

Generate, but Verify: Reducing Hallucination in Vision-Language Models with Retrospective Resampling

Tsung-Han Wu¹
Joseph E. Gonzalez¹

Heekyung Lee^{1,2}
Trevor Darrell¹

Jixin Ge¹
David M. Chan¹

¹University of California, Berkeley ²POSTECH

Abstract

Vision-Language Models (VLMs) excel at visual understanding but often suffer from visual hallucinations, where they generate descriptions of nonexistent objects, actions, or concepts, posing significant risks in safety-critical applications. Existing hallucination mitigation methods typically follow one of two paradigms: generation adjustment, which modifies decoding behavior to align text with visual inputs, and post-hoc verification, where external models assess and correct outputs. While effective, generation adjustment methods often rely on heuristics and lack correction mechanisms, while post-hoc verification is complicated, typically requiring multiple models and tending to reject outputs rather than refine them. In this work, we introduce REVERSE, a unified framework that integrates hallucination-aware training with on-the-fly self-verification. By leveraging a new hallucination-verification dataset containing over 1.3M semi-synthetic samples, along with a novel inference-time retrospective resampling technique, our approach enables VLMs to both detect hallucinations during generation and dynamically revise those hallucinations. Our evaluations show that REVERSE achieves state-of-the-art hallucination reduction, outperforming the best existing methods by up to 12% on CHAIR-MSCOCO and 28% on HaloQuest. Our dataset, model, and code are available at: <https://reverse-vlm.github.io>.

1. Introduction

Vision-Language Models (VLMs) have revolutionized visual understanding, achieving dramatic improvements in tasks like visual question-answering and image captioning, yet they still struggle with a significant limitation: visual hallucination — the tendency to describe objects that aren't actually present in the scene. Such hallucinations pose significant risks when applying VLMs to safety-critical environments, ranging from autonomous driving scenarios and decision-making to assistive technologies for the visually impaired.

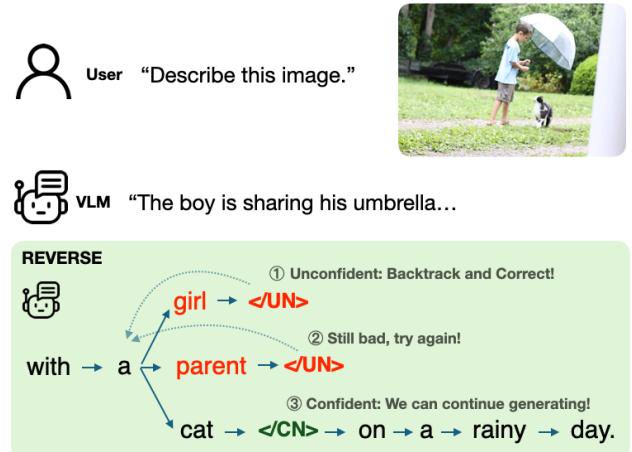


Figure 1. REVERSE, our proposed training and decoding paradigm for hallucination reduction, enables a single VLM to both **verify** if it has generated a hallucination and then **correct** itself iteratively. When uncertainty is detected through the generation of a (*</UN>*), the model backtracks and regenerates until a confident phrase (*</CN>*) is found.

To tackle these issues, researchers have generally pursued methods following one of two paradigms: *generation adjustment* or *post-hoc verification*. Generation adjustment methods focus on aligning textual outputs more closely with visual inputs by modifying the VLM's generation behavior, either in a “training-free” way (modifying the logits at decoding time) [4, 13–15, 38], or using a “training-based” strategy requiring additional supervision or custom objective functions [20, 27, 28, 34–36]. Unfortunately, these methods have no means of correcting erroneous tokens once they have been generated, and they do not leverage powerful retrospective tools such as chain-of-thought reasoning to reason about and evaluate the quality of their generation. In contrast to generation adjustment approaches, post-hoc verification methods [24, 28, 31, 33, 37] leverage large external models, such as GPT-4 [23], as verifiers to evaluate outputs *after they have been generated*. Post-hoc verifiers are accurate at predicting hallucination, but are complicated, requiring multiple models. Post-hoc verifiers often

do not provide a way for the model to *correct* the hallucination but instead adopt generic refusal strategies.

In this paper, we introduce **REVERSE** (*RE*troscopic *V*erification and *S*elf-correction), the first framework that unifies generation adjustment with online post-hoc verification in a single VLM architecture (Figure 1). REVERSE consists of two key components. First, our method fine-tunes a VLM on a specially constructed training dataset consisting of synthetic hallucination phrases tagged by a special token. Unlike prior VLMs instructed with only well-grounded data, our resulting hallucination-aware model is now able to tag likely phrase-level hallucinations during the generation process.

Second, we introduce retrospective resampling, a technique that allows the hallucination-aware VLM to serve as its own verifier. During the generation process, when the hallucination-aware VLM places sufficient probability on the special hallucination token we trigger a backtracking self-correction process. Specifically, we backtrack to a previous confident section and then apply rejection sampling and query-rewriting to correct the hallucination. As illustrated in Figure 1, both components in REVERSE combine to effectively mitigate visual hallucinations.

We evaluate REVERSE against SOTA hallucination reduction baselines across a wide range of benchmarks designed for hallucination evaluation. On hallucination-sensitive captioning and open-ended tasks, REVERSE achieves up to a 12% reduction in CHAIR scores on CHAIR-MSCOCO [26] and AMBER [30] over the best existing methods. It also delivers over a 10% and 28% performance improvement on MMHal [28] and HaloQuest [31], respectively. For tasks with discriminative (binary or multiple-choice) questions, REVERSE achieves performance on par with the best-performing models, demonstrating its robustness across different task formats.

In summary, this paper both (i) introduces REVERSE, the first hallucination reduction method unifying the generation adjustment and post-hoc verification approaches, addressing hallucination in both the training and inference stages and (ii) provides a new public training dataset and data curation pipeline for training-time hallucination mitigation consisting of 1.3M semi-synthetic samples. Together, these contributions allow REVERSE to achieve up to a 12% improvement on the CHAIR-MSCOCO benchmark, and a 28% improvement on the HaloQuest benchmark over existing SOTA methods for hallucination reduction under the same setting, especially in questions with false premise and insufficient context.

2. Background & Related Work

Following the success of Large Language Models (LLMs) [1, 5, 7, 19, 29], Vision-Language Models (VLMs) have shown success across various multimodal tasks, such as image captioning, visual question answering, visual reasoning, and image segmentation [2, 3, 16, 21–23]. Despite their impressive

performance, VLMs are prone to hallucinations: generating incorrect or nonexistent visual information [17]. To address this issue, several hallucination-specific benchmarks, such as CHAIR-MSCOCO [26], AMBER [30], MMHal [28], and POPE [17], and HaloQuest [31] have been introduced. These benchmarks evaluate VLM hallucinations across both discriminative and generative tasks, with a growing trend of using VLMs for automatic visual hallucination detection.

Beyond detection, several recent methods attempt to mitigate hallucinations by adjusting a VLM’s generation process. Training-free approaches primarily focus on improving decoding strategies [4, 13–15, 38]. For instance, VCD [15] employs contrastive decoding, OPERA [13] introduces a penalty term during beam search, and DoLA [9] enhances decoding by contrasting different model layers. Training-based methods, on the other hand, aim to reduce hallucinations through improved training objectives and additional data. Some approaches leverage data augmentation [6], while others refine training via reinforcement learning from human feedback (RLHF) [28, 34, 36]. Additional methods fine-tune VLMs using custom loss functions, such as EOS token-based penalties for lengthy descriptions [35], contrastive learning from paired correct-hallucination data [14], and visual instruction tuning with improved datasets [20]. However, these approaches merely adjust the generator’s behavior rather than fundamentally eliminating hallucinations—once incorrect information is produced, there is no built-in mechanism for correction.

The closest prior work to ours includes Woodpecker [33] and LURE [37], which use external models to verify and rewrite initial outputs from a VLM. While effective to some extent, these methods rely on complex, multi-stage pipelines with external dependencies. Moreover, they suffer from error propagation, as a single-round rewriting step is often insufficient to fully recover from low-quality initial outputs.

In contrast, our method is the first unified framework where the VLM itself serves as both the generator and verifier, enabling self-correction in a streamlined and integrated manner. Compared to prior generation-adjustment methods, our self-verification pipeline allows VLMs to retrospect and iteratively self-correct after content has been generated. Compared to existing post-hoc refinement approaches, our method eliminates the need for external models or complex multi-stage pipelines, achieving better results as the verifier can instantly and iteratively correct the generator’s outputs.

3. REVERSE: Retrospective Verification and Self-Correction

REVERSE (*RE*troscopic *V*erification and *S*elf-correction) is a hallucination reduction paradigm for Vision-Language Models (VLMs) that unifies generation adjustment and post-hoc verification methods. REVERSE allows VLMs to be hallucination-

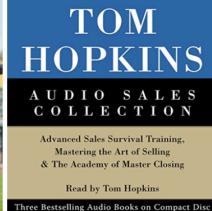
Image					Attribute Object World Entity Scene/View
Question	How many total baseball players are shown in the image?	Who wrote this book?	Describe the following image.	Where are the elephants located?	
Pos Answer	There are three baseball players shown in the image.	Tom Hopkins	The image features an old military aircraft on display ...	The elephants are located in a dry lot , which	
Neg Answer	There are five soccer players .	John Steinbeck	The image features a modern commercial airplane	The elephants are located in a muddy swamp	

Figure 2. Our 1.3M semi-synthetic instruction-tuning dataset for hallucination-aware VLM training. We constructed the dataset by augmenting negative phrases from the original LLaVA-v1.5-665k [22] dataset. Our negative phrases span a diverse range, including attributes, objects, world entities, and novel scenes. Positive noun phrases are marked with `` and `</CN>`, while negative samples are enclosed with `` and `</UN>`, terminating immediately. Further details about our dataset creation and statistics can be found in subsection 3.1 and Appendix B.

aware by explicitly modeling and monitoring the likelihood that each generated phrase is well-grounded. During training, the model is explicitly trained to classify each groundable phrase as either “confident” or “unconfident” and during inference, the model generates responses while continuously verifying the confidence of each phrase using the likelihood of the “unconfident” predictor. If a phrase is sufficiently ungrounded, the model then performs retrospective adjustment to refine the segment, enabling self-correction on the fly.

