Evaluation and Enhancement of Semantic Grounding in Large Vision-Language Models

Jiaying Lu[*♠†], Jinmeng Rao[♦†], Kezhen Chen[♦], Xiaoyuan Guo[♦], Yawen Zhang[♦], Baochen Sun[♦],
Carl Yang[♠], Jie Yang[♦]

[♦]Mineral, [♠]Emory University
[♦]{jinmengrao, kezhenchen, xiaoyuanguo, yawenz, baochens, yangjie}@mineral.ai
[♠]{jiaying.lu, j.carlyang}@emory.edu

Abstract

Large Vision-Language Models (LVLMs) offer remarkable benefits for a variety of vision-language tasks. However, a challenge hindering their application in real-world scenarios, particularly regarding safety, robustness, and reliability, is their constrained *semantic grounding* ability, which pertains to connecting language to the physical-world entities or concepts referenced in images. Therefore, a crucial need arises for a comprehensive study to assess the semantic grounding ability of widely used LVLMs. Despite the significance, sufficient investigation in this direction is currently lacking. Our work bridges this gap by designing a pipeline for generating large-scale evaluation datasets covering fine-grained semantic information, such as color, number, material, *etc.*, along with a thorough assessment of seven popular LVLMs' semantic grounding ability. Results highlight prevalent *misgrounding* across various aspects and degrees. To address this issue, we propose a data-centric enhancement method that aims to improve LVLMs' semantic grounding ability through multimodal instruction tuning on fine-grained conversations. Experiments on enhanced LVLMs demonstrate notable improvements in addressing misgrounding issues.

Introduction

Large Vision-Language Models (LVLMs) (Zhao et al. 2023b; Tang et al. 2023; OpenAI 2023; Yang et al. 2023) expand the powerful large language models (Ouyang et al. 2022; Touvron et al. 2023; Ling et al. 2023) to versatile general-purpose vision-language understanding and generation interfaces. This is achieved through the integration of the vision encoder and the autoregressive large language model (Liu et al. 2023a; Li et al. 2023a; Luo et al. 2023). While demonstrating promising performance in solving various vision-language benchmarks, comprehensive examination and analysis are desired before deploying LVLMs into real-world critical-sensitive applications. Recent studies reveal that many LVLMs still suffer from text-image misalignment (Yarom et al. 2023), adversary perturbed input (Zhao et al. 2023a), and object hallucination (Li et al. 2023c) even in some seemingly simple cases. In this paper, we specifically focus on the evaluation of the under-explored semantic grounding ability in LVLMs.

Semantic grounding (*i.e.* the capability to connect words to the physical-world entities or concepts they refer to) (Yun, Sun, and Pavlick 2021; Li et al. 2022) is critical to the safe, robust, and reliable development of LVLMs. Although identifying "a Siberian tiger" as "a tabby cat" in the user-shared image under a social chatbot setting seems harmless, stakes escalate significantly when an LVLM-aided disease diagnosis assistant interprets instruction to analyze the patient's "left lung" as "right lung". In this study, we comprehensively evaluate the semantic grounding ability of existing LVLMs through our proposed *evaluation suite*. Specifically, we automatically generate $9,000$ vision-language test samples, exploring semantic grounding proficiency across four formats and addressing six types of grounding targets. Seven state-of-the-art LVLMs are evaluated using the $9,000$ test samples, and the experimental results reveal that most of them exhibit semantic grounding deficiency across various aspects and degrees. To facilitate the scalability of test sample generation and the interpretability of evaluation metrics, we propose to adopt multiple-choice questions as the test format. Further technical details are presented in the *Evaluation of Semantic Grounding* Section.

Moreover, we introduce a data-centric *enhancement method* designed to enhance the semantic grounding capabilities of LVLMs. Diverging from classical model-centric approaches (Zha et al. 2023) that emphasize advancements in model architecture, our approach focuses on curating a substantial volume of diverse multimodal, multi-round conversation data that can be leveraged by any LVLM. In total, we curate $180,000$ fine-grained instructional instances for the purpose of semantic grounding enhancement. Experimental results with LVLMs fine-tuned on our enhancement data consistently demonstrate improvements in multimodal semantic grounding.

Related Work

Trustworthiness Evaluation of LVLMs. Many works leverage existing vision-language datasets to derive trustworthiness evaluation benchmarks for LVLMs. MME (Fu et al. 2023) consists of 14 subtasks based on public images with manually constructed annotations, which measure both perception and cognition abilities of LVLMs in the

---

[*] Work was done when Jiaying Lu was an intern at Mineral.

[†] These authors contributed equally.

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

form of Yes-or-No question answering. The LAMM benchmark (Yin et al. 2023) covers nine common 2D image tasks and three common point cloud tasks with specifically curated inference instruction. Other similar benchmarks include LVLM-eHub (Xu et al. 2023), MM-Vet (Yu et al. 2023), and MMBench (Liu et al. 2023b), etc. There also exist benchmarks focusing on evaluating specific properties of LVLMs. POPE (Li et al. 2023c) focuses on evaluating object hallucination by asking Yes-or-No questions regarding the object existence of input images. M-HalDetect (Gunjal, Yin, and Bas 2023) proposes the hallucination task as sentence-level classification, using human-annotated labels. (Zhao et al. 2023a) proposes evaluating the robustness of LVLMs by adding adversarial noise into input images. Our work also provides a valuable resource to serve as a comprehensive trusworthiness benchmark for LVLMs, from a novel perspective focusing on semantic grounding. Moreover, we provide a generic framework to enable researchers to conveniently create test samples for evaluating LVLMs.

Methods for Building Safe, Robust, and Reliable LVLMs. There are two streams of approaches to improve the safeness, robustness, and responsibilities of LLMs and LVLMs (Ji et al. 2023): model-centric approaches and data-centric approaches. Model-centric approaches often focus on the model advancements, which involve (1) designing robust training paradigms (Berg et al. 2022; Dong et al. 2023), (2) robust inference (Wang et al. 2022; Zhang et al. 2023), (3) refining generated response (Madaan et al. 2023), *etc*. Data-centric approaches (Zha et al. 2023; Bai et al. 2024), on the other hand, focus on ensuring data quality and reliability, which often involve (1) faithful training data development such as data collection (Liu et al. 2023a; Lu et al. 2023) and data cleaning (Northcutt, Jiang, and Chuang 2021; Monarch and Munro 2021), (2) inference stage data augmentation such as retrieving supporting knowledge (Chen et al. 2022; Cui et al. 2023). Our framework follows the data-centric idea. Instead of modifying the model structure, our work aims to improve the semantic grounding ability of LVLMs via multimodal instruction tuning, which has proved to be a generic and efficient approach for improving LVLMs.

Preliminaries

We illustrate one representative architecture of LVLMs in Figure 1, which typically consists of the following components:

1. a visual encoder that extracts features from the input image;
2. a projector that projects visual features to the language embedding space;
3. a tokenizer that tokenizes textual input into textual tokens and maps them into language embedding space;
4. a decoder-only LLM (e.g., LLaMA) that generates textual responses based on the multimodal inputs.

The overall output generation process of a LVLM $\mathcal{F}_\Theta$ can be formally described by

$$\mathbf{Y} = \mathcal{F}_\Theta(\mathbf{X}^v, \mathbf{X}^q), \tag{1}$$

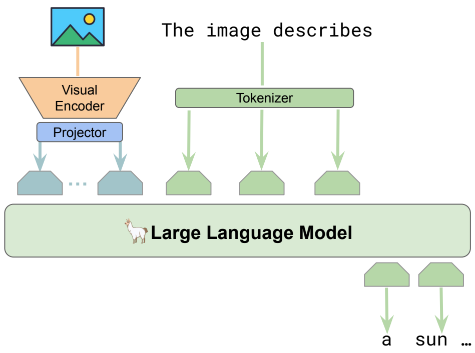


Figure 1: One representative architecture of existing LVLMs. Cyan hexagons denote visual embeddings, and green hexagons denote textual embeddings.

where Θ are parameters of the LVLM, $\mathbf{X}^v$ denotes the visual input, $\mathbf{X}^q$ denotes the textual input, and $\mathbf{Y}$ denotes the generated output sequence.

