Highlights

Sarah: Hallucination Detection for Large Vision Language Models with Semantic Information Locator and Purifier in Uncertainty Quantification Method

Fang Yue, Yang Zhang, Yawen Liu, Yetian Yu

- We propose a hallucination detection method grounded in uncertainty quantification, which achieves performance comparable to latest related methods while being significantly more cost-effective, as it avoids multiple inference rounds and complex external tools.

- We introduce two novel, orthogonal modules designed to address the uneven distribution of semantic information. By operating independently—one dedicated to semantic cooperation strengthening and the other to semantic interference mitigation—they work in a complementary fashion to maximize overall effectiveness.

- The versatile framework is applicable to various large vision language models and generation tasks.

Sarah: Hallucination Detection for Large Vision Language Models with Semantic Information Locator and Purifier in Uncertainty Quantification Method

Fang Yue[a,b], Yang Zhang[a,*], Yawen Liu[a], Yetian Yu[a]

[a]*School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, 100044, China*
[b]*School of Cyber Science and Engineering, Southeast University, Nanjing, 210096, China*

Abstract

Large Vision-Language Models (LVLMs) have demonstrated remarkable potential in multi-modal applications, yet their reliability is compromised by hallucination – misalignment between generated text and visual inputs, linguistic context, or factual knowledge. Existing detection methods often rely on multi-round inference or external tools, limiting their efficiency and generalization. To address this, we propose Sarah (Hallucination Detection for Large Vision Language Models with Semantic Information Locator and Purifier in Uncertainty Quantification Method), a novel hallucination detection framework based on uncertainty quantification. Sarah requires only single-round inference and keeps its core mechanism self-contained, eliminating reliance on external tools in critical steps. The framework introduces two orthogonal modules: the Semantic Information Locator, which dynamically weights token importance through constrained perturbation, and the Semantic Information Purifier, which disentangles semantic and expression uncertainty via lexical clustering. Extensive experiments on five off-the-shelf LVLMs and three open-ended and closed-ended benchmarks show that Sarah outperforms most latest related methods, achieving competitive accuracy with the best method while being 10× faster. Analysis over LVLMs further exposes critical limitations: over 13.4% of state-of-the-art LVLM outputs contain hallucinations. Our work provides a efficient, scalable, and interpretable solution for hallucination detection and model diagnosis, which leaves room for future improvements.

Keywords: Hallucination Detection, Large Vision Language Model, Semantic Information, Uncertainty Quantification, Single-round Inference

1 Introduction

Large Vision-Language Models (LVLMs) have represent a significant advancement in multi-modal AI systems, demonstrating the ability to process and integrate visual and textual data for solving complex tasks that require joint understanding of both modalities [1, 2, 3, 4]. Despite their impressive capabilities, these models are prone to hallucination —— a critical issue where generated content deviates from factual accuracy or contradicts the provided visual and textual context [2, 5]. This phenomenon primarily stems from limitations such as constrained knowledge boundaries, failures in knowledge recall, and reliance on outdated pre-training corpora [2, 6]. Addressing hallucination detection, which involves verifying the factual consistency and accuracy of each information unit in generated outputs relative to the input image and text, presents three major challenges. First, LVLM outputs typically interweave correct and hallucinated information [7], transforming the detection task into a complex, fine-grained analysis rather than a simple binary classification. Second, exhaustive validation against external references proves computationally intensive and resource-prohibitive. Third, existing evaluation metrics often produce opaque, single-dimensional scores, offering limited interpretability regarding the specific nature and location of errors.

*Corresponding author. Email: zhang.yang@bjtu.edu.cn

To address these challenges, we present Sarah, a novel uncertainty-based framework for LVLM hallucination detection. Our approach introduces four key innovations(see Figure1): (1)Independent Claim Extraction: The framework decomposes the generated content into discrete claims, where each claim constitutes a concise statement representing an individual unit of information[8, 9, 10];(2)Semantic Information Processing: Through our Semantic Information Locator and Purifier modules, we enhance semantic coherence while mitigating interference. The locator identifies semantically significant words within the global context of the output, while the purifier processes token-level distributions, aggregates semantically equivalent tokens, and eliminates uncertainties arising from linguistic variations; (3)Uncertainty Quantification: Each extracted claim undergoes rigorous validation through our uncertainty quantification method; (4)Decision Mechanism: Our Multi-Thresholds Decision Mechanism focuses exclusively on high-uncertainty tokens, effectively minimizing interference from low-hallucination content.