Key to the first goal of classifying each phrase as “confident” or “unconfident” is training the model to understand if a phrase is well-grounded. While VLMs and LLMs inherently provide implicit confidence scores through token probabilities, these scores are often mis-calibrated and do not consistently correlate with output correctness, making them unreliable for verification [8, 32]. Furthermore, even when accurate, these probabilities offer no indication of where to backtrack for phrase re-generation and self-correction.

To overcome these limitations, we introduce three tokens to the VLM vocabulary that can be used to explicitly mark key phrases and represent the model’s confidence level:

- ``: Marks the beginning of key or object phrases.
- `</CN>`: Marks the end of confident, grounded phrases.
- `</UN>`: Marks the end of unconfident, hallucinated phrases.

These tokens, when placed before/after objects or phrases in the scene can serve as ad-hoc classifiers of the confidence of the model. I.e. if a model generates a `</UN>` token after a phrase, that phrase can be considered to be ungrounded, while if it generates a `</CN>`, that phrase is likely grounded in the image. Annotating our data with such tokens, as is shown in Figure 2, will allow us to train the VLM itself to perform

post-hoc verification instead of relying on an external model.

3.1. Data Curation

Towards models that are capable of automatically tagging phrases as confident or unconfident, we constructed a 1.3M VLM instruction-tuning dataset containing a total of 6,810,679 question-answer pairs (turns), of which 3,855,303 (56.6%) are correct answers and 2,955,376 (43.4%) are hallucinated answers. In each response, all positive phrases are enclosed with `` and `</CN>` while the negative phrases are enclosed with `</UN>`.

The phrases in this dataset are automatically generated by annotating existing training data for VLM with these tokens. Our dataset is initially sourced from the LLaVA-v1.5-665k instruction tuning data [22], which contains only “positive” or “well-grounded” samples. To introduce “negative” or “un-grounded” samples, we designed a multi-step pipeline that generates incorrect answers leveraging rule-based algorithms and gpt-4o-mini-0718 [23]. Specifically, we first classify the answer types, in which we can augment most of the answers with rule-based methods easily, such as binary Yes/No questions or counting questions. For the remaining general answers, mostly long answers or descriptions, we apply AI inference for high-quality and diverse data augmentation. For all negative samples, we enforce the constraint that the sentence immediately terminates upon reaching `</UN>`. This design not only prevents VLMs from continuing to generate ungrounded descriptions after detecting hallucinated content but also helps maintain training data quality, as any remaining context may become meaningless once the preceding information has been altered. To support retrospective query re-writing, we further inject

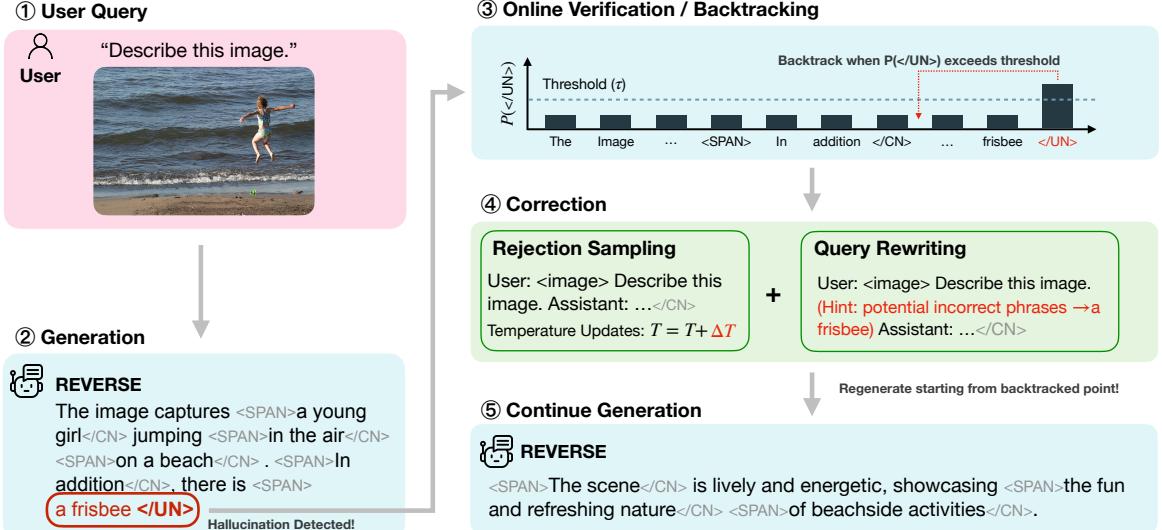


Figure 3. **Illustration of our retrospective resampling process.** During inference, we monitor the hallucination-aware VLM’s generation. When the likelihood of the $</UN>$ token surpasses a predefined threshold, we trigger backtracking to the most recent confident checkpoint ($</CN>$) and apply corrections using rejection sampling and query rewriting. This self-correction mechanism can be applied iteratively throughout the generation process.

negative keywords as hints (further discussed in subsection 3.3).

Our dataset has twice the number of samples compared to the LLaVA-v1.5-665k instruction tuning dataset while maintaining a similar overall composition. It preserves the same average question-answer pairs per sample and a comparable question type distribution. More details about the dataset statistics and the whole generation pipeline are provided in Appendix B.

3.2. Hallucination-aware Training

To train REVERSE to recognize and respond to the new tokens, we introduce a modified cross-entropy next-token prediction loss that prevents hallucination while modeling confidence levels. Our training objectives are threefold. First, we aim to enable conventional instruction tuning to allow VLMs to perform next-token prediction to generate accurate answers. Second, we wish to reduce the likelihood of generating the hallucinated tokens that we have introduced in the new dataset. Third, we want to teach the model to generate $$ at the start of key phrases, and $</CN>$ or $</UN>$ as explicit confidence estimators around those phrases. We achieve all of these goals by assigning a weight to each token during training; positive weights are assigned to tokens outside the $... </UN>$ bounds, encouraging standard next-token prediction while zero-weights are assigned to tokens within $$ and $</UN>$ (i.e., masking out the targets) to avoid impacting the likelihood when training on ungrounded data (and reinforcing language priors).

Formally, let θ be our model and D be the labeled VQA dataset, where each sample S consists of an input sequence $X = \{x_1, x_2, \dots, x_m\}$ and an output sequence $Y = \{y_1, y_2, \dots, y_n\}$. Here, X includes both encoded image features and question (query) tokens, while each y_i in Y can be either a text token

corresponding to the answer or one of the three special tokens.

The model θ predicts the next token probability as follows:

$$P(y_i | x_1, x_2, \dots, x_m, y_1, y_2, \dots, y_{i-1}; \theta) \quad (1)$$

We then define the modified negative log-likelihood loss for a given sample as:

$$L(S) = - \sum_{y_i \in Y} \mathbb{1}_{Hall(i)} \cdot \log P(y_i | X, y_1, \dots, y_{i-1}; \theta) \quad (2)$$

where $\mathbb{1}_{Hall(i)} \in \{0, 1\}$ is an indicator variable taking the value 1 for all tokens **except** those enclosed by $$ and $</UN>$. For tokens within these markers, $\mathbb{1}_{Hall(i)} = 0$. During training, we optimize our model θ using Equation 2. A more detailed discussion of the training procedure is provided in section 4 and Appendix E.

3.3. Retrospective Resampling

During inference, the model follows standard next-token prediction but continuously monitors the likelihood of $</UN>$, triggering retrospective resampling when that likelihood exceeds a pre-defined threshold (see Figure 3). Specifically, whenever $</CN>$ or $</UN>$ is generated (the end of a span), we compute the probability of $</UN>$, denoted as $P(</UN>)$, across previous tokens. If $P(</UN>)$ surpasses a predefined threshold τ , the model initiates a self-correction process via backtracking and retrospective resampling. Otherwise, generation proceeds normally, with $$, $</CN>$ and $</UN>$ tokens removed before presenting the final output.

Backtracking Strategies A critical challenge in self-correction is determining both (1) where to backtrack and (2)

how to regenerate content. To determine where to backtrack to, our approach follows a hierarchical fallback strategy:

1. The model first backtracks to the most recent $\langle/\text{CN}\rangle$, which attempts to adjust only the local information to reduce the likelihood of hallucination.
2. If the issue persists after K local correction attempts, it is likely that the hallucination issue stems from earlier information in the sequence. Thus, we revert further, backtracking to the last sentence boundary (indicated by the last punctuation token).
3. If self-correction continues to fail after N total attempts, the output is finalized and returned to the user, along with an indication that a hallucination was detected, but could not be corrected.

Since $P(\langle/\text{UN}\rangle)$ is typically low, explicitly waiting for $\langle/\text{UN}\rangle$ to appear in the output is impractical. Instead, setting the confidence threshold τ allows proactive identification of hallucinated phrases before they fully form, as well as allows for cautionary reduction of hallucination (even when the likelihood is low). The effect of τ selection is analyzed in subsection 4.2.

After the backtracking has occurred, we leverage *rejection sampling* and *query rewriting* for self-correction.