Specifically, given a visual input $\mathbf{X}^v$, a visual encoder f_ϕ firstly extracts visual features by

$$\mathbf{Z}^v = f_\phi(\mathbf{X}^v), \tag{2}$$

where $\mathbf{Z}^v \in \mathbb{R}^{l^v \times d}$, and ϕ are parameters of f_ϕ typically frozen in LVLMs training. Here l^v denotes the length of visual tokens, and d is the dimension of visual tokens. Regarding the visual encoder f_ϕ, most existing LVLMs employ ViT-based structures (Dosovitskiy et al. 2020) and select certain layers outputs to construct a certain length of visual tokens. For instance, LLaVA (Liu et al. 2023a) utilizes grid features before and after the last Transformer layer of ViT, while LaVIN (Luo et al. 2023) utilizes the $[CLS]$ embeddings from every fourth layer of ViT.

A trainable projector, denoted as g_ω, is then applied to convert $\mathbf{Z}^v$ into language embedding space, which can be defined by

$$\mathbf{H}^v = g_\omega(\mathbf{Z}^v) = g_\omega(f_\phi(\mathbf{X}^v)), \tag{3}$$

where $\mathbf{H}^v \in \mathbb{R}^{l^v \times h}$, and ω are trainable parameters of g_ω. h denotes the dimension of language embedding space. In practise, an efficient implementation of g_ω can be a projection matrix $\mathbf{W} \in \mathbb{R}^{d \times h}$ (Liu et al. 2023a), thus $\mathbf{H}^v = \mathbf{Z}^v\mathbf{W}$. More sophisticated projectors are also proposed, such as Q-former (Li et al. 2023b).

Given the input textual query $\mathbf{X}^q$, a tokenizer k_ψ is employed to tokenize and map them into "textual tokens" by:

$$\mathbf{H}^q = k_\psi(\mathbf{X}^q), \tag{4}$$

where $\mathbf{H}^q \in \mathbb{R}^{l^q \times h}$, and ψ are trainable parameters of k_ψ. It is worth noting that $\mathbf{H}^q$ and projected $\mathbf{H}^v$ have the same dimensionality $\mathbb{R}^h$ as the input of the large language model p_θ. Similarly, the length of textual tokens varies depending on the choice of tokenizer, as words are chunked into subwords. Moreover, special tokens $[s]$, $[/s]$ can be added to indicate the span of visual tokens and textual tokens. Therefore, we denote the process of LVLM to prepare the multimodal input of its LLM component as:

$$\mathbf{H} = [\mathbf{H}^v, \mathbf{H}^q] \in \mathbb{R}^{(l^q + l^v) \times h}. \tag{5}$$

Once the multimodal input features are processed, the LLM component of an LVLM is responsible for generating responses. The LLM component is essentially a probabilistic auto-regressive model p_θ with trainable parameters θ that can predict the next token y_t step by step based on the input $\mathbf{H}$ and tokens predicted so far $\mathbf{Y}_{0:t-1}$. The process can be formulated by:

$$Pr(\mathbf{Y}_{0:\tau}|\mathbf{H}) = \prod_{t=1}^{\tau} p_\theta(y_t|\mathbf{H}, \mathbf{Y}_{0:t-1}). \quad (6)$$

where Pr denotes the probability of generating the output sequence $\mathbf{Y}_{0:\tau}$.

Evaluation of Semantic Grounding

A wide range of forms can be used for the evaluation of semantic grounding in LVLMs, and each form has its own advantages and disadvantages. Free-form questions (Yarom et al. 2023) are easy to design but require resource-intensive human evaluations and are difficult to score consistently. Similarity-based evaluation uses less resources, but is heavily reliant on high-quality ground truth responses and the bias of similarity metrics. Yes-or-No questions (Fu et al. 2023) are non-ambiguous and easier to evaluate. However, they may be too easy and cannot capture all aspects of semantic grounding in LVLMs.

We propose **MSG-MCQ** for the novel **M**ultimodal **S**emantic **G**rounding evaluation based on **M**ultiple-**C**hoice **Q**uestions (MCQs) (Lu et al. 2022a,b). This form presents a question along with a set of predetermined choices, allowing respondents to select the option they believe to be correct. The MCQs facilitate efficient grading and analysis of responses, and the difficulty level can be controlled by adjusting the number and the feasibility of distractor choices (Lu et al. 2022a,b). Moreover, the Yes-or-No form can be regarded as a special case in MCQs where the choices are "(A)Yes. (B) No.". We use *accuracy* as the evaluation metric for MCQs. Following the notation in the *Preliminaries* section, we extend the textual input question $\mathbf{X}^q$ into two parts: the question body $\mathbf{X}^{qb}$ and the multiple choices $\mathbf{X}^{qc}$. Therefore, we formally define the task of MSG-MCQ:

Definition 1 (MSG-MCQ) *Given an input image* $\mathbf{X}^v$*, input question body* $\mathbf{X}^{qb}$ *, and* K *input choices* $\mathbf{X}^{qc} = \{\mathbf{C}^1, \mathbf{C}^2, \ldots, \mathbf{C}^K\}$*,* MSG-MCQ *expects a LVLM* $\mathcal{F}_\Theta$ *to select the correct choice* $\mathbf{C}^i$ *where* $1 \leq i \leq K$ *from the input choices* $\mathbf{X}^{qc}$*.*

In the scope of LVLM, the selection action is typically determined by the generated sequence of LVLM. Depending on the specific implementation of LVLMs, the output can be as simple as a choice indicator like "C", or a comprehensive response that includes both the selected choice and an explanation of selecting it such as "The correct choice is A, because . . . ".

MSG-MCQ Generation Pipeline

In this work, we propose an efficient automated MCQ generation method to derive our MSG-MCQ. Figure 2 provides several automatically generated evaluation examples. Specifically, we pose four specific kinds of MCQs: *Yes-or-No, Fill-in-the-Blank, What, Correction* in different columns in Figure 2. Orthogonal to the specific kinds of MCQs, the evaluation focuses on different targets of semantic grounding that indicate the particular deficiency of evaluated LVLMs. In this work, we include six targets of semantic grounding: *Entity, Number, Color, Material, Action, and Spatial*. These target categories are shown in different rows in Figure 2. It is worth noting that the evaluation module can be easily extended to more types and targets in the future.

