaluated by a two human evalu-

Instruction

Instruction:
1. You will be given a question, a statement, and an external document.
2. First, extract all subclaims within the statement that need verification.
3. Assess how well each subclaim is supported by the document.
4. Assign one of the following labels: "fully support," "partially support," or "not support."
- If all subclaims are supported by the document, select "fully support."
- If only some of the subclaims are supported, select "partially support."
- If none of the subclaims are supported, select "not support."

Important:
Provide a brief explanation for your chosen level of support. The final answer should begin with "Answer: ".

Statement: {statement}
Documents: {document}

Table 11: Prompt template for groundedness evaluator.

Instruction

Instruction:
1. You will be given a response and a statement.
2. First, identify all subclaims within the statement that require verification.
3. Evaluate how thoroughly each subclaim is addressed in the response.
4. Assign one of the following labels: "fully complete," "partially complete," or "not complete."
- Fully complete: Statement is fully addressed, and all subclaims are verified.
- Partially complete: Only some of the subclaims are addressed.
- Not complete: None of the subclaims are addressed.

Important:
Provide a brief explanation for your chosen level of completeness. The final answer should begin with "Answer: ".

Response: {answer}
Statement: {statement}

Table 12: Prompt template for completeness evaluator.

Instruction

Evaluation Criteria:

Relevance (1-7) – measures how much the answer sentences are semantically aligned with the question.
The answer should directly address the question by providing information that is closely related and relevant.
Sentences in the answer that do not correspond to or deviate from the question reduce the relevance score.

Evaluation Steps:

1. Read the question carefully to understand what is being asked.
2. Read the answer and evaluate how well the sentences in the answer semantically correspond to the question.
3. Assign a relevance score on a scale from 1 to 7, where:
- 1 means the answer is mostly irrelevant or off-topic,
- 7 means the answer is highly relevant and fully aligned with the question.

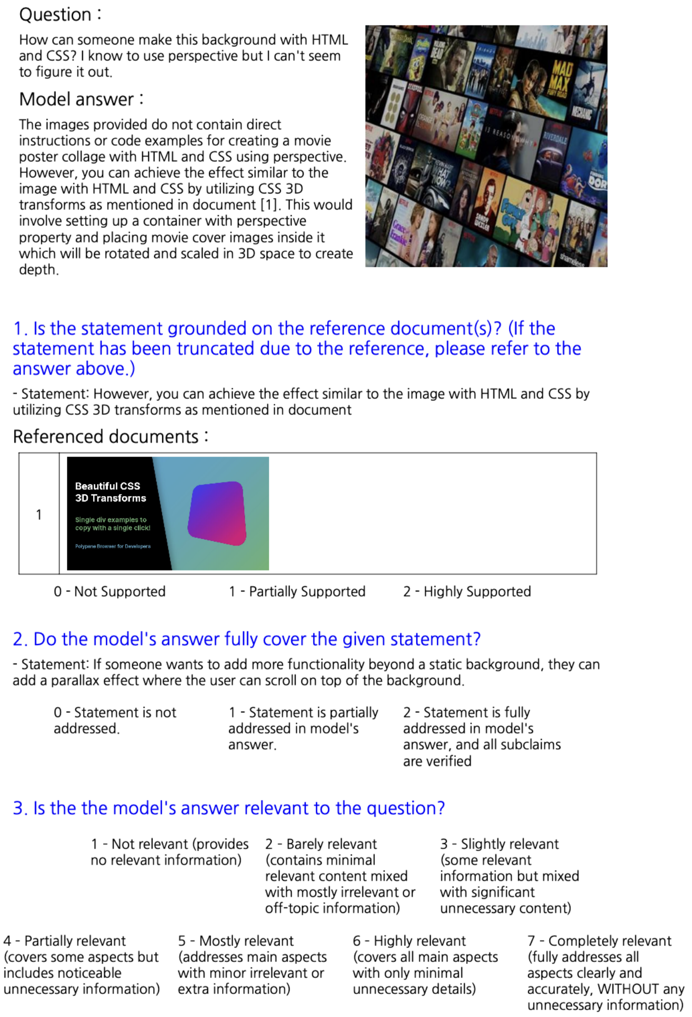
Example:

Question:
{Question}

Answer:
{Answer}

Evaluation Form (scores ONLY):
- Relevance:

Table 13: Prompt template for relevance evaluator.