Rejection Sampling Rejection sampling refines uncertain phrases by resampling multiple times at an increased temperature, seeking an alternative where $P(\langle/\text{UN}\rangle)$ remains below τ . The process continues until reaching a confident phrase (marked by $\langle/\text{CN}\rangle$) or exhausting the maximum resampling attempts. In this work, we make no attempts to “ban” the generation of the same tokens during the re-sampling process. This procedure, while potentially more efficient, often leads to issues where innocuous tokens such as “a” or “the” are banned, leading to disfluencies in the final generated text. Instead, we rely on the increased temperature to lead to new candidates over several repeated generations. While rejection sampling is effective for resolving localized hallucinations, its success depends on the ability of the model to generate valid alternatives within a reasonable number of attempts. In cases where repeated resampling fails to produce an acceptable phrase, the model may need to fall back on broader correction strategies, such as query rewriting, to address deeper inconsistencies in the generated content.

Query Rewriting In addition to rejection sampling, we found that “query rewriting” can provide stronger signals for VLMs to do self-correction. Query rewriting dynamically modifies the prompt to encourage better factual grounding. Specifically, the input prompt is augmented with a clarification hint:

```
<sys-prompt> <image> <question>
(Hint: potential incorrect phrases →
<placeholder>)
```

This signals the model to reconsider flagged segments and

generate a more reliable response. In addition to rejection sampling with increased temperature, which iteratively refines outputs by resampling under varied decoding conditions, query rewriting can directly influence the model’s contextual understanding by reformulating its input conditions. Since our training data includes hallucination-corrected phrase pairs, we randomly inject 20% of this query-rewriting prompt into the instruction-tuning process. This improves the model’s ability to recognize the hint, making retrospective resampling more effective.

The full decoding algorithm, including hyperparameter choices such as correction attempt limits, backtracking depth, and temperature scaling, is detailed in the following section and Appendix E. Ablation studies analyzing their impact are presented in subsection 4.2.

4. Experiments

Implementation Details We applied our method, REVERSE, on two VLM backbones: LLaVA-v1.5-7B [22] and LLaVA-More (LLaVA with Llama-v3.1) [2, 10]. For training, REVERSE is instruction fine-tuned using LoRA (rank=128, $\alpha = 256$) with the modified cross-entropy loss (described in subsection 3.2) on our 1.3M dataset for one epoch. More details on additional training parameters are provided in Appendix E.

During inference, we apply retrospective resampling with different threshold values: $\tau = 0.003$ for generative tasks and $\tau = 0.5$ for discriminative tasks (e.g., binary and multiple-choice questions) to optimize performance. Further discussions on this can be found in subsection 4.2. For the correction mechanism, we allow up to a $N = 50$ total correction attempts, with local correction attempts of $K = 10$. Additionally, we implement rejection sampling with a base temperature of T_0 , gradually increasing it with a step size of $\Delta T = 0.1$, capped at a maximum temperature of $T_0 + 0.5$: $T = \min(T + \Delta T, T_0 + 0.5)$.

Evaluation Protocol To evaluate our approach, we compare REVERSE with various hallucination mitigation methods, including training-free or training-based generative adjustment techniques [4, 9, 13, 15, 27, 35, 36, 38], and post-hoc verification with refinement [33]. All methods are evaluated on both VLM backbones under consistent settings, where we fix the decoding temperature at 0 and use only the base prompts provided by each dataset to ensure fair comparisons. Since REVERSE does stochastic sampling at inference time, we report the mean performance over 100 bootstrapped runs for robustness. The exact numbers with 95% confidence intervals are provided in Appendix D.

Our evaluation spans multiple common VQA tasks designed to assess visual hallucination, categorized into image captioning, open-ended question answering, and discriminative tasks. For image captioning, we use CHAIR-MSCOCO [26, 35] and the generative subset of AMBER [30] (denoted as AMBER-G). CHAIR-MSCOCO evaluates object hallucination using the

Table 1. Performance comparison of various hallucination reduction methods across various image captioning benchmarks, which are commonly used to evaluate visual hallucinations in generative tasks for VLMs. This includes the CHAIR-MSCOCO benchmarks from [35] and the generative subset of AMBER. \dagger and \ddagger mean that we reproduced the results of these methods on CHAIR-MSCOCO and AMBER-G respectively. Otherwise, numbers came from [35] and [27].

Base VLM	Method Type	Method	CHAIR-MSCOCO		AMBER-G		
			CHAIR _i (\downarrow)	CHAIR _s (\downarrow)	CHAIR (\downarrow)	Cover (\uparrow)	Hall (\downarrow)
	None		15.4	50.0	7.8	51.0	36.4
LLaVA-v1.5 7B [22]	Gen-Adjust w/o Train	VCD [15]	14.9	48.6	-	-	-
		OPERA \ddagger [13]	14.6	47.8	7.3	49.6	32.0
		DoLA $\dagger\ddagger$ [9]	14.1	51.6	7.6	51.6	36.0
		AGLA [4]	14.1	43.0	-	-	-
		MEMVR [38]	13.0	46.6	-	-	-
LLaVA-v1.5 7B [22]	Gen-Adjust w/ Train	EOS [35]	12.3	40.2	5.1	49.1	22.7
		HALVA [27]	11.7	41.4	6.6	53.0	32.2
		HA-DPO [36]	11.0	38.2	6.7	49.8	30.9
LLaVA-MORE 8B [10]	Post-hoc Refine	Woodpecker \dagger [33]	14.8	45.8	6.9	48.9	30.4
	Combination	REVERSE $_{(\tau=0.003)}$	10.3	37.0	6.0	52.2	30.4
		REVERSE $_{(\tau=0.0003)}$	6.1	13.6	4.0	26.9	10.2
	LLaVA-MORE 8B [10]	None	14.4	52.0	7.8	53.1	36.6
		DoLA [9]	13.8	51.8	7.9	53.1	38.4
		Woodpecker [33]	14.3	51.0	7.4	50.7	36.7
		REVERSE $_{(\tau=0.003)}$	12.2	42.4	6.5	54.8	35.5
		REVERSE $_{(\tau=0.0003)}$	8.4	25.2	5.1	38.9	20.8
							2.1

CHAIR score, which measures the degree of misalignment between objects mentioned in a model-generated caption and objects actually present in the image. It is defined as 1 minus the intersection over union (IoU) between the sets of mentioned and ground-truth objects. We report both CHAIR_i, which aggregates the CHAIR score across all object instances, and CHAIR_s, which quantifies the proportion of images where at least one hallucination occurs. For AMBER-G, we report four key metrics: CHAIR, Coverage (Cover), Hallucination (Hall), and Cognition (Cog). CHAIR is the same as CHAIR_i above, and Coverage measures how well the caption mentions all objects in the image, similar to recall. The definitions of the remaining two metrics and further details on these datasets are provided in Appendix C.

For open-ended question answering, we evaluate on MMHal-Bench [28] and HaloQuest [31], which generally test VLMs on false-premise questions, questions with insufficient visual evidence, and visually complex queries. Following standard evaluation protocols, we assess MMHal-Bench responses using gpt-4-0314 and HaloQuest with Gemini-1.5-Pro, as the original paper used Gemini-1.0-Pro, which is no longer available. These benchmarks require models to generate free-form text answers, testing their ability to comprehend and reason about visual content in an open-ended manner.

For discriminative tasks, we utilize the discriminative subset of AMBER (AMBER-D) [30], POPE [17], and MME-Hal [11], all of which are binary (Yes/No) questions to assess a model’s understanding of object existence, attributes, spatial relationships, and locations.

Table 2. Performance comparison on MMHal-Bench [28] with different methods. Results re-implemented by us are marked with \dagger . We observe that lower thresholds ($\tau = 0.0003$) enable our VLM to handle false-premise questions—common in MMHal—leading to improved scores without the need for specialized training.

Base VLM	Method	Score (\uparrow)	Hall. Rate (\downarrow)
LLaVA-v1.0 7B	LLaVA-RLHF [28]	2.05	0.68
	None [22]	2.11	0.54
	HAACL [14]	2.13	0.50
	HA-DPO [36]	1.97	0.60
	EOS [35]	2.03	0.59
	HALVA [27]	2.25	0.54
LLaVA-v1.5 7B	DoLA \dagger [9]	2.33	0.56
	Woodpecker \dagger [33]	2.19	0.58
	REVERSE $_{(\tau=0.003)}$	2.56	0.47
	REVERSE $_{(\tau=0.0003)}$	3.28	0.30
LLaVA-MORE 8B	None \dagger	2.50	0.53
	DoLA \dagger	2.54	0.51
	Woodpecker \dagger	2.28	0.58
	REVERSE $_{(\tau=0.003)}$	2.28	0.54
	REVERSE $_{(\tau=0.0003)}$	2.93	0.40

4.1. Experimental Results

Image Captioning Tasks Table 1 presents results on image captioning tasks. With the default parameters ($\tau = 0.003$), our method reduces the CHAIR_i value by up to 12% on CHAIR-MSCOCO and AMBER-G compared to the best existing methods. In comparisons using LLaVA-v1.5-7B, training-free generative adjustments generally perform worse than training-based methods. The post-hoc refinement method, Woodpecker [33], employs a multi-stage process for verification and correction; however, its one-time correction approach may suffer from error propagation, limiting its effectiveness. Additionally, the EOS method [35] achieves significantly better results on

 Describe this image.



LLaVA

The image features a young woman playing soccer on a grassy field. She is running towards a soccer ball, which is positioned towards the right side of the field. The woman appears to be focused and determined as she approaches the ball. In the background, there are two other people, one on the left side and another on the right side of the field. The scene captures the excitement and energy of a soccer game.

OPERA

The image features a young woman playing soccer on a grassy field. She is running towards a soccer ball, which is positioned towards the right side of the field. The woman appears to be focused and determined as she approaches the ball. In the background, there is a bench located on the left side of the field, possibly for players to rest or for spectators to sit and watch the game.

Woodpecker

The image features a woman([0.067, 0.051, 0.453, 0.938]) playing **frisbee** on a grassy field([0.001, 0.002, 0.999, 0.998]). There is no soccer ball or soccer in the image. There is no ball or person in the image.