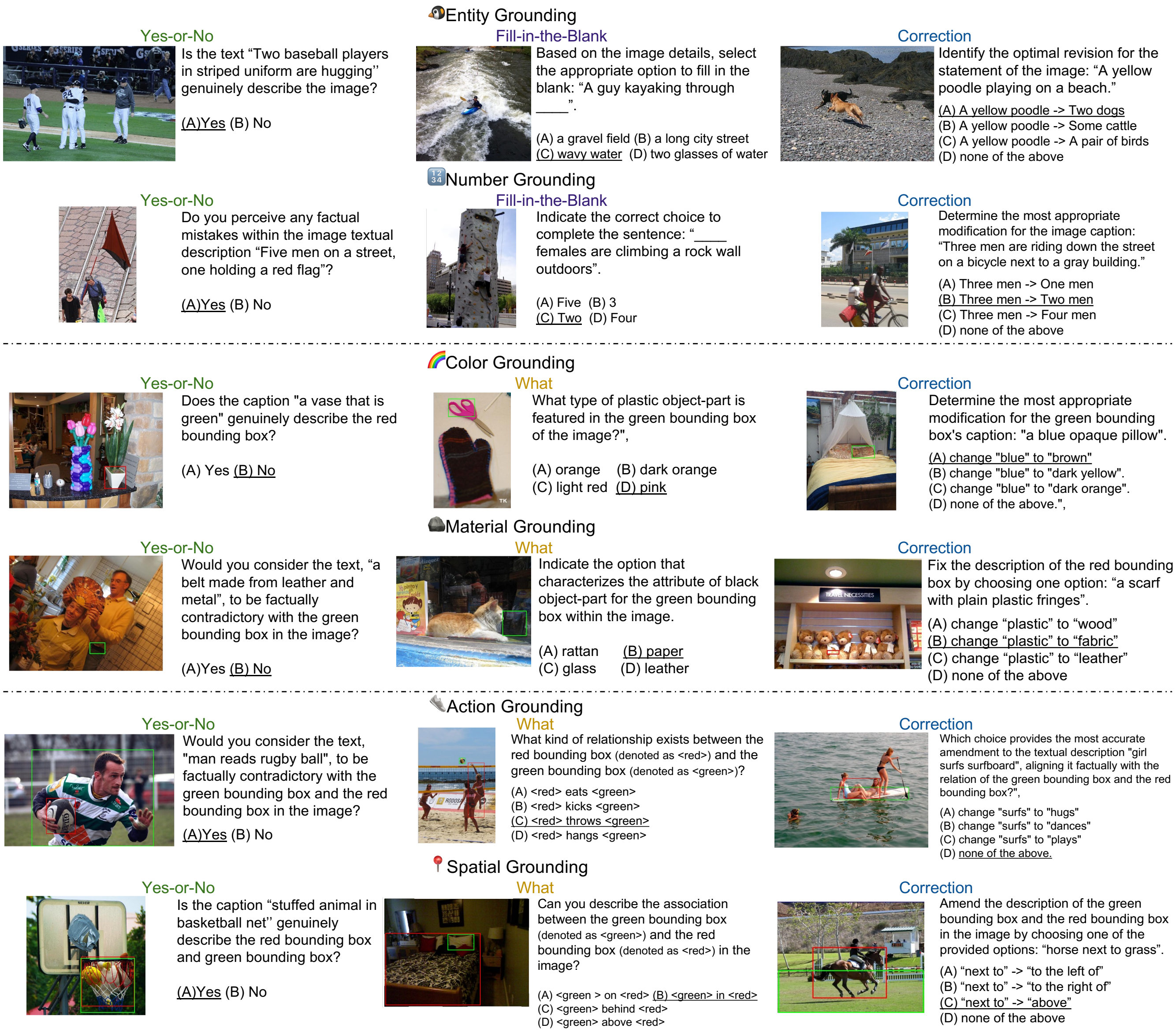
All these MCQs are generated from ground-truth source data through a four-step generation pipeline. In **step 1**, we curate a pool of question templates with placeholders for each kind of MCQ focusing on one specific target of semantic grounding, according to the source data we have. In **step 2**, we randomly sample one question template from the pool. As an illustration, a question template of *What* question focusing on color grounding is:

Example 1 (What question) *What color of [obj-attr] object is featured in [bbox-color] bounding box of the image?*
(A) [distractor#1] *(B) [ground-truth]*
(C) [distractor#2] *(D) [distractor#3].*

As can be seen, the sampled question template contains several placeholders, and the ground-truth is randomly placed into the choice (A), (B), (C), or (D). In **step 3**, we use source data to fill in these placeholders. For example, "*[obj-attr], [ground-truth]*" can be filled using the gold annotation in the source data. Regarding the "*[bbox-color]*", we first use gold bounding box coordinates referring to the querying object to draw a box in the original image using a random color (green or red), then replace "*[bbox-color]*" with the name of that color. In **step 4**, we generate the distractors of ground truth based on the multimodal input information. This is one of the most critical steps in the evaluation pipeline, since distractors determine the difficulty level of MCQs. Distractors should not be semantically equivalent to ground truth, but they should be plausible enough to serve as an answer candidate to the question. Various distractor generation methods (Lu et al. 2022a) can be used, such as manual generation, thesaurus-based generation, and end-to-end generation. In this work, we utilize thesaurus-based generation with post-human verification. According to the type of ground truth (entity, number, color, *etc.*), we randomly sample 15 distractors. For entity grounding, we use the sentence transformer model (Reimers and Gurevych 2019) to filter out candidates that have overly high similarity scores to the gold answer. For other targets of grounding, the thesaurus is guaranteed to contain semantically different antonyms of the gold answer.

We introduce the necessary tweak of the overall pipeline for each specific kind of MCQ, and explain the purpose of why we need them as below:

- **Yes-or-No:** *Can a model identify whether a textual description is appropriate for a given image?* For each Yes-or-No MCQ, two choices are given, as shown in the first column of Figure 2. The correct textual descriptions are either directly taken from the source data (image captioning datasets), or obtained using some sentence templates



Entity Grounding

Yes-or-No

Is the text "Two baseball players in striped uniform are hugging" genuinely describe the image?

(A)Yes (B) No

Fill-in-the-Blank

Based on the image details, select the appropriate option to fill in the blank: "A guy kayaking through ____".

(A) a gravel field (B) a long city street
(C) wavy water (D) two glasses of water

Correction

Identify the optimal revision for the statement of the image: "A yellow poodle playing on a beach."

(A) A yellow poodle -> Two dogs
(B) A yellow poodle -> Some cattle
(C) A yellow poodle -> A pair of birds
(D) none of the above

Number Grounding

Yes-or-No

Do you perceive any factual mistakes within the image textual description "Five men on a street, one holding a red flag"?

(A)Yes (B) No

Fill-in-the-Blank

Indicate the correct choice to complete the sentence: "____ females are climbing a rock wall outdoors".

(A) Five (B) 3
(C) Two (D) Four

Correction

Determine the most appropriate modification for the image caption: "Three men are riding down the street on a bicycle next to a gray building."

(A) Three men -> One men
(B) Three men -> Two men
(C) Three men -> Four men
(D) none of the above

Color Grounding

Yes-or-No

Does the caption "a vase that is green" genuinely describe the red bounding box?

(A) Yes (B) No

What

What type of plastic object-part is featured in the green bounding box of the image?",

(A) orange (B) dark orange
(C) light red (D) pink

Correction

Determine the most appropriate modification for the green bounding box's caption: "a blue opaque pillow".

(A) change "blue" to "brown"
(B) change "blue" to "dark yellow".
(C) change "blue" to "dark orange".
(D) none of the above.",

Material Grounding

Yes-or-No

Would you consider the text, "a belt made from leather and metal", to be factually contradictory with the green bounding box in the image?

(A)Yes (B) No

What

Indicate the option that characterizes the attribute of black object-part for the green bounding box within the image.

(A) rattan (B) paper
(C) glass (D) leather

Correction

Fix the description of the red bounding box by choosing one option: "a scarf with plain plastic fringes".

(A) change "plastic" to "wood"
(B) change "plastic" to "fabric"
(C) change "plastic" to "leather"
(D) none of the above

Action Grounding

Yes-or-No

Would you consider the text, "man reads rugby ball", to be factually contradictory with the green bounding box and the red bounding box in the image?

(A)Yes (B) No

What

What kind of relationship exists between the red bounding box (denoted as <red>) and the green bounding box (denoted as <green>)?

(A) <red> eats <green>
(B) <red> kicks <green>
(C) <red> throws <green>
(D) <red> hangs <green>

Correction

Which choice provides the most accurate amendment to the textual description "girl surfs surfboard", aligning it factually with the relation of the green bounding box and the red bounding box?",

(A) change "surfs" to "hugs"
(B) change "surfs" to "dances"
(C) change "surfs" to "plays"
(D) none of the above.

Spatial Grounding

Yes-or-No

Is the caption "stuffed animal in basketball net" genuinely describe the red bounding box and green bounding box?

(A)Yes (B) No

What

Can you describe the association between the green bounding box (denoted as <green>) and the red bounding box (denoted as <red>) in the image?

(A) <green > on <red> (B) <green> in <red>
(C) <green> behind <red>
(D) <green> above <red>

Correction

Amend the description of the green bounding box and the red bounding box in the image by choosing one of the provided options: "horse next to grass".

(A) "next to" -> "to the left of"
(B) "next to" -> "to the right of"
(C) "next to" -> "above"
(D) none of the above

Figure 2: Instances of our MSG-MCQ, which covers four types of MCQs and six types of grounding targets.

(object detection). The wrong descriptions are generated by negative replacement on correct descriptions, where we change specific textual spans for corresponding semantic grounding target (*e.g.* sampling "yellow" to replace the original "blue" in color grounding).