Beautiful CSS 3D Transforms
Single div examples to copy with a single click!
Polypane Browser for Developers

Question :

How can someone make this background with HTML and CSS? I know to use perspective but I can't seem to figure it out.

Model answer :

The images provided do not contain direct instructions or code examples for creating a movie poster collage with HTML and CSS using perspective. However, you can achieve the effect similar to the image with HTML and CSS by utilizing CSS 3D transforms as mentioned in document [1]. This would involve setting up a container with perspective property and placing movie cover images inside it which will be rotated and scaled in 3D space to create depth.

1. Is the statement grounded on the reference document(s)? (If the statement has been truncated due to the reference, please refer to the answer above.)

- Statement: However, you can achieve the effect similar to the image with HTML and CSS by utilizing CSS 3D transforms as mentioned in document

Referenced documents :

1

0 - Not Supported
1 - Partially Supported
2 - Highly Supported

2. Do the model's answer fully cover the given statement?

- Statement: If someone wants to add more functionality beyond a static background, they can add a parallax effect where the user can scroll on top of the background.

0 - Statement is not addressed.
1 - Statement is partially addressed in model's answer.
2 - Statement is fully addressed in model's answer, and all subclaims are verified

3. Is the the model's answer relevant to the question?

1 - Not relevant (provides no relevant information)
2 - Barely relevant (contains minimal relevant content mixed with mostly irrelevant or off-topic information)
3 - Slightly relevant (some relevant information but mixed with significant unnecessary content)
4 - Partially relevant (covers some aspects but includes noticeable unnecessary information)
5 - Mostly relevant (addresses main aspects with minor irrelevant or extra information)
6 - Highly relevant (covers all main aspects with only minimal unnecessary details)
7 - Completely relevant (fully addresses all aspects clearly and accurately, WITHOUT any unnecessary information)

Figure 9: Human evaluation example provided for human evaluators to obtain labels for groundedness, completeness, and relevance.

ator. Participants were paid $1.50 for each model answer they evaluated.

Inter-annotator agreement (IAA)—excluding the authors' ratings—was measured using averaged Cohen's κ, yielding 0.588 for groundedness, 0.648 for completeness, and 0.659 for relevance. For relevance, we categorized human answers into three groups: Not Relevant (scores 1–3), Partially Relevant (scores 4–5), and Relevant (scores 6–7) prior to measuring agreement. Finally, we used the average score of the two annotators as the human evaluation result and compared it with the model-based evaluation results.

C Experimental Details

C.1 Details in Prompting

Task Instruction. In this section, we explain our task instruction templates. For the query generation, refer to Table 14. For the referential answer generation, refer to Table 15.

Image Captioning Prompting. For the image captioning method, as shown in Table 16, we instruct the model to extract factual information from the image documents. The generated image captions are then concatenated immediately after the image document.

Answer Integration Prompting. To merge the two answers obtained from each modality document into a coherent single response, we applied the prompt template shown in Table 17. For sentences expressing the same claim in both answers, we combined them into a single sentence and included the citation numbers together.

Zero-shot Chain-of-Thought Prompting. For zero-shot CoT prompting, we follow OpenAI's rec-

Instruction

<Instruction>
1. Based on the given image and question, generate {N} search queries.
2. Formulate queries to retrieve documents that provide information to generate the answer.
3. List the generated search queries separated by commas. For example: “query 1”, “query 2”, ...

Question: <image>\n{question}
Search queries:

Table 14: Instruction for query generation.

Instruction

Based on the documents, provide a helpful answer to the query. Your answer must be faithful to the content in the documents.
You should cite the passage number (indices) in the format of [1], [2], [3, 4], etc. at the end of each sentence.
Do not include sentences that are not supported by the documents.

Question: <image>{question}
Document:
...

Answer:

Table 15: Instruction for referential answer generation.

Instruction

You are a powerful image captioner. Extract all factual and observable information from the image. Instead of describing the imaginary content, only describing the content one can determine confidently from the image. Do not describe the contents by itemizing them in list form. Minimize aesthetics descriptions as much as possible.

Important:
- Do not generate any content for which there is no clear evidence in the image.
- Make sure to recognize and include all information that is actually present in the image.
- Avoid overgeneralizing from specific details to broad conclusions that are not explicitly shown.

Question: {question}

Table 16: Instruction for generating image caption.