REVERSE

The image features a young woman wearing a black and white striped shirt, running through a lush green field. She is in the process of kicking a soccer ball with her leg, which is positioned towards the right side of the field. The woman appears to be enjoying her time playing soccer in the open space. The field is surrounded by grass, creating a natural and inviting environment for the game.



The image features a cat lying on a desk, resting its head on a computer keyboard. The keyboard is positioned in the lower left part of the scene, while the cat occupies the majority of the space, stretching from the center to the right side of the desk. In addition to the keyboard, there is a computer mouse located on the right side of the desk, close to the cat. The desk appears to be a workspace, with a few other items scattered around, such as a bowl and a bottle. The bowl is situated in the upper left corner, while the bottle is located in the upper right corner of the scene.

The image features a cat lying on a desk, resting its head on a computer keyboard. The keyboard is positioned in front of the cat, and a computer mouse can be seen nearby. The cat appears to be sleeping or relaxing on the desk, taking up a significant portion of the space. In the background, there is a bowl placed on the desk, possibly containing snacks for the cat or the person using the computer.

The image features a cat([...] lying on **two desks**[...]). The cat is scattered around the desks. The cat is resting its head on a computer keyboard([...]), which is positioned in the lower left part of the scene. In addition to the keyboard, there is a computer mouse([...]) located on the right side of the desk, close to the cat. The desk appears to be a workspace, with a few other items scattered around. There is no bowl in the scene. There is a bottle([...]) situated in the upper right corner of the image.

The image features a large cat lying on a wooden table, with a computer keyboard and a computer mouse placed in front of it. The cat appears to be sleeping or resting on the table, with its head resting on the keyboard. The computer mouse is positioned near the right side of the table, and the keyboard is on the left side. The scene suggests that the cat is in close proximity to the computer setup, possibly enjoying the warmth of the devices or simply resting on the table.

Figure 4. Qualitative Examples of different Methods. When generating captions for an image, LLaVA, OPERA, and Woodpecker tend to hallucinate non-existing objects. REVERSE generates correct captions of similar length. Additional qualitative results are provided in Appendix D.

Table 3. Performance comparison across different models on HaloQuest [31]. FP, VC, and IC stand for false premise, visually challenging, and insufficient context, three subsets in the benchmark dataset. With lower threshold ($\tau = 0.0003$), REVERSE shows a huge gain on unanswerable questions (e.g., those with insufficient context) even without being specifically trained on these cases.

Base VLM	Method	Avg. Acc. (\uparrow)	FP Acc.	VC Acc.	IC Acc.
LLaVA-v1.5	None [†]	22.6	17.1	39.5	10.7
	DoLA [†]	22.9	17.2	40.1	11.6
	HALVA [†]	23.9	21.1	37.4	10.7
	REVERSE $(\tau = 0.003)$	30.7	31.8	31.5	26.9
	REVERSE $(\tau = 0.0003)$	32.3	29.4	18.7	58.8
LLaVA-MORE	None [†]	22.4	15.8	43.4	7.4
	DoLA [†]	22.8	15.5	45.1	7.4
	REVERSE $(\tau = 0.003)$	26.7	30.0	31.3	11.7
	REVERSE $(\tau = 0.0003)$	36.7	39.5	30.9	38.1
GPT-4o		63.2	65.2	55.2	68.7
Gemini 1.5 Pro		77.9	83.7	56.3	92.5

AMBER-G, likely because it encourages the model to produce more concise captions, leading to less informative outputs and reduced coverage. Unlike prior methods, REVERSE is flexible across thresholds, allowing for a tradeoff between coverage and hallucination rate, allowing us to surpass GPT-4V on the CHAIR scores with the tradeoff of lower coverage as discussed in subsection 4.2. Figure 4 presents qualitative results using four different methods. LLaVA-v1.5-7B, OPERA, and Woodpecker hallucinate non-existing objects, while REVERSE can generate the correct caption without reducing caption length too much.

Open-ended Question Answering We use MMHal-Bench and HaloQuest as evaluation datasets for open-ended question answering. Both datasets include questions with false premises or insufficient context, requiring the model to either refuse or correct the query. For these questions, we observed that REVERSE often produces empty responses and we interpret this behavior as the model identifying the query as unanswerable. In all such cases we apply query rewriting

Table 4. Performance comparison across multiple discriminative visual hallucination benchmarks including the discriminative subset of AMBER (F1), POPE (F1), and the hallucination subset of MME (Score).

Base VLM	Method	AMBER-D	POPE	MME-Hall
LLaVA-v1.5	None	74.7	85.9	648.3
	VCD [15]	-	84.5	604.7
	EOS [35]	75.6	86.0	606.7
	OPERA [†] [13]	74.8	85.5	592.3
	DoLA [†] [9]	74.5	85.7	656.7
	HA-DPO [36]	78.1	86.9	618.3
	MEMVR [38]	-	85.9	648.3
	AGLA [4]	-	86.0	640.0
	HALVA [27]	83.4	84.8	665.0
	Woodpecker [33]	67.0	-	366.7
LLaVA-MORE	REVERSE $(\tau = 0.5)$	74.2	85.9	601.6
	None [†]	71.6	85.1	678.3
	DoLA [†] [9]	72.0	85.2	683.3
GPT-4o	REVERSE $(\tau = 0.5)$	69.3	84.4	657.6

with the prompt: “For this question, please point out the false premises or note what information is missing, rather than answering it directly.” The complete mechanism for handling unanswerable questions is given in Appendix E.

As shown in Table 2 and Table 3, REVERSE improves accuracy by up to 10% on MMHal-Bench and 28% on HaloQuest compared with the SOTA models using default hyperparameters ($\tau = 0.003$). The breakdown shows that most improvements come from better handling of false-premise and insufficient-context questions. However, performance on visually challenging questions decreases, as the model takes a more cautious stance and avoids speculative answers. Using a lower threshold ($\tau = 0.0003$) further increases this conservativeness, leading to more gains on unanswerable questions without requiring task-specific fine-tuning on both tasks.

Discriminative Questions Table 4 presents the results for three commonly used discriminative tasks. Our method maintains comparable results to existing methods but does not show significant improvement. This outcome may be due to

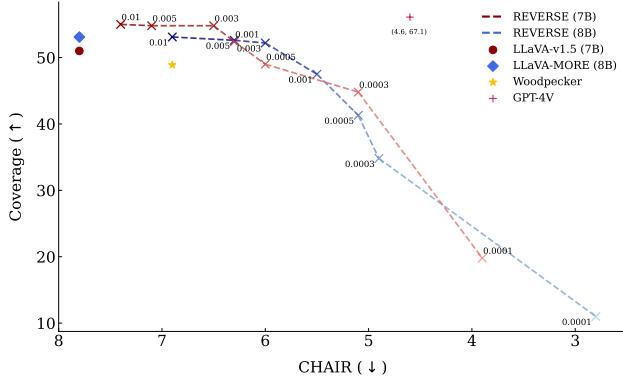


Figure 5. Ablation studies on the optimal threshold τ for generative tasks. This plot illustrates the trade-off between CHAIR (\downarrow) and Coverage (\uparrow) across different threshold values. REVERSE establishes a clear performance frontier, surpassing base models (LLaVA-v1.5, LLaVA-MORE) and the existing post-hoc refinement method (Woodpecker [33]). Notably, with appropriate threshold tuning, our approach can even achieve lower hallucination rates than GPT-4V. the nature of these tasks, which often require straightforward yes or no answers, rendering retrospective resampling or further reasoning less beneficial.

4.2. Discussions

Trade-offs Between Expressiveness and De-Hallucination
As discussed in subsection 3.3, a key component of our retrospective resampling method is the predefined threshold τ . When the predicted probability of $\langle /UN \rangle$ exceeds τ , the model triggers backtracking and self-correction. Figure 5 presents an analysis of the effect of τ on two VLMs. The 2D plot illustrates the performance trade-off between CHAIR (hallucination metric) and coverage across different threshold values. Based on our experiments, we selected $\tau = 0.003$ as a global threshold for image captioning and other generative tasks, as it represents the peak of the performance frontier. For discriminative tasks (e.g., binary and multiple-choice questions), where retrospective resampling has limited impact, we choose $\tau = 0.5$ as the default, and increasing the threshold beyond 0.5 does not significantly affect the results.

Compared with prior hallucination reduction methods, such as the post-hoc correction techniques [33], REVERSE enables us to dynamically adjust the balance between expressiveness and trustworthiness. Notably, with a relatively high threshold of $\tau = 0.01$, our method already surpasses the base VLMs (LLaVA-v1.5 and LLaVA-MORE) in both hallucination reduction and content coverage. Moreover, with a lower threshold ($\tau = 0.0001$), our model can even outperform GPT-4V in hallucination control.

Ablation Studies We conduct ablation studies to evaluate the contributions of different components of our method, as shown in Table 5. Comparing the first and second rows, we observe that hallucination-aware training alone already improves performance across all metrics, outperforming existing VLMs.

Table 5. Ablation studies demonstrate that hallucination-aware training enhances VLM performance across all metrics, achieving higher coverage and lower hallucination scores. Moreover, our retrospective resampling further reduces hallucination, with each component contributing to overall improvement.