- **Fill-in-the-Blank.** *Can a model infer the missing pieces of information regarding multimodal input?* For each Fill-in-the-Blank MCQ, four choices are given where only one choice is correct. The second column of the first two rows of Figure 2 gives real examples of such type MCQs. A text span that contains the concept of interest (*e.g.* entity, number) is blanked, and the span is used as one choice. Three distracting choices are generated using negative sampling.

- **What.** *Can a model recognize and identify an object or attribute that is specified in the multimodal input?* For each What MCQ, four choices are given where only one choice is correct. The second column of the last four rows of Figure 2 gives real examples of such type MCQs. Similarly, three distracting choices are generated using negative sampling.

- **Correction** *Can a model identify the inconsistency across different modalities and propose an appropriate correction?* For each Correction MCQ, four choices are given where only one choice is correct. The last column of Figure 2 provides real examples of such type MCQs. The choice "(D) none of the above" is always included in each Correction MCQ, indicating the scenario where the original description is grounded. Correction MCQs are challenging, as they require models to imagine whether the corrected text is consistent with the image.

Enhancement of Semantic Grounding

Our solution for enhancing the semantic grounding of LVLMs is a data-centric approach (Zha et al. 2023) to systematically and algorithmically generate the instruction-tuning dataset to feed LVLMs. The emerging success of transformer model architectures in both large language models (Ouyang et al. 2022; OpenAI 2023; Touvron et al. 2023) and large vision-language models (Zhu et al. 2023; Awadalla et al. 2023) advocates a fundamental shift from the model-centric AI to data-centric AI. Instead of focusing on designing specific model architectures for a particular downstream task, we curate a substantial (180K instances) of new multimodal multi-round conversational instruction data for LVLMs to be further improved upon.

Instruction Tuning Data Generation

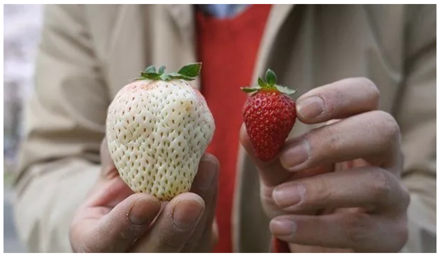
Image	
User	What colors are the strawberries the man is holding?
LVLM	The man is holding one white strawberry and one red strawberry.
User	Do you think the white strawberry is to the right of the red strawberry?
LVLM	No, the white strawberry appears to be on the left of the red strawberry in the image.

Table 1: Toy Example of Multimodal Multi-Round Instruction-Tuning Data

For the instruction data generation, we first curate a set of fine-grained conversations of multimodal inputs that help LVLMs interpret the information in the image step by step and eventually answer the question. Table 1 provides a toy example of such instruction data. All the conversations are provided as multi-step chain-of-thought (Zhang et al. 2023) instructions to simulate the human reasoning process and guide LVLMs to pay attention to the right information at each step. Specifically, we design three types of instructions:

- **Multi-Round Conversation**. Given an image, we provide a multi-round conversation between users and models. The user asks a few questions about some fine-grained content in the image and the LVLM gives factual and precise answers based on the observation.
- **Vision-Prompted Recognition**. Given an image, we draw visual prompts (e.g., bounding boxes) on the image and ask an LVLM to tell the names, attributes, or relations of the objects indicated by visual prompts. The LVLM needs to learn how to follow the visual prompts to first localize the indicated objects and then recognize detailed attributes or relations of them.
- **Fact Checking**. Given an image and a statement of some facts in the image, we ask the LVLM if and why a given statement is factually consistent. In the instructions, we provide reference answers to guide the LVLM to localize main objects, recognize their attributes and relations, and determine if there are any factual misalignments. Eventually, the LVLM determines whether it is factual.

All three types of instructions essentially consist of two fundamental components: the question and the response. We manually curate diverse templates for both questions and responses. In a similar fashion, the response templates include placeholders to be filled with ground-truth data. Once these placeholders are populated, we employ chatGPT to rephrase these instructions, enhancing their diversity.

Instruction Tuning

After the fine-grained conversations are generated, we conduct instruction tuning (Liu et al. 2023a; Ouyang et al. 2022) on LVLMs to enhance semantic grounding. Following the notation we introduce in the *Preliminary section*, the inference process of LVLM can be described as:

$$\hat{\mathbf{Y}} = \mathcal{F}_{\Theta}(\mathbf{X}^v, \mathbf{X}^q), \tag{7}$$

where $\mathcal{F}_{\Theta}$ denotes the LVLM with trainable parameters Θ, $\mathbf{X}^v$ denotes the visual inputs, $\mathbf{X}^q$ denotes the textual inputs, and $\hat{\mathbf{Y}}$ denotes the textual response. It is worth noting that the trainable parameters Θ of $\mathcal{F}$ actually contains $\Theta = \{\phi, \omega, \psi, \theta\}$. We denote the n-th training sample $(\mathbf{X}^v_{(n)}, \mathbf{X}^q_{(n)}, \mathbf{Y}^r_{(n)})$ from our generated instruction tuning data, where they represent the visual input, textual query input, and the ground truth response, respectively. The loss function of LVLM is

$$\mathcal{L}(\Theta) = -\sum_{n=1}^{N} log\, Pr(\mathbf{Y}^r_{(n)} | \mathbf{X}^v_{(n)}, \mathbf{X}^q_{(n)}), \tag{8}$$

where Pr denotes the probability of generating the output sequence $\mathbf{Y}^r_{(n)}$ from input $(\mathbf{X}^v_{(n)}, \mathbf{X}^q_{(n)})$ using equation (6).

Experiments and Analysis

Evaluation Examples

Table 2 gives an overview of key statistics of test multiple-choice questions generated by MSG-MCQ. We collect six subsets for each semantic grounding type, together with 9K MCQs, 8.1K images, and 1.7K unique answers. These MCQs are built from four public datasets: Flickr30K (Young et al. 2014), PACO (Ramanathan et al. 2023), OpenImage-V7 (Krasin et al. 2017), SpatialSense (Yang, Russakovsky, and Deng 2019). Flickr30K is an image captioning dataset that includes images obtained from Flickr and each image is provided with five manually annotated captions.

Grounding Target	Q Type	#IMGs	#Qs	#As	Data Source
Entity	YoN, FiB, Corr	1,339	1,500	628	Flickr30K
Number	YoN, FiB, Corr	977	1,500	389	Flickr30K
Color	YoN, What, Corr	1,500	1,500	153	PACO
Material	YoN, What, Corr	1,500	1,500	68	PACO
Action	YoN, What, Corr	1,498	1,500	322	OpenImage-V7
Spatial	YoN, What, Corr	1,500	1,500	144	SpatialSense
Overall	YoN, FiB, What, Corr	8,119	9,000	1,683	

Table 2: Overview of MCQs generated by MSG-MCQ. #IMGs denotes unique images, #Qs denotes unique questions, and #A denotes unique answers.

PACO, OpenImage-V7 and SpatialSense are object detection datasets that contain fine-grained object/object-part bounding boxes, categories, and attribute annotations. Furthermore, OpenImage-V7 and SpatialSense provide relational annotations between two objects within one image. We curate these testing MCQs by our proposed MSG-MCQ. Also, we balance the answer distribution and the concept of semantic grounding target for promising evaluation.

Evaluated Models

We select seven SOTA LVLMs for evaluation and all use the 7B parameters variants for a fair comparison:

- mPLUG-Owl (Ye et al. 2023), MiniGPT4 (Zhu et al. 2023), LLaVA (Liu et al. 2023a), InstructBLIP (Dai et al. 2023): LVLMs based on the visual encoder and the pre-trained LLM that are similar to the architecture we introduce in the *Preliminary Section*. Both follow a two-stage training, where stage-1 is a pre-training stage to align visual and textual concepts when LLM is frozen, and stage-2 is a fine-tuning stage to feed into instruction-following data to train both visual encoder and LLM simultaneously.
- Otter (Li et al. 2023a), LLaMA-AdapterV2 (Gao et al. 2023), LaVIN (Luo et al. 2023): LVLMs with a slightly different architecture, where visual features after adaptation/fusion are attended with text embedding in each/some layers of the LLM component.