The proposed detection framework offers significant advantages: it precisely identifies the specific components that contribute to hallucination, and provides a mathematical approach for uncertainty quantification that requires neither external knowledge nor model modifications. Comprehensive evaluations across three established vision-language benchmarks demonstrate that Sarah achieves strong alignment with both human assessments and our specialized automated labeling system[11, 12]. Our results show substantial improvements in hallucination detection for LVLM-generated content. Furthermore, our comprehensive analysis using Sarah reveals critical insights into the reliability of various LVLMs, with GPT-4o emerging as the top performer among tested models, particularly in image captioning tasks. These findings not only corroborate human evaluations but also align with previous research, establishing Sarah as a robust tool for model assessment and improvement. Our contributions can be summarized as follows:

- We propose Sarah, a single-round, self-contained uncertainty quantification framework for hallucination detection in LVLMs, which operates without external tools in its core steps, enhancing robustness and generalization.
- We design two novel, orthogonal modules—the Semantic Information Locator and Purifier—that jointly refine token importance and disambiguate uncertainty types, enabling fine-grained detection without interference.
- Our method consistently achieves leading performance across five LVLMs and three benchmarks, surpassing most competitive baselines while significantly reducing inference time.
- We provide model-agnostic diagnostic insights, revealing that over 13.4% of outputs from leading LVLMs contain hallucinations, highlighting critical reliability gaps.

2 Related Work

2.1 Large Vision Language Models

Large Vision-Language Models (LVLMs) have emerged as a transformative paradigm in multimodal AI, integrating three core architectural components: a text encoder, an image encoder, and a cross-modal alignment module [13, 14]. Recent advancements in LVLMs have significantly pushed the boundaries of performance across a diverse range of vision-language (VL) tasks, including autonomous driving, embodied robotics, and medical diagnosis [15, 16, 17, 18].

Notable contributions in this domain include LLaVA [19], which bridges visual and textual modalities by projecting visual encoder outputs into the input space of the LLaMA language model [20] and training on synthetic multimodal data[21]. mPLUG-Owl [22] further advances multimodal capabilities through a multitask learning framework, enhancing cross-modal understanding. GPT-4o [23] achieves finer-grained image comprehension by refining its visual encoder and optimizing the alignment of visual features with its language model via an enhanced adapter. Most recently, Llama-v3.2-vision[24] has set new benchmarks in generalization and complex VL task performance by incorporating more efficient visual attention mechanisms and leveraging a larger-scale, diverse dataset.

Despite their remarkable capabilities, LVLMs inherit the hallucination issue prevalent in Large Language Models (LLMs), posing significant challenges to AI safety and reliability. To address this critical limitation, we propose an intrinsic hallucination detection method designed to enhance the

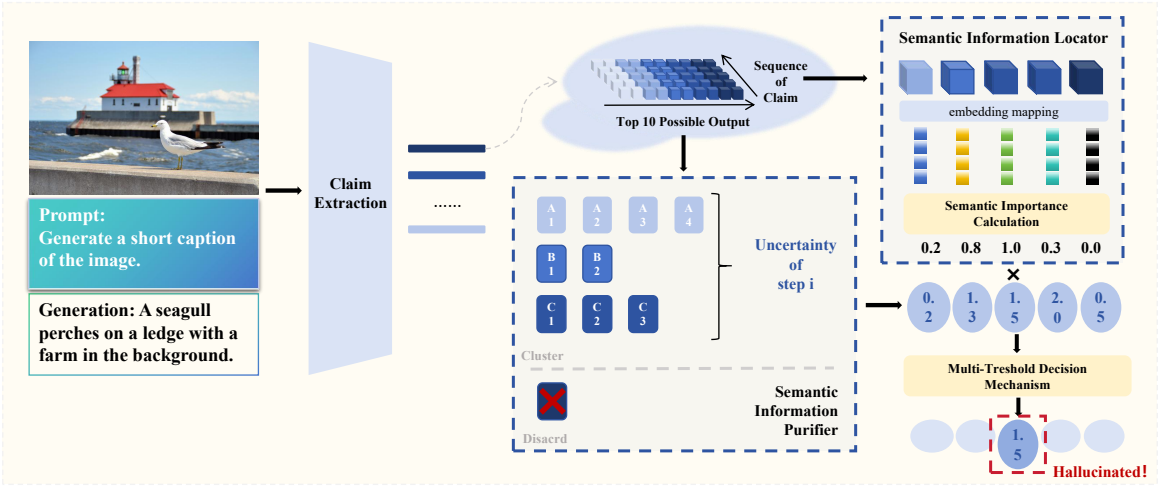