Components	CHAIR (\downarrow)	Cover (\uparrow)	Hall (\downarrow)	Cog (\downarrow)
LLaVA-v1.5-7B	7.8	51.0	36.4	4.2
+ Hall-aware Training	7.2	53.2	36.3	3.4
+ Rejection Sampling ($\tau = 0.003$)	6.0	51.0	30.5	3.0
+ Query Rewriting ($\tau = 0.003$)	6.0	52.2	30.4	3.0

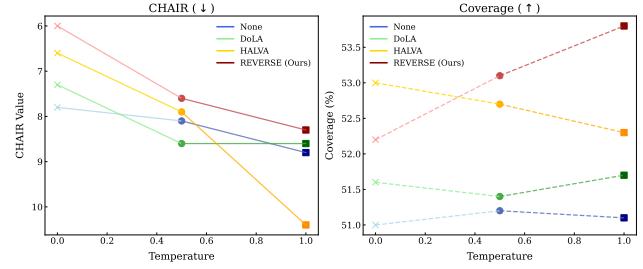


Figure 6. Further analyses on the effectiveness of temperature for answer quality. Ideally, increasing the temperature in image description tasks should encourage the model to capture more objects and details while also increasing its tolerance for hallucinated, incorrect content. REVERSE is the only approach that achieves the best hallucination-expressiveness tradeoff, maintaining higher coverage as temperature increases. Moreover, it consistently outperforms all existing methods on the hallucination metric, CHAIR, across all settings.

We hypothesize that this improvement arises from the model’s ability to contrast positive and negative phrases, effectively learning to distinguish between $\langle /CN \rangle$ and $\langle /UN \rangle$ during training—a mechanism that may be similar to DPO [25]. This finding suggests a potential research direction for future work. Interestingly, even a naive rejection sampling strategy reduces CHAIR hallucination scores by 1.2. When combined with query rewriting, the coverage improves by a further 1.2 points, indicating that rewriting helps the model explore alternative phrasing and correct itself more effectively.

Impact of Temperature on Hallucination and Coverage
We also analyze how temperature settings affect hallucination rates and object coverage in generated outputs. In this experiment, we use LLaVA-v1.5-7B as the backbone and increase the number of local correction attempts (K) while keeping all other parameters in REVERSE unchanged. As shown in Figure 6, REVERSE is robust to increasing temperature. For tasks such as image captioning, a higher temperature is often desirable to enhance diversity in generated descriptions. However, existing methods not only suffer from increased hallucinations at higher temperatures but also exhibit a decline in object coverage. In contrast, REVERSE balances expressiveness and reliability—slightly reducing hallucination while improving object coverage as temperature increases.

5. Conclusion

In this paper, we introduced REVERSE, a framework that reduces hallucinations in Vision-Language Models by combining hallucination-aware training with retrospective resampling. REVERSE achieves up to 12% improvement on image CHAIR-MSCOCO and 28% on HaloQuest over existing SOTA methods. While preliminary, REVERSE highlights the potential of self-correction in multimodal models. Future work may explore integrating structured verification and causal reasoning to further reduce hallucinations. As multimodal AI progresses, we see self-verification and retrospective techniques as a promising direction for building more trustworthy systems.

Acknowledgments

We thank Shalini Ghosh and Konpat Preechakul for their invaluable feedback during the discussions. Authors, as part of their affiliation with UC Berkeley, were supported in part by the National Science Foundation, US Department of Defense, and/or the Berkeley Artificial Intelligence Research (BAIR) industrial alliance program, as well as gifts from Amazon. Sky Computing Lab is supported by gifts from Accenture, AMD, Anyscale, Cisco, Google, IBM, Intel, Intesa Sanpaolo, Lambda, Microsoft, NVIDIA, Samsung SDS, SAP, and VMware. This research was also developed with funding from the Defense Advanced Research Projects Agency (DARPA) under Contract No. FA8650-23-C-7316. The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of any sponsor, the Department of Defense, or the U.S. Government.

References

- [1] The claude 3 model family: Opus, sonnet, haiku. [2](#)
- [2] Meta AI. Introducing meta llama 3: The most capable openly available llm to date, 2024. [2](#), [5](#)
- [3] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022. [2](#)
- [4] Wenbin An, Feng Tian, Sicong Leng, Jiahao Nie, Haonan Lin, QianYing Wang, Guang Dai, Ping Chen, and Shijian Lu. Agla: Mitigating object hallucinations in large vision-language models with assembly of global and local attention. *arXiv preprint arXiv:2406.12718*, 2024. [1](#), [2](#), [5](#), [6](#), [7](#)
- [5] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. [2](#)
- [6] Ali Furkan Biten, Lluís Gómez, and Dimosthenis Karatzas. Let there be a clock on the beach: Reducing object hallucination in image captioning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1381–1390, 2022. [2](#)
- [7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. [2](#)
- [8] Yu-Neng Chuang, Helen Zhou, Prathusha Sarma, Parikshit Gopalan, John Boccio, Sara Bolouki, and Xia Hu. Learning to route llms with confidence tokens. *arXiv preprint arXiv:2410.13284*, 2025. [3](#)
- [9] Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R. Glass, and Pengcheng He. Dola: Decoding by contrastive layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*, 2024. [2](#), [5](#), [6](#), [7](#)
- [10] Federico Cocchi, Nicholas Moratelli, Davide Caffagni, Sara Sarto, Lorenzo Baraldi, Marcella Cornia, and Rita Cucchiara. LLaVAMORE: A Comparative Study of LLMs and Visual Backbones for Enhanced Visual Instruction Tuning, 2025. [5](#), [6](#), [11](#), [14](#)
- [11] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. [6](#), [12](#)
- [12] Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spacy: Industrial-strength natural language processing in python, 2020. [11](#)
- [13] Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13418–13427, 2024. [1](#), [2](#), [5](#), [6](#), [7](#)
- [14] Chaoya Jiang, Haiyang Xu, Mengfan Dong, Jiaxing Chen, Wei Ye, Ming Yan, Qinghao Ye, Ji Zhang, Fei Huang, and Shikun Zhang. Hallucination augmented contrastive learning for multimodal large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27036–27046, 2024. [2](#), [6](#)
- [15] Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13872–13882, 2024. [1](#), [2](#), [5](#), [6](#), [7](#)
- [16] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. [2](#)
- [17] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models, 2023. [2](#), [6](#), [11](#)
- [18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6–12, 2014, proceedings, part v 13*, pages 740–755. Springer, 2014. [12](#)

- [19] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024. 2
- [20] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Mitigating hallucination in large multi-modal models via robust instruction tuning. In *The Twelfth International Conference on Learning Representations*, 2023. 1, 2
- [21] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 2, 11
- [22] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26296–26306, 2024. 3, 5, 6, 14
- [23] OpenAI. Hello gpt-4o, 2024. 1, 2, 3
- [24] Suzanne Petryk, David Chan, Anish Kachinthaya, Haodi Zou, John Canny, Joseph Gonzalez, and Trevor Darrell. ALOHa: A new measure for hallucination in captioning models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 342–357, Mexico City, Mexico, 2024. Association for Computational Linguistics. 1
- [25] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36: 53728–53741, 2023. 8
- [26] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4035–4045, Brussels, Belgium, 2018. Association for Computational Linguistics. 2, 5, 11, 12
- [27] Pritam Sarkar, Sayna Ebrahimi, Ali Etemad, Ahmad Beirami, Sercan O Arik, and Tomas Pfister. Data-augmented phrase-level alignment for mitigating object hallucination. In *The Thirteenth International Conference on Learning Representations*, 2025. 1, 5, 6, 7
- [28] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. Aligning large multimodal models with factually augmented RLHF. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13088–13110, Bangkok, Thailand, 2024. Association for Computational Linguistics. 1, 2, 6, 13
- [29] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023. 2
- [30] Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Ming Yan, Ji Zhang, and Jitao Sang. An llm-free multi-dimensional benchmark for mllms hallucination evaluation. *arXiv preprint arXiv:2311.07397*, 2023. 2, 5, 6, 11
- [31] Zhecan Wang, Garrett Bingham, Adams Wei Yu, Quoc V Le, Thang Luong, and Golnaz Ghiasi. Haloquest: A visual hallucination dataset for advancing multimodal reasoning. In *European Conference on Computer Vision*, pages 288–304. Springer, 2024. 1, 2, 6, 7, 13
- [32] Spencer Whitehead, Suzanne Petryk, Vedaad Shakib, Joseph Gonzalez, Trevor Darrell, Anna Rohrbach, and Marcus Rohrbach. Reliable visual question answering: Abstain rather than answer incorrectly. In *European Conference on Computer Vision*, pages 148–166. Springer, 2022. 3
- [33] Shukang Yin, Chaoyou Fu, Sirui Zhao, Tong Xu, Hao Wang, Dianbo Sui, Yunhang Shen, Ke Li, Xing Sun, and Enhong Chen. Woodpecker: Hallucination correction for multimodal large language models. *Science China Information Sciences*, 67(12): 220105, 2024. 1, 2, 5, 6, 7, 8
- [34] Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, and Tat-Seng Chua. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13807–13816, 2024. 1, 2
- [35] Zihao Yue, Liang Zhang, and Qin Jin. Less is more: Mitigating multimodal hallucination from an EOS decision perspective. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11766–11781, Bangkok, Thailand, 2024. Association for Computational Linguistics. 2, 5, 6, 7, 12
- [36] Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. Beyond hallucinations: Enhancing lmlms through hallucination-aware direct preference optimization. *arXiv preprint arXiv:2311.16839*, 2023. 1, 2, 5, 6, 7
- [37] Yiyang Zhou, Chenhang Cui, Jaehong Yoon, Linjun Zhang, Zhun Deng, Chelsea Finn, Mohit Bansal, and Huaxiu Yao. Analyzing and mitigating object hallucination in large vision-language models. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2
- [38] Xin Zou, Yizhou Wang, Yibo Yan, Sirui Huang, Kening Zheng, Junkai Chen, Chang Tang, and Xuming Hu. Look twice before you answer: Memory-space visual retracing for hallucination mitigation in multimodal large language models. *arXiv preprint arXiv:2410.03577*, 2024. 1, 2, 5, 6, 7

Appendix

The appendix consists of the following further discussion:

- [Appendix A](#) provides information on the code release, including links to the codebases and datasets used in the project.
- [Appendix B](#) describes the dataset that we constructed containing (</UN>) and (</CN>) tokens.
- [Appendix C](#) describes the evaluation datasets and metrics.
- [Appendix D](#) provides more qualitative and quantitative results.
- [Appendix E](#) details the implementation for REVERSE.