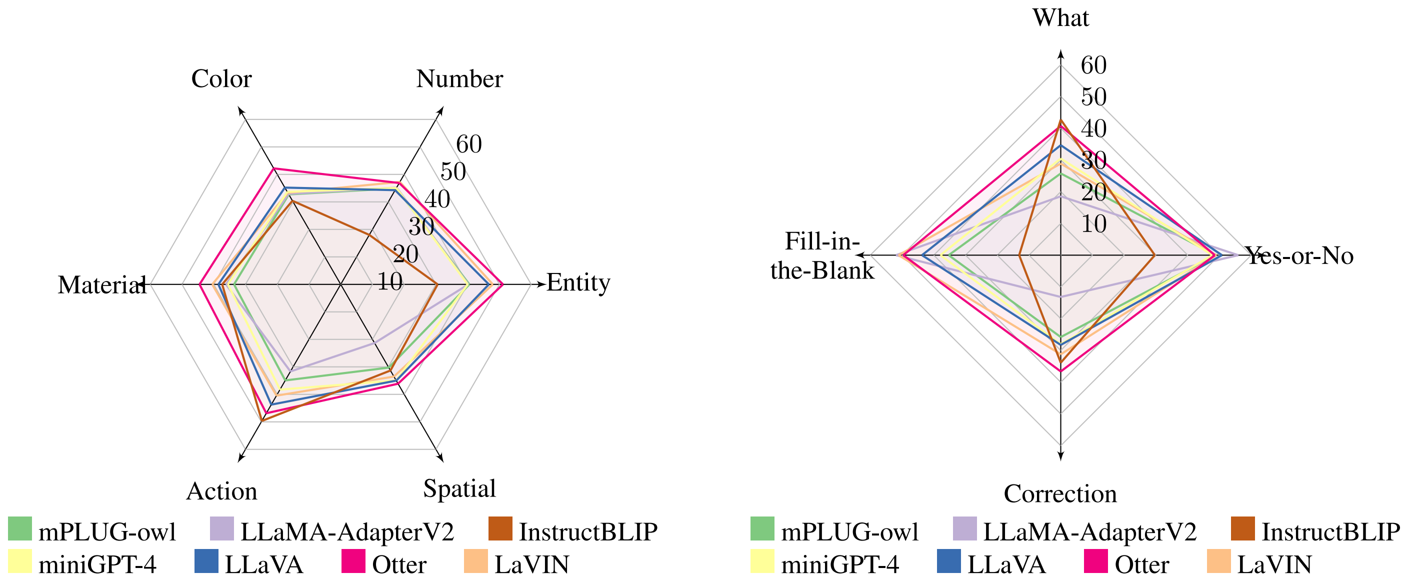
We also test one large language model (language-only): LLaMA-2-chat (Touvron et al. 2023) to verify the necessity of multimodal perception ability for answering the generated MCQs. To extract the choice indices (A, B, C, or D) from the free-form responses of LVLMs, we use regular expressions if they explicitly contain indices. Otherwise, we use ChatGPT to help extract indices.

Zero-Shot Setting

Figure 3 provides the radar charts of seven advanced LVLMs on each semantic grounding target or each question type. Among these LVLMs, Otter, LLaVA, and LaVIN are top competitors that consistently deliver better performance than other LVLMs. In terms of semantic grounding targets, LVLMs are more effective at perceiving and understanding Entity and Action, but extremely struggle in Spatial relations. We speculate the different performances on different kinds of semantic grounding come from the bias of training data used by LVLMs. mPLUG-owl, LLaMA-AdapterV2, MiniGPT-4, and InstructBLIP all mainly trained on coarse-grained text-image pairs corpus. On the other hand, Otter has been trained on their curated MIMIC-IT corpus that covers perception, reasoning, and planning-oriented text-image QA pairs. LLaVA has been trained on not only the captions, but also bounding boxes with detailed descriptions. The model checkpoint of LaVIN we used here is trained on ScienceQA (Lu et al. 2022b) corpus, which is in the format of MCQs. These additional training corpora provide more fine-grained information on the multimodal input, thus helping these LVLMs achieve better performance.

Table 3 provides a detailed view of the zero-shot accuracy including human, random-guess, and LLaMA2 baselines. The accuracy of random-guess in each semantic grounding type is always 33.33, since each semantic grounding subset contains 500 two-way (50.00 accuracy by random guess) and 1000 four-way MCQs (25.00 accuracy by random guess). Human performance (five annotators carefully answer random 500 MCQs from MSG-MCQ) sets up the upper bound of the zero-shot experiments, which significantly outperforms all other LVLMs. Interestingly, humans do not perform well in color, material, and spatial. One possible reason for human deficiency in color and material is that human perception organs are a bit weaker at distinguishing attributes, as compared to recognizing objects and actions. For Spatial MCQs, we observe that the source data quality is not high. For example, sometimes the spatial relations are labeled according to the physical position, while sometimes they are labeled according to what the annotator perceived. Comparing the language only LLaMA2-chat and other LVLMs, the best LVLMs are consistently better in every semantic grounding type. This indicates the importance of the perception ability of input vision modality. Besides, LVLM models struggle more with correction questions, which is also more challenging for humans.

Moreover, Table 4 supplies LVLMs performance on different Q-Types with human, random-guess, and LLaMA2 baselines. As can be seen, Correction type MCQs are most challenging for both humans and LVLMs. Given the accuracy of the Random-Guess baseline as 25.00 (four-way MCQs with one correct answer), the best LVLM (Otter) only outperforms it by 11.63. While for other four-way What and Fill-in-the-Blank MCQs, the best LVLM achieves 17.70



(a) Zero shot accuracy on each semantic grounding target.

(b) Zero shot accuracy on each question type.

Figure 3: Comparison of seven advanced LVLMs on MSG-MCQ generated MCQs in accuracy (maximum 100).

	Entity	Number	Color	Material	Action	Spatial	Overall
Human	94.74	83.95	75.95	79.55	91.14	70.10	81.00
	zero-shot						
Random-Guess	33.33	33.33	33.33	33.33	33.33	33.33	33.33
LLaMA2-chat	39.20	35.13	34.60	39.60	38.07	34.67	36.88
mPLUG-owl	40.53	34.93	32.73	33.60	34.93	30.27	34.50
LLaMA-AdapterV2	40.20	35.20	32.93	36.13	31.67	21.33	32.91
LaVIN	47.93	**37.27**	33.33	40.47	40.46	33.53	38.83
MiniGPT-4	40.00	35.07	33.67	35.53	38.27	34.93	36.24
LLaVA	46.67	34.27	35.20	38.53	43.73	35.00	38.90
Otter	**51.13**	36.87	**42.20**	**44.47**	46.87	**36.13**	**42.94**
InstructBLIP	30.53	18.00	30.40	37.33	**49.67**	31.33	29.63

Table 3: Accuracy on MSG-MCQ different targets of semantic grounding.

	YoN	What	FiB	Corr.	Overall
Human	78.63	85.15	91.11	76.42	81.00
	zero-shot				
Random-Guess	50.00	25.00	25.00	25.00	33.00
LLaMA2-chat	50.20	28.55	37.20	9.00	36.88
mPLUG-owl	48.77	25.75	35.30	25.80	34.50
LLaMA-AdapterV2	**55.90**	18.55	**51.90**	13.17	32.91
LaVIN	49.00	28.85	51.10	31.23	38.83
MiniGPT-4	47.77	30.45	37.90	28.03	36.24
LLaVA	50.77	34.65	43.70	28.27	38.90
Otter	48.53	40.70	49.60	**36.63**	**42.94**
InstructBLIP	29.63	**42.70**	13.10	33.87	29.63

Table 4: Experimental results (Accuracy) on MSG-MCQ different types of questions (Q-Type).

and 26.10 accuracy gains, separately. On the other hand, Yes-or-No type MCQs are quite challenging too. Given the Random-Guess accuracy as 50.00, the best LVLM (LLaMA-Adapter) only achieves 5.9 higher accuracy. The patterns of experimental outcomes indicate that LVLMs are better at responding to descriptive queries (What and Fill-in-the-Blank MCQs). One speculation for that is these LVLMs have been trained on a great amount of image captioning data and image description instruction data. With the pretraining on such training corpora, LVLMs may be relatively less prepared for judgemental and corrective queries.