Instruction

Given two separate answers obtained from different modality documents, your task is to merge them into a single coherent response.
  - For sentences that express the same claim in both answers, combine them into a single sentence and include all relevant citation numbers together.
  - Avoid repetition and redundancy.
  - Maintain factual accuracy only based on the content of both answers.
  - Keep the merged response clear, concise, and well-structured.

Answer 1: {answer_1}
Answer 2: {answer_2}

Coherent answer:

Table 17: Instruction for answer integration.

ommended prompting[6]. After the referential answer generation prompt in Table 15, we add following prompt.

```
First, think carefully step by
step about what documents are
needed to answer the query. Put
your thinking process between
<thinking> and </thinking> tags.
```

Self-refine Prompting. For the self-refine method, as shown in Table 18, we construct a self-feedback prompt to enhance image grounding ability. The response is finalized when the combined score reaches 6 or when three iterations have been completed. After generating the referential answer, the self-feedback prompt is appended directly to the model's chat history. The generated feedback and score are passed to the model in the next iteration.

C.2 Implementation Details

We collect responses using Nucleus sampling with $\mathcal{T} = 0.7$ and $p = 0.95$, by selecting the most likely sequence. We set the maximum new token length as 2048 tokens. Image resolution was rescaled such that the maximum dimension—either width or height—did not exceed 512 pixels. We utilize 8 × NVIDIA RTX A6000s to generate responses with InternVL2.5 and Qwen2.5VL.

C.3 Error Analysis

Examples of model errors are provided in Table 19.

---

[6] https://platform.openai.com/docs/guides/text?api-mode=chat

Instruction

We want to iteratively improve the provided responses. Scores for each response on desired traits are provided:

1) Evidence Existence (0 to 3): Did the response rely solely on the information and evidence present in the image documents?
   Score 0: Does not use any information from the image documents
   Score 3: Relies solely on information present in the image documents

2) Evidence Utilization (0 to 3): Did the response effectively identify and use the information in the image documents?
   Score 0: Fails to identify or use key information from the image documents
   Score 3: Accurately identifies and effectively uses key information from the image documents

3) Appropriate Generalization (0 to 3): Did the response rely solely on the information and evidence present in the image documents?
   Score 0: Includes inaccurate or unsupported generalizations beyond the image documents
   Score 3: Makes appropriate generalizations strictly based on the image documents

1. Read through the given documents and your response.
2. For each criterion, perform an evaluation.
3. Write your combined score between <total score> and </total score>

Table 18: Instruction for generating self-feedback during referential answer generation.

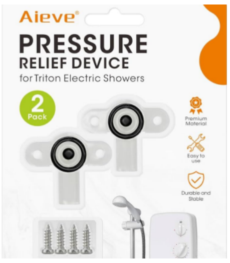
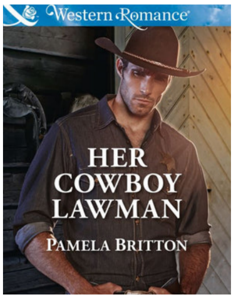
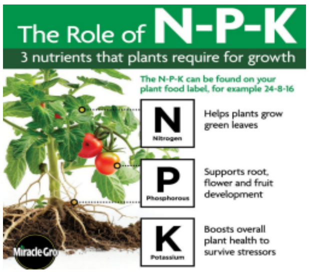
Category	Referenced Document	Model Answer (Author's Explanation)	%
(1)	 Aieve PRESSURE RELIEF DEVICE for Triton Electric Showers 2 Pack Premium Material Easy to use Durable and Stable	The Madrid Triton 2 relies on mains water pressure to operate [2]. (The image does not contain any information indicating that Madrid Triton 2 depends on water pressure.)	55.21%
(2)		...and the cover does not prominently feature the name of a well-known author [3]. (The cover shown in the image displays the well-known author's name, Pamela Britton.)	26.08%
(3)	 The Role of N-P-K 3 nutrients that plants require for growth The N-P-K can be found on your plant food label, for example 24-8-16 N Nitrogen Helps plants grow green leaves P Phosphorous Supports root, flower and fruit development K Potassium Boosts overall plant health to survive stressors MiracleGro	The nutrients N-P-K found in plant food support root, flower, and fruit development, which is beneficial for all types of plants. (There is no information indicating that the nutrients N-P-K are beneficial for all types of plants.)	12.17%

Table 19: Categories that common errors in image grounding made by GPT-4o and Qwen2.5VL. An erroneous grounding may belong to more than one category. Authors provide explanations of the error causes for clarification.