A. Code and Model Release

The project website is available at: <https://reverse-vlm.github.io>. The code for REVERSE is released under the MIT license at https://github.com/tsunghan-wu/reverse_vlm, and builds upon the Apache 2.0-licensed codebases of LLaVA [21] and LLaVA-MORE [10].

We also release model checkpoints of REVERSE (based on LLaVA-v1.5 and LLaVA-MORE), along with a 1.3M-sample semi-synthetic dataset, at [Hugging Face](#). Both are available under the MIT license.

The dataset contains elements adapted from LLaVA [21], which is licensed under the Creative Commons Attribution 4.0 International License. Use of the dataset should also comply with [OpenAI's usage policy](#).

B. Dataset Details

In the construction of the dataset, the automatically annotated phrases include all noun phrases (multilingual support) and a set of predefined keywords relevant to the task, such as “Yes” or “No.” We maintain a skip list to prevent the model from marking common but uninformative phrases, like “in the image.” Noun phrases with their corresponding prefix prepositions are automatically extracted using Part-of-Speech (POS) tagging tools [12], and the full data curation pipeline is in [Figure B.7](#). The skip list is in [Figure B.9](#). In this work, we only annotate model response phrases rather than the questions.

To further enhance dataset quality and balance, we validate that all tokens are tagged properly and then redistribute positive and negative answers. The final image distribution of our data composition is the same as the original LLaVA-v1.5-665k dataset as shown in [Figure B.8](#).

C. Evaluation Datasets & Metrics

To evaluate REVERSE, we use the following datasets:

POPE: The Polling-based Object Probing Evaluation (POPE) [17] is a benchmark designed to assess object hallucination in vision-language models (VLMs). Unlike traditional

instruction-based evaluations, POPE employs a polling-based query method, prompting LVLMs with simple yes-or-no questions about the presence of specific objects in images. This approach converts the evaluation into a binary classification task, allowing for more stable and flexible assessment of object hallucination. Performance on POPE is measured in F1 score following LLaVA’s standard [21].

AMBER: AMBER (An LLM-free Multi-dimensional Benchmark for MLLMs Hallucination Evaluation) [30] is a comprehensive evaluation framework designed to assess hallucination phenomena in Multi-modal Large Language Models (MLLMs). Unlike previous benchmarks that often rely on human or advanced LLM evaluations, AMBER offers an automated, low-cost approach to evaluate both generative and discriminative tasks. It addresses three key dimensions of hallucination: existence, attribute, and relation.

In this work, we refer to the generative subset as AMBER-G and the discriminative subset as AMBER-D. The AMBER-D subset comprises binary Yes/No questions and is evaluated using the F1 score, consistent with the POPE benchmark. Evaluation on AMBER-G is more complex and involves multiple metrics, as described below.

CHAIR [26] measures the percentage of hallucinated objects in a scene, and is defined as:

$$CHAIR(R) = 1 - \frac{\text{len}(R'_{obj} \cap A_{obj})}{\text{len}(R'_{obj})}. \quad (\text{C.3})$$

where $A_{obj} = \{obj_1^A, obj_2^A, \dots, obj_n^A\}$ is an annotated list of objects, and $R_{obj} = \{obj_1^R, obj_2^R, \dots, obj_n^R\}$ are nouns extracted from the captions using NLTK.

Cover measures the object coverage of responses, namely, the proportion of objects mentioned in the response R'_{obj} relative to the objects identified in the A_{obj} , and is defined by:

$$Cover(R) = \frac{\text{len}(R'_{obj} \cap A_{obj})}{\text{len}(A_{obj})}. \quad (\text{C.4})$$

Hal is a more general metric, measuring the portion of responses containing hallucinations (similar to $CHAIR_s$). It is defined as:

$$Hal(R) = \begin{cases} 1 & \text{if } CHAIR(R) \neq 0, \\ 0 & \text{otherwise.} \end{cases} \quad (\text{C.5})$$

Cog measures how similar the hallucinations that a model generates are to human hallucinations. It is defined as:

$$Cog(R) = \frac{\text{len}(R'_{obj} \cap H_{obj})}{\text{len}(R'_{obj})}. \quad (\text{C.6})$$

for a set of target hallucinatory objects $H_{obj} = \{obj_1^H, obj_2^H, \dots, obj_n^H\}$.

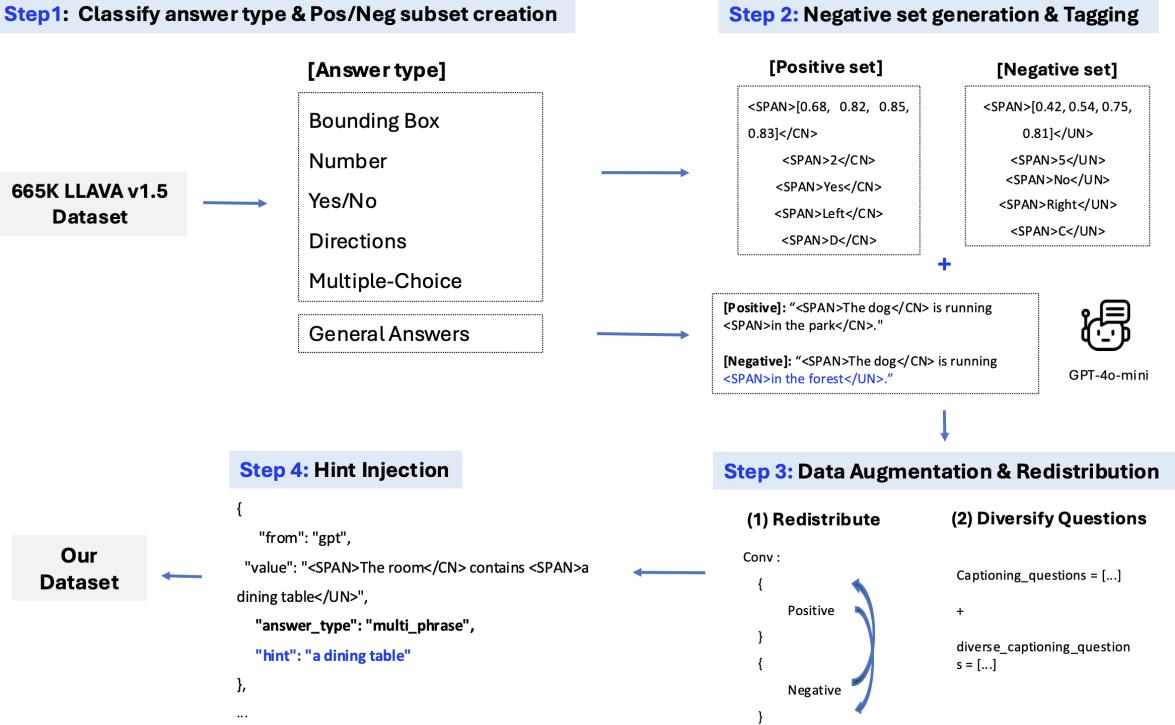


Figure B.7. The data generation pipeline for our 1.3M instruction-tuning data.

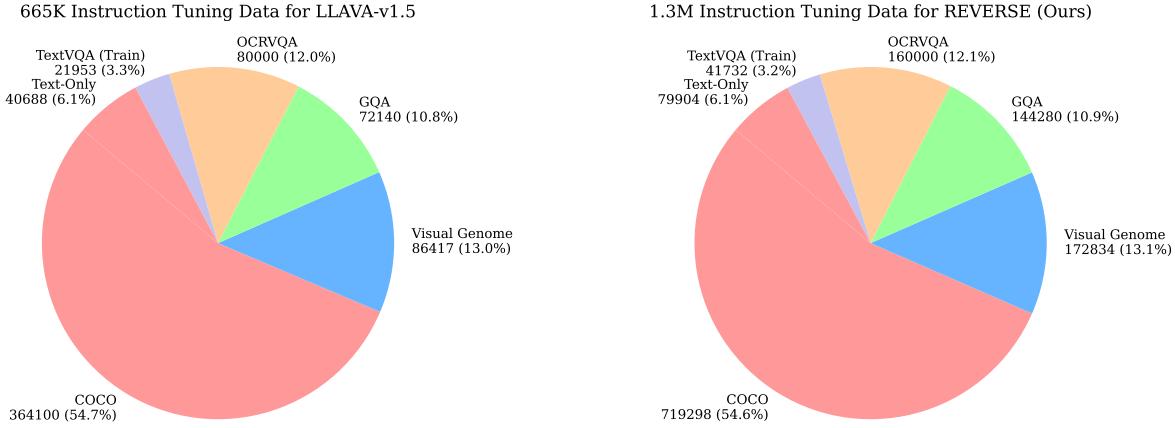


Figure B.8. Comparison between LLaVA-v1.5-665k and our 1.3M instruction tuning dataset

CHAIR-MSCOCO: MS COCO (Microsoft Common Objects in Context) [18] is a large-scale dataset designed for object detection, segmentation, and captioning tasks in computer vision. It contains over 330,000 images, with more than 200,000 labeled images spanning 80 object categories. The dataset includes detailed instance annotations, allowing for precise object localization and segmentation. Additionally, MS COCO provides five human-generated captions per image, making it a popular benchmark for image captioning and vision-language models (VLMs).

The CHAIR-MSCOCO benchmark was first introduced by Rohrbach et al. [26], which uses the full MSCOCO validation set to evaluate hallucination in vision-language models using the CHAIR score. In this work, we follow the evaluation protocol of Yue et al. [35] and assess a subset of 500 captions for efficient benchmarking.