We further conduct an in-depth analysis using critical difference (CD) analysis (Demšar 2006). Figure 4 shows the CD diagram, which is a powerful tool to compare outcomes of multiple compared models over multiple observations. The CD analysis involves several hypothesis tests, and models connected with each other in the diagram means that the performances of these models are not that different in the sense of statistical significance. As shown in Figure 4, the performances of LVLMs are relatively similar, although they can be divided into two groups. Noticeably, there still ex-

	Entity	Number	Color	Material	Action	Spatial	Overall
LLaMA-AdapterV2+	53.00 *(↑<u>12.80</u>)*	47.80 *(↑<u>12.60</u>)*	44.40 *(↑<u>11.47</u>)*	48.93 *(↑12.80)*	46.20 *(↑14.53)*	<u>43.53</u> ***(↑22.20)***	47.31 *(↑<u>14.40</u>)*
mPLUG-owl+	51.61 *(↑11.08)*	<u>48.81</u> ***(↑13.48)***	39.92 *(↑7.19)*	43.10 *(↑9.50)*	44.63 *(↑9.70)*	39.70 *(↑9.43)*	43.66 *(↑9.16)*
LaVIN+	47.20 *(↑9.27)*	**49.53** *(↑12.26)*	**66.80** ***(↑33.47)***	**73.33** ***(↑32.86)***	<u>69.99</u> ***(↑29.54)***	**49.80** *(↑<u>16.27</u>)*	**61.11** ***(↑22.28)***
LLaVA+	<u>60.27</u> ***(↑13.60)***	45.20 *(↑10.93)*	41.27 *(↑6.07)*	60.00 *(↑<u>21.47</u>)*	**72.93** *(↑<u>29.20</u>)*	37.47 *(↑2.47)*	<u>52.86</u> *(↑13.96)*
Otter+	**62.73** *(↑11.60)*	46.20 *(↑9.33)*	<u>48.60</u> *(↑6.40)*	<u>61.60</u> *(↑17.13)*	57.53 *(↑10.66)*	40.47 *(↑4.34)*	52.85 *(↑9.91)*

Table 5: Accuracy (numbers in regular font) and accuracy gain (numbers in italic font in parentheses) on MSG-MCQ different grounding targets with enhanced LVLMs (denoted as *[method+]*).

	YoN	What	FiB	Corr.	Overall
LLaMA-AdapterV2+	51.63 *(↓4.27)*	33.75 *(↑<u>15.20</u>)*	47.20 *(↓4.70)*	52.07 ***(↑38.90)***	47.31 *(↑<u>14.40</u>)*
mPLUG-owl+	56.04 *(↑7.27)*	37.93 *(↑12.18)*	53.00 ***(↑17.70)***	34.72 *(↑8.92)*	43.66 *(↑9.16)*
LaVIN+	**74.00** ***(↑25.00)***	**52.20** ***(↑23.35)***	35.90 *(↓15.20)*	**62.56** *(↑<u>31.33</u>)*	**61.11** ***(↑22.28)***
LLaVA+	<u>61.04</u> *(↑<u>10.27</u>)*	<u>49.65</u> *(↑15.00)*	**58.30** *(↑<u>15.60</u>)*	44.67 *(↑16.40)*	<u>52.86</u> *(↑13.96)*
Otter+	53.70 *(↑5.17)*	41.10 *(↑0.40)*	<u>57.90</u> *(↑8.30)*	<u>57.06</u> *(↑21.53)*	52.85 *(↑9.91)*

Table 6: Accuracy (numbers in regular font) and accuracy gain (numbers in italic font in parentheses) on MSG-MCQ different question types with enhanced LVLMs (denoted as *[method+]*).

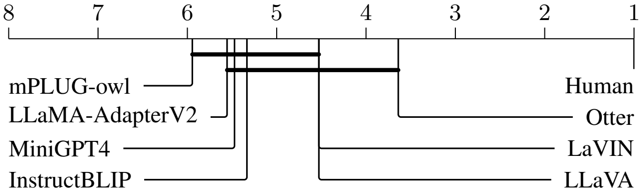


Figure 4: The critical difference diagrams of each LVLM for each semantic grounding target with each question type. The lower rank (further to the right) represents the better performance. LVLMs connected by thick bars indicate that these models are not significantly different ($p < 0.05$).

ists a significant gap between humans and all LVLMs. Overall, MSG-MCQ serves as a good evaluation module for researchers to understand the limitations of LVLMs with regard to specific fine-grained semantic grounding.

Enhanced LVLMs Setting

In order to enhance semantic grounding in LVLMs, we conduct instruction tuning[1] on LVLMs by generating 180K fine-grained multimodal instruction data covering multi-round conversations, vision-prompted recognition, and fact-checking. Table 5 shows the performance gains on several enhanced LVLMs by instruction tuning. Following the same format, we use **bold font** to highlight the best accuracy or accuracy gain, and <u>underline font</u> to highlight the second-best accuracy or accuracy gain. For those LVLMs not included, they typically do not release well-established instruction tuning scripts (or scripts are specifically for image captioning tasks). As can be seen, we observe consistent improvements over all tuned LVLMs, which indicates the effectiveness of our enhancement method. Interestingly, the performance gain for LAVIN is significantly higher than for other models. We believe this is primarily due to LAVIN incorporating trainable adaptors into both the vision encoder and the LLM, enabling end-to-end optimization of the entire model. Moreover, we supply Table 6 that offers a detailed breakdown of accuracy improvements based on MCQ types. A closer investigation of the table reveals that while performance enhancements are evident across most MCQs, there are instances of decreased accuracy in specific MCQ types. For instance, LaVIN exhibits a notable decrease of ↓ 15.20 accuracy in Fill-in-the-Blank MCQs, while LLaMA-AdapterV2 records slightly lower accuracies in Yes-or-No and Fill-in-the-Blank MCQs. Despite these isolated variations, consistent performance improvements are observed in other categories. In summary, LVLMs enhanced with our automatically generated instruction data deliver better semantic grounding performance. The instruction data is fundamentally different from the training data, since instruction data does not share the same data distribution and formats of the MSG-MCQ testing data. The instruction data also serves as a valuable resource for instruction tuning, readily available for any LVLMs to utilize.

Conclusion and Future Work

In this work, we evaluate and enhance the ability of LVLMs to ground fine-grained vision-language inputs. We propose an evaluation method MSG-MCQ to automatically generate testing samples focusing on specific grounding targets in various question formats, along with comprehensive experiments to understand how well the current state-of-the-art LVLMs perform in semantic grounding. We further propose a data-centric approach to enhance LVLMs. Its effectiveness is validated by observing consistent performance improvement of these LVLMs, after instruction tuning on multimodal multil-round conversations generated by the enhancement method. In the future, we aim to extend the scope of semantic grounding in LVLMs into (1) more modalities such as audio, time-series, and tabular; (2) more semantic grounding targets such as social-emotion, terrains, and human organs.

---

[1] Instruction tuning details are elaborated in the Appendix.

References

Awadalla, A.; Gao, I.; Gardner, J.; Hessel, J.; Hanafy, Y.; Zhu, W.; Marathe, K.; Bitton, Y.; Gadre, S.; Sagawa, S.; Jitsev, J.; Kornblith, S.; Koh, P. W.; Ilharco, G.; Wortsman, M.; and Schmidt, L. 2023. OpenFlamingo: An Open-Source Framework for Training Large Autoregressive Vision-Language Models. *arXiv preprint arXiv:2308.01390*.

Bai, G.; Chai, Z.; Ling, C.; Wang, S.; Lu, J.; Zhang, N.; Shi, T.; Yu, Z.; Zhu, M.; Zhang, Y.; Yang, C.; Cheng, Y.; and Zhao, L. 2024. Beyond Efficiency: A Systematic Survey of Resource-Efficient Large Language Models. *arXiv preprint arXiv:2401.00625*.

Berg, H.; Hall, S.; Bhalgat, Y.; Kirk, H.; Shtedritski, A.; and Bain, M. 2022. A Prompt Array Keeps the Bias Away: Debiasing Vision-Language Models with Adversarial Learning. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, 806–822.

Chen, W.; Hu, H.; Chen, X.; Verga, P.; and Cohen, W. 2022. MuRAG: Multimodal Retrieval-Augmented Generator for Open Question Answering over Images and Text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 5558–5570.

Cui, H.; Lu, J.; Wang, S.; Xu, R.; Ma, W.; Yu, S.; Yu, Y.; Kan, X.; Fu, T.; Ling, C.; et al. 2023. A Survey on Knowledge Graphs for Healthcare: Resources, Application Progress, and Promise. In *ICML 3rd Workshop on Interpretable Machine Learning in Healthcare (IMLH)*.

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. *arXiv preprint arXiv:2305.06500*.

Demšar, J. 2006. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine learning research*, 7(1): 1–30.

Dong, X.; Zhu, Z.; Wang, Z.; Teleki, M.; and Caverlee, J. 2023. Co2PT: Mitigating Bias in Pre-trained Language Models through Counterfactual Contrastive Prompt Tuning. *Findings-EMNLP*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

Fu, C.; Chen, P.; Shen, Y.; Qin, Y.; Zhang, M.; Lin, X.; Qiu, Z.; Lin, W.; Yang, J.; Zheng, X.; et al. 2023. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models. *arXiv preprint arXiv:2306.13394*.

Gao, P.; Han, J.; Zhang, R.; Lin, Z.; Geng, S.; Zhou, A.; Zhang, W.; Lu, P.; He, C.; Yue, X.; et al. 2023. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*.

Gunjal, A.; Yin, J.; and Bas, E. 2023. Detecting and Preventing Hallucinations in Large Vision Language Models. *arXiv preprint arXiv:2308.06394*.

Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y. J.; Madotto, A.; and Fung, P. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12): 1–38.

Krasin, I.; Duerig, T.; Alldrin, N.; Ferrari, V.; Abu-El-Haija, S.; Kuznetsova, A.; Rom, H.; Uijlings, J.; Popov, S.; Kamali, S.; Malloci, M.; Pont-Tuset, J.; Veit, A.; Belongie, S.; Gomes, V.; Gupta, A.; Sun, C.; Chechik, G.; Cai, D.; Feng, Z.; Narayanan, D.; and Murphy, K. 2017. OpenImages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://storage.googleapis.com/openimages/web/index.html*.

Li, B.; Zhang, Y.; Chen, L.; Wang, J.; Yang, J.; and Liu, Z. 2023a. Otter: A Multi-Modal Model with In-Context Instruction Tuning. *arXiv preprint arXiv:2305.03726*.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023b. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *ICML*.

Li, L. H.; Zhang, P.; Zhang, H.; Yang, J.; Li, C.; Zhong, Y.; Wang, L.; Yuan, L.; Zhang, L.; Hwang, J.-N.; et al. 2022. Grounded language-image pre-training. In *CVPR*.

Li, Y.; Du, Y.; Zhou, K.; Wang, J.; Zhao, W. X.; and Wen, J.-R. 2023c. Evaluating Object Hallucination in Large Vision-Language Models. *arXiv preprint arXiv:2305.10355*.

Ling, C.; Zhao, X.; Lu, J.; Deng, C.; Zheng, C.; Wang, J.; Chowdhury, T.; Li, Y.; Cui, H.; Zhao, T.; et al. 2023. Beyond One-Model-Fits-All: A Survey of Domain Specialization for Large Language Models. *arXiv preprint arXiv:2305.18703*.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023a. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Liu, Y.; Duan, H.; Zhang, Y.; Li, B.; Zhang, S.; Zhao, W.; Yuan, Y.; Wang, J.; He, C.; Liu, Z.; et al. 2023b. MMBench: Is Your Multi-modal Model an All-around Player? *arXiv preprint arXiv:2307.06281*.

Lu, J.; Qian, Y.; Zhao, S.; Xi, Y.; and Yang, C. 2023. MuG: A Multimodal Classification Benchmark on Game Data with Tabular, Textual, and Visual Fields. In *Findings-EMNLP*.

Lu, J.; Ye, X.; Ren, Y.; and Yang, Y. 2022a. Good, better, best: Textual distractors generation for multiple-choice visual question answering via reinforcement learning. In *CVPR 2022 Workshop on Open-Domain Retrieval Under a Multi-Modal Setting*, 4921–4930.

Lu, P.; Mishra, S.; Xia, T.; Qiu, L.; Chang, K.-W.; Zhu, S.-C.; Tafjord, O.; Clark, P.; and Kalyan, A. 2022b. Learn to Explain: Multimodal Reasoning via Thought Chains for Science Question Answering. In *The 36th Conference on Neural Information Processing Systems (NeurIPS)*.

Luo, G.; Zhou, Y.; Ren, T.; Chen, S.; Sun, X.; and Ji, R. 2023. Cheap and quick: Efficient vision-language instruction tuning for large language models. *arXiv preprint arXiv:2305.15023*.

Madaan, A.; Tandon, N.; Gupta, P.; Hallinan, S.; Gao, L.; Wiegreffe, S.; Alon, U.; Dziri, N.; Prabhumoye, S.; Yang, Y.; et al. 2023. Self-refine: Iterative refinement with self-feedback. *arXiv preprint arXiv:2303.17651*.

Monarch, R.; and Munro, R. 2021. *Human-in-the-Loop Machine Learning: Active learning and annotation for human-centered AI.* Simon and Schuster.

Northcutt, C.; Jiang, L.; and Chuang, I. 2021. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70: 1373–1411.

OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744.

Ramanathan, V.; Kalia, A.; Petrovic, V.; Wen, Y.; Zheng, B.; Guo, B.; Wang, R.; Marquez, A.; Kovvuri, R.; Kadian, A.; et al. 2023. PACO: Parts and Attributes of Common Objects. *arXiv preprint arXiv:2301.01795*.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*.

Tang, Z.; Yang, Z.; Zhu, C.; Zeng, M.; and Bansal, M. 2023. Any-to-Any Generation via Composable Diffusion. *arXiv preprint arXiv:2305.11846*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q. V.; Chi, E. H.; Narang, S.; Chowdhery, A.; and Zhou, D. 2022. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *ICLR*.

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, D.; Chen, K.; Rao, J.; Guo, X.; Zhang, Y.; Yang, J.; and Zhang, Y. 2023. Tackling Vision Language Tasks Through Learning Inner Monologues. *arXiv preprint arXiv:2308.09970*.

Yang, K.; Russakovsky, O.; and Deng, J. 2019. SpatialSense: An Adversarially Crowdsourced Benchmark for Spatial Relation Recognition. In *International Conference on Computer Vision (ICCV)*.

Yarom, M.; Bitton, Y.; Changpinyo, S.; Aharoni, R.; Herzig, J.; Lang, O.; Ofek, E.; and Szpektor, I. 2023. What You See is What You Read? Improving Text-Image Alignment Evaluation. *arXiv preprint arXiv:2305.10400*.

Ye, Q.; Xu, H.; Xu, G.; Ye, J.; Yan, M.; Zhou, Y.; Wang, J.; Hu, A.; Shi, P.; Shi, Y.; Jiang, C.; Li, C.; Xu, Y.; Chen, H.; Tian, J.; Qi, Q.; Zhang, J.; and Huang, F. 2023. mPLUG-Owl: Modularization Empowers Large Language Models with Multimodality. arXiv:2304.14178.

Yin, Z.; Wang, J.; Cao, J.; Shi, Z.; Liu, D.; Li, M.; Sheng, L.; Bai, L.; Huang, X.; Wang, Z.; et al. 2023. LAMM: Language-Assisted Multi-Modal Instruction-Tuning Dataset, Framework, and Benchmark. *arXiv preprint arXiv:2306.06687*.

Young, P.; Lai, A.; Hodosh, M.; and Hockenmaier, J. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2: 67–78.

Yu, W.; Yang, Z.; Li, L.; Wang, J.; Lin, K.; Liu, Z.; Wang, X.; and Wang, L. 2023. MM-Vet: Evaluating Large Multimodal Models for Integrated Capabilities. *arXiv preprint arXiv:2308.02490*.

Yun, T.; Sun, C.; and Pavlick, E. 2021. Does Vision-and-Language Pretraining Improve Lexical Grounding? In *Findings-EMNLP*.