MME-Hall: MME-Hall [11] is a specialized subset of the Multimodal Large Language Model Evaluation (MME) benchmark, focusing specifically on assessing object-related

hallucinations in multimodal large language models (MLLMs). It evaluates models across four key dimensions: object existence, counting, positional accuracy, and color recognition.

HaloQuest: HaloQuest [31] is a visual question answering (VQA) dataset designed to evaluate and mitigate hallucination in vision-language models (VLMs). The evaluation set consists of over 600 examples, featuring both real images from the OpenImages dataset and synthetic images generated using tools like Midjourney. The dataset focuses on three categories of questions: those with false premises, those lacking sufficient context, and visually challenging ones, aiming to trigger common hallucination scenarios in VLMs. Performance on Haloquest is measured using “accuracy” as evaluated by Gemini-1.0 Pro, however since this model is no longer available, we leverage Gemini 1.5-Pro in this paper.

MMHal-Bench: MMHal-Bench [28] is an evaluation benchmark specifically designed to assess hallucination phenomena in Large Multimodal Models (LMMs). It comprises 96 challenging image-question pairs sourced from the OpenImages dataset, each accompanied by corresponding ground-truth answers and detailed image content annotations. The benchmark focuses on penalizing hallucinations by evaluating model responses against these ground truths. Performance on MMHal-Bench is measured using “score” (ranging from 0-6) and “hallucination rate” (ranging from 0-1) as evaluated by a GPT model.

D. Additional Results

We present additional qualitative and quantitative results in this section. Representative qualitative examples are shown in Figure D.11. For quantitative analysis, we include raw bootstrapped evaluation results in Table D.6, Table D.7, and Table D.8. As described in section 4, we apply 100-round bootstrapping to account for variability introduced by sampling during inference. Across all captioning and open-ended VQA tasks, REVERSE consistently demonstrates strong robustness and significantly outperforms the base VLM within the 95% confidence interval.

what, where, which, who, whom, whose, why, how,
 What, Where, Which, Who, Whom, Whose, Why, How,
 that, this, these, those, That, This, These, Those,
 he, she, it, we, you, they, me, him, her, us, them, I,
 He, She, It, We, You, They, Me, Him, Her, Us, Them, I,
 my, your, his, her, its, our, their, mine, yours, ours, theirs,
 My, Your, His, Her, Its, Our, Their, Mine, Yours, Ours,
 Theirs,
 a, an, the, A, An, The,
 in the image, the image, The image, In the image, in the
 picture, the picture, The picture, In the picture.

Figure B.9. Words that we skip during data processing to ensure high-quality hallucination generation

E. Implementation Details

E.1. Training Details

For both LLaVA-v1.5-7B and LLaVA-MORE (LLaVA with Llama 3.1 as the LLM backbone), we initialize from a pretrained LLM and visual projector. We perform instruction fine-tuning using our 1.3M dataset for one epoch with LoRA (rank = 128, α = 256). Our loss function is the modified cross-entropy loss described in subsection 3.2.

We use the AdamW optimizer with a learning rate of 2e-5 for the visual projector and (2e-4, 1e-4) for the LoRA parameters of LLaVA-v1.5-7B and LLaVA-MORE, respectively. The batch size is 16, with gradient accumulation steps set to 1. Training takes approximately 24 hours for LLaVA-v1.5-7B and 36 hours for LLaVA-MORE, both conducted on eight DGX A100 80GB GPUs using DeepSpeed ZeRO-2.

E.2. Full Decoding Algorithm

The full decoding algorithm used in REVERSE is shown in 1. For captioning and discriminative tasks, we directly apply this algorithm.

For open-ended tasks such as MMHal-Bench and HaloQuest, as described in section 4, we adopt a two-stage decoding process. In the first round, REVERSE runs the standard retrospective resampling algorithm. Since many of these queries contain false premises or lack sufficient information, the model is expected to abstain from answering and return a blank response. In such cases, we initiate a second round of inference with the query modified as:

```
Q := Q + "For this question,
please point out the false
premises or note what information
is missing, rather than answering
it directly."
```

This prompting strategy requires no additional training and enables the model to handle underspecified or invalid queries more effectively.

Table D.6. Bootstrapped results on CHAIR-MSCOCO and AMBER-G. We report mean scores with 95% confidence intervals as subscripts.

Base VLM	Method	CHAIR-MSCOCO		AMBER-G		
		CHAIR _i (↓)	CHAIR _s (↓)	CHAIR(↓)	Cover(↑)	Hall(↓)
LLaVA-v1.5 7B [22]	Base VLM	15.4	50.0	7.8	51.0	36.4
	REVERSE _(τ = 0.003)	10.3 _(8.94–11.68)	37.0 _(32.90–41.50)	6.0 _(5.4–6.5)	52.2 _(51.1–53.5)	30.4 _(27.8–32.9)
	REVERSE _(τ = 0.0003)	6.1 _(4.53–7.60)	13.6 _(10.80–16.71)	4.0 _(2.3–5.9)	26.9 _(24.1–30.6)	10.2 _(6.6–14.2)
LLaVA-MORE 8B [10]	Base VLM	14.4	52.0	7.8	53.1	36.6
	REVERSE _(τ = 0.003)	12.2 _(10.55–13.81)	42.4 _(38.09–46.02)	6.5 _(5.9–7.1)	54.8 _(53.6–56.2)	35.5 _(32.4–38.9)
	REVERSE _(τ = 0.0003)	8.4 _(7.16–10.06)	25.2 _(21.39–28.60)	5.1 _(4.5–5.6)	38.9 _(37.3–40.4)	20.8 _(18.6–23.2)

Table D.7. Bootstrapped results on MMHal and HaloQuest. 95% confidence intervals are shown as subscripts.

Backbone	Method	MMHal		HaloQuest		
		Score(↑)	Hall. Rate(↓)	Avg Acc.(↑)	FP Acc.(↑)	VC Acc.(↑)
LLaVA-v1.5 7B	Base VLM	2.11	0.54	22.6	17.1	39.5
	REVERSE _(τ = 0.003)	2.56 _(2.22–3.01)	0.47 _(0.35–0.56)	30.7 _(27.4–35.2)	31.8 _(27.5–35.7)	31.5 _(25.2–38.1)
	REVERSE _(τ = 0.0003)	3.28 _(2.86–3.72)	0.30 _(0.20–0.40)	32.3 _(29.4–36.5)	29.4 _(25.1–35.7)	18.7 _(13.5–24.6)
LLaVA-MORE 8B	Base VLM	2.50	0.53	22.4	15.8	43.4
	REVERSE _(τ = 0.003)	2.28 _(1.96–2.68)	0.54 _(0.44–0.62)	26.7 _(23.4–30.0)	30.0 _(26.4–35.2)	31.3 _(25.3–38.2)
	REVERSE _(τ = 0.0003)	2.93 _(2.53–3.22)	0.40 _(0.31–0.51)	36.7 _(33.9–39.8)	39.5 _(34.8–45.0)	30.9 _(26.6–37.2)

Table D.8. Bootstrapped results on discriminative hallucination benchmarks. We report means with 95% confidence intervals.

Backbone	Method	AMBER-D	POPE	MME-Hall
LLaVA-v1.5 7B	Base VLM	74.7	85.9	648.3
	REVERSE	74.2 _(73.4–75.2)	85.9 _(82.8–88.5)	601.60 _(555.00–642.54)
LLaVA-MORE	Base VLM	71.6	85.1	678.3
	REVERSE	69.3 _(68.5–70.3)	84.4 _(81.8–86.9)	657.63 _(615.71–696.83)

Algorithm 1 On-the-Fly Retrospective Resampling During Generation

Require: Input prompt $\{I, Q\}$ (image and question), maximum total corrections N , local correction threshold K , base temperature T_0 , temperature step $\Delta T = 0.05$

- 1: **Initialize:** Attempt count $n \leftarrow 0$, local failures $k \leftarrow 0$, temperature $T \leftarrow T_0$, placeholder list $P \leftarrow \emptyset$
- 2: Initialize empty sequence S
- 3: **while** generation is not finished **do**
- 4: Generate next token: $w_t = \text{VLM}_\theta(\{I, Q\} + S, T)$
- 5: Append w_t to S
- 6: **if** $P(w_t) \geq \tau$ **then** ▷ Hallucination detected
- 7: Identify most recent $</\text{CN}>$ as local checkpoint C_{local}
- 8: Append hallucinated phrase w_t to placeholder list P
- 9: **while** $n < N$ **do** ▷ Inside the correction loop
- 10: Backtrack to C_{local} and apply **Rejection Sampling and Query Rewriting**
- 11: Update temperature: $T \leftarrow \min(T + \Delta T, T_0 + 0.5)$
- 12: Modify prompt with clarification hint:

$$Q = Q + (\text{Hint: potential incorrect phrases} \rightarrow P)$$
- 13: Generate resampled phrase: $h_t = \text{VLM}_\theta(\{I, Q\} + S, T)$
- 14: **if** $P(h_t) < \tau$ for all tokens in h_t **then** ▷ Verify all token probabilities
- 15: Accept h_t and continue generation
- 16: Append h_t to S
- 17: **Reset temperature:** $T \leftarrow T_0$, **Reset failure count:** $k \leftarrow 0$
- 18: **Break** out of correction loop
- 19: **else** ▷ Track consecutive failures
- 20: $k \leftarrow k + 1$
- 21: **end if**
- 22: $n \leftarrow n + 1$
- 23: **if** $k \geq K$ **then** ▷ Escalate backtracking if local corrections fail
- 24: Identify last sentence boundary (last punctuation token) as C_{global}
- 25: Backtrack to C_{global} and reset $k \leftarrow 0$
- 26: **end if**
- 27: **end while**
- 28: **end if**
- 29: **end while**
- 30: **Return** generated sequence S

Given one provided question-answer pair, please select one of the tagged segments (...) in the "answer" and replace it with an alternative that captures a similar aspect but differs in meaning (e.g., varying noun phrases, numbers, keywords, etc.) appropriate for a typical context to create hard, incorrect negative samples. The substitution should }

Use a word that differs visually distinct from the original word }
}

- Belongs to the same superclass or broader category as the original answer (e.g. - both animals/fruits etc.) but the meaning should change significantly.)
- Don't change the first word if that already appears in the question }
Please explain the reasoning first and then return the original word/phrase and the substituted one.)
}

Provide your answer in the following JSON format:

```
{ Reasoning: Provide an explanation of why a specific word or phrase was chosen for substitution and the rationale behind the chosen alternative.,
Output: [Original Text, Alternative] }
```

Here are good examples of the task

```
{ Reasoning: In the provided answer, the phrase '<SPAN>A red plastic cup</SPAN>' describes a specific type of object (a cup) that is used in a context (likely related to beverages), and it is paired with another object ('a clear straw'). To create a hard, incorrect negative sample, I chose to substitute this phrase with '<SPAN>A green glass bottle</SPAN>', which refers to a different type of container while maintaining the overall theme of objects associated with drinks. However, it alters the context sufficiently to be incorrect as an answer to the original question about the region's description.,
Output: [ A red plastic cup, A green glass bottle ] }
```

{ Reasoning: The original phrase 'giant hotdog' is specific and unusual, which makes it memorable. By replacing it with 'small burger,' I create a phrase that is similarly nonsensical in this context but alters its meaning. 'Small burger' retains a food-related theme, making it seem plausible while still not fitting the context of a region description. Additionally, 'the man's mouth' was kept intact to maintain a semblance of continuity in the sentence structure.,
Output: [giant hotdog, small burger] }

{ Reasoning: The original answer identifies Sinclair Lewis as the author, which is accurate. For the negative sample, I replaced 'Sinclair Lewis' with 'Mark Twain,' another well-known author. This substitution maintains the aspect of being a famous author but is incorrect in the context of the question about the specific book mentioned.,
Output: [Sinclair Lewis, Mark Twain] }

{ Reasoning: The original phrase 'the back view' is chosen for substitution because it describes a specific angle or perspective of the subject (an adult giraffe). To create a hard, incorrect negative sample, I replaced it with 'the frontal view', which refers to a different perspective entirely. This alteration maintains the structure of the sentence but changes the meaning significantly, making it incorrect in the context.,
Output: [the back view, the frontal view] }

Here's the input:

- Question {question}
- Answer: {answer}

Figure B.10. The prompt used for negative set data generation for general answer type answers. We change the key noun phrases, tag them with special </UN> tokens.

 Describe this image.	LLaVA	OPERA	Woodpecker	REVERSE
	The image features a small orange cat walking down a cobblestone street next to a parked motorcycle. The cat appears to be curiously looking at the motorcycle, possibly intrigued by its presence. The motorcycle is positioned on the left side of the scene, with the cat walking towards it. The street is lined with cobblestones, giving it a charming and old-world atmosphere.	The image features a small orange cat walking down a cobblestone street next to a parked motorcycle. The cat appears to be exploring the area, possibly looking for food. The motorcycle is positioned on the right side of the scene, with its front wheel visible. In addition to the cat and the motorcycle, there are two people in the scene. One person is standing near the left edge of the image, while the other person is located further back on the right side. A potted plant can be seen on the right edge of the image, adding a touch of greenery to the scene.	The image features a cat([0,147, 0.63, 0.316, 0.974]) walking down a cobblestone street([0,165, 0.442, 1.0, 0.999]), with a motorcycle([0.515, 0.083, 0.757, 0.756]) parked nearby. The cat appears to be curiously looking at the motorcycle. The motorcycle is positioned on the side of the path, and the cat seems to be approaching it. The scene takes place in a city setting, with a brick wall visible in the background.	The image features a small orange cat walking along a stone path near a red motorcycle. The cat appears to be curious and exploring its surroundings. The motorcycle is parked on the side of the path, and the cat seems to be approaching it. The scene takes place in a city setting, with a brick wall visible in the background.
	The image features a woman wearing a helmet and riding a bicycle on a road. She is stopped next to a fence, where two horses are standing. The woman appears to be petting one of the horses, creating a friendly interaction between her and the animals. There are several backpacks placed around the scene, with one near the woman on the bicycle, another on the ground close to the horses, and a third one further away. The presence of these backpacks suggests that the woman might be on a journey or a leisurely ride, taking a break to interact with the horses.	The image features a woman wearing a helmet and riding a bicycle on a road. She is stopped next to a fence, where two horses are grazing. The woman appears to be petting one of the horses while she is on her bike. In addition to the woman and the horses, there are two other people in the scene, one standing near the fence and another person further away. A backpack can be seen placed on the ground near the woman on the bicycle. The scene captures a moment of interaction between the woman and the horses, creating a peaceful and serene atmosphere.	The image features a woman([0,497, 0.291, 0.738, 0.961]) wearing a helmet and riding a bicycle([0,457, 0.558, 0.897, 0.92]) on a road([0,69, 0.409, 0.999, 0.998]). She is stopped next to a fence([0,002, 0.552, 0.514, 0.993]), where two horses([0,0, 0.352, 0.506, 0.991];[0,002, 0.555, 0.356, 0.981]) are standing. The woman is petting one of the horses. She is also holding a bag. A dog is near the woman. There are several backpacks placed around the scene, with one near the woman on the bicycle([0,781, 0.602, 0.895, 0.799]), another on the ground close to the horses([0,534, 0.703, 0.613, 0.884]), and a third one further away.	The image features a woman on a bicycle, standing near a fence in a field. She is interacting with a horse, which is standing near her. The horse appears to be curious about the woman and her bicycle. The woman is wearing a helmet. In the background, there are two other horses in the field, one of which is closer to the woman and the other further away. The woman seems to be enjoying her time with the horses, and the scene appears to be a peaceful moment in the countryside.
	The image features a dog running across a sandy beach, enjoying the open space and the sunny day. The dog is positioned in the center of the scene, with its body facing the camera. The beach is lined with a few trees, providing a natural backdrop for the scene. The dog appears to be running towards the water, possibly to cool off or explore the shoreline.	The image features a dog running across a sandy beach, enjoying the open space and the feeling of freedom. The dog is positioned towards the left side of the scene, with its body facing towards the right. The beach is surrounded by a body of water, which can be seen in various parts of the image, from near the left edge to the right side. The scene captures the dog's energetic and playful nature as nobody else is visible in the image.	The image features a dog([0,345, 0.369, 0.475, 0.565]) running across a sandy beach([0,002, 0.377, 0.998, 0.997]), enjoying the open space and the sunny day. The beach is lined with a few trees. The dog appears to be running towards the water([0,002, 0.295, 0.998, 0.449]), possibly to cool off or explore.	The image features a dog running on a sandy beach near the water. The dog is in the center of the scene, with its shadow visible on the sand in front of it. The dog appears to be enjoying its time on the beach, playing and running along the shoreline. The beach is a beautiful place, with a clear blue sky and a calm ocean in the background.
	The image features a man sitting at a desk, writing on a piece of paper with a pen. He is wearing a gray shirt and appears to be focused on his task. A cup, possibly a coffee mug, is placed on the desk in front of him, adding to the casual atmosphere of the scene. The man is seated in a chair, which is positioned near the desk.	The image features a man sitting at a desk, writing on a piece of paper with a pen. He is wearing a gray shirt and appears to be focused on his task. Next to him, there is a coffee mug with a cartoon character on it, adding a touch of humor to the scene. The desk is also equipped with a keyboard and a mouse, suggesting that the man might be working on a computer. A book can be seen on the desk, possibly providing reference material or inspiration for his writing.	The image features a man([0,184, 0,001, 0.999, 0.809]) sitting at a desk([0,006, 0.766, 1.0, 0.999]), writing on a piece of paper with a pen([0,417, 0.5, 0.578, 0.807]). Next to him, there is a cup([0,083, 0.419, 0.35, 0.848]) with a cartoon character on it. The man is also holding a book([0,397, 0.634, 0.999, 0.834]), which might be related to his writing or research. The scene suggests a casual and comfortable workspace.	The image features a man sitting at a desk, wearing a gray shirt.
	The image features a group of three people riding bicycles on a dirt path near the ocean. They are enjoying a leisurely ride along the coastline, taking in the beautiful scenery. The first person is riding a bicycle on the left side of the path, while the second person is positioned in the middle, and the third person is on the right side. The ocean is visible in the background, providing a serene and picturesque setting for the cyclists. The dirt path they are riding on is surrounded by grass, adding to the natural and peaceful atmosphere of the scene.	The image depicts a group of three children riding bicycles on a dirt road near the ocean. They are enjoying their time outdoors, with one child riding a bicycle in the foreground, another in the middle, and the third one further back. The children are wearing helmets for safety while they ride. The scene is set against a beautiful backdrop of the ocean, with the sky visible in the background. The children seem to be having a great time exploring the area on their bicycles.	The image features a group of three people, two children and an adult, riding bicycles along a path near the ocean. The image features a group of three people, including one child([0,714, 0.542, 0.778, 0.835]) and two adults([0,693, 0.507, 0.724, 0.594];[0,692, 0.504, 0.725, 0.595]), riding bicycles([0,704, 0.665, 0.806, 0.881];[0,7, 0.557, 0.719, 0.611]) along a path near the ocean([0,001, 0.476, 0.655, 0.785]). They are enjoying a leisurely ride on a sunny day, with the ocean in the background. The children are riding bikes, while the adults are walking alongside them.	The image features a group of people, including two children, riding bicycles along a dirt path near the ocean. The path is surrounded by grass and a few trees, providing a scenic backdrop for the group. The children are wearing helmets, indicating their safety while enjoying their ride. The scene captures a moment of leisure and outdoor activity for the group.

Figure D.11. Additional Qualitative Results.