Zha, D.; Bhat, Z. P.; Lai, K.-H.; Yang, F.; and Hu, X. 2023. Data-centric ai: Perspectives and challenges. In *SDM*, 945–948.

Zhang, Z.; Zhang, A.; Li, M.; Zhao, H.; Karypis, G.; and Smola, A. 2023. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*.

Zhao, Y.; Pang, T.; Du, C.; Yang, X.; Li, C.; Cheung, N.-M.; and Lin, M. 2023a. On evaluating adversarial robustness of large vision-language models. *arXiv preprint arXiv:2305.16934*.

Zhao, Z.; Guo, L.; Yue, T.; Chen, S.; Shao, S.; Zhu, X.; Yuan, Z.; and Liu, J. 2023b. Chatbridge: Bridging modalities with large language model as a language catalyst. *arXiv preprint arXiv:2305.16103*.

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*.

Appendices

Implementations of Compared LVLMs

Implementations of the compared baselines are from codebases of original authors, with their open-sourced URLs as follows:

- LLaMA2-chat (Touvron et al. 2023): https://github.com/facebookresearch/llama, with the *LLaMA2-7B-chat* checkpoint.
- mPLUG-owl (Ye et al. 2023): https://github.com/X-PLUG/mPLUG-Owl, with the *mPLUG-Owl-7B* checkpoint.
- LLaMA-AdapterV2 (Gao et al. 2023): https://github.com/OpenGVLab/LLaMA-Adapter, with the *LLaMA-AdapterV2-multimodal* checkpoint.
- LaVIN (Luo et al. 2023): https://github.com/luogen1996/LaVIN, with the *LaVIN-7B* checkpoint.
- MiniGPT-4 (Zhu et al. 2023): https://github.com/Vision-CAIR/MiniGPT-4, with the *MiniGPT-4-Vicuna7B-V0* checkpoint.
- LLaVA (Liu et al. 2023a): https://github.com/haotian-liu/LLaVA, with the *LLaVA-Vicuna7B-v1.1* checkpoint.
- Otter (Li et al. 2023a): https://github.com/Luodian/Otter, with the *OTTER-Image-MPT7B* checkpoint.
- InstructBLIP (Dai et al. 2023): https://github.com/salesforce/LAVIS/tree/main/projects/instructblip, with the *blip2-vicuna7B-instruct* checkpoint.

As can be seen, we use 7B parameter size checkpoints for all compared LVLMs and the text-only LLaMA2-chat. All LVLMs utilize LLaMA1 as the large language model component for a fair comparison. The hyper-parameters for inference (*e.g.* temperature, number of beams, *etc.*) for each LVLM are directly taken from the default ones recommended by their authors in the original codebases, respectively.

For instruction tuning, we also keep the default hyper-parameters (*e.g.* batch size, optimizer, learning rate, *etc.*) except explicitly set the *max tuning epoch* as 3 for all LVLMs. Specifically, for instruction tuning of LLaMA-AdapterV2, the visual projector and language model adaptor are updated while leaving the pre-trained visual encoder and the pre-trained language model frozen. For mPLUG-Owl, only the pre-trained language model is updated. For LaVIN, the visual encoder adaptors, the visual projector, and the language model adaptors (namely "Mixture-of-Modality Adapter" in the original paper) are updated. For LLaVA, the visual projector and the language model are updated. For Otter, the Perceiver resampler of the visual encoder, input/output embeddings of the language model, and the cross-attention layers are updated.

Case Studies for Enhanced LVLMs

We investigate the differences between responses from original LVLMs and responses from treated LVLMs. Table 7 shows two real examples using the same data source with free-form questions. The compared LVLM is LLaVA, between the original 7B checkpoint version and the instruction-tuned version (denoted by *[method+]*). In Case #1, the image depicts a well-furnished bedroom softly illuminated, creating a cozy and inviting ambiance. A red bounding box highlights a chair, while a green bounding box highlights a bed adorned with a blue comforter and pillows. Before instruction tuning, LLaVA struggled to provide accurate responses to questions about this image. For example, LLaVA mistakenly identified the chair as highlighted by the green bounding box, and the bed as highlighted by the red bounding box. In contrast, LLaVA+ produced more accurate responses to these questions. Furthermore, when asked about the spatial relationship between the chair and the bed, LLaVA responded incorrectly, stating that "the chair is in front of the bed." In contrast, LLaVA+ correctly indicated that the bed is to the left of the chair. In Case#2, the image features a woman wearing a Santa hat while riding a surfboard in the water. This time, both LLaVA and LLaVA+ successfully identify the object within the green bounding box, which is the woman. When moving to the red bounding box, LLaVA fails to ground it and still recognizes it as the woman, while LLaVA+ adeptly recognizes it as the surfboard. LLaVA's failure with the red bounding box results in an incorrect response to the subsequent question concerning the interaction between these two bounding boxes. These two illustrative examples support the effectiveness of our proposed data-centric enhancement method in aiding LVLMs in recognizing and interpreting nuanced multimodal information.

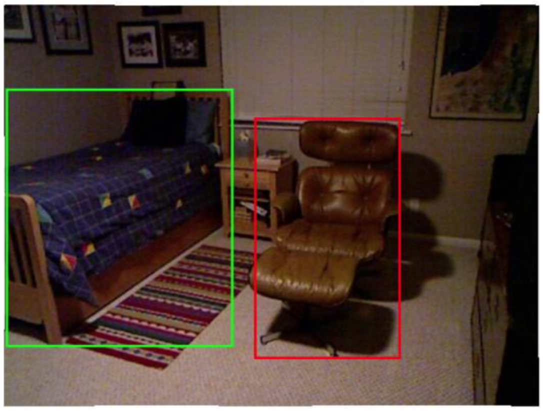
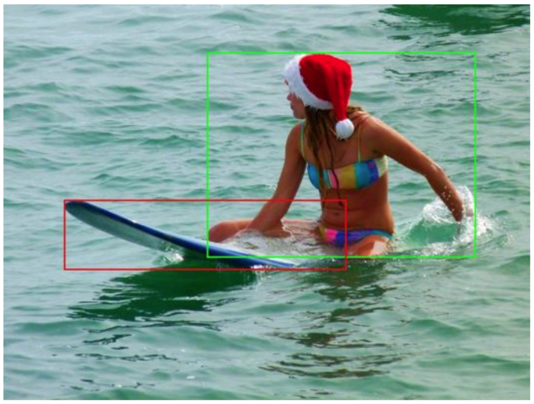
Case#1	
	 Source: SpatialSense (Yang, Russakovsky, and Deng 2019)
User	What is in the green bounding box in the image?
LLaVA	In the green bounding box in the image, there is a chair.
LLaVA+	The image shows the bed located in the green bounding box.
User	What is in the red bounding box in the image?
LLaVA	In the red bounding box in the image, there is a bed.
LLaVA+	The image shows the chair located in the green bounding box.
User	What is the spatial relationship between them?
LLaVA	The bed is located next to the chair, with the chair positioned in front of the bed.
LLaVA+	In the image, the bed in the green bounding box is to the left of the chair in the red bounding box.
Case#2	
	 Source: OpenImage-V7 (Krasin et al. 2017)
User	What is in the green bounding box?
LLaVA	In the green bounding box in the image, there is a woman wearing a Santa hat while riding a surfboard in the ocean.
LLaVA+	The image displays the woman located in the green bounding box.
User	What is in the red bounding box?
LLaVA	In the red bounding box, there is a close-up of a woman wearing a Santa hat while riding a surfboard in the ocean.
LLaVA+	The surfboard is situated within the red bounding box in the picture.
User	What is the interaction between them?
LLaVA	The interaction between the objects in the red bounding box and the green bounding box is that the woman in the red box is riding a surfboard in the ocean, while the woman in the green box is also riding a surfboard in the ocean.
LLaVA+	In the context of the image, the woman in the red bounding box surf the surfboard in the green bounding box.

Table 7: Case study regarding spatial misgrounding. LLaVA+ denotes the enhanced LLaVA using our proposed instruction tuning data.Text in cyan indicates the precise statement. Text in orange indicates the ambiguous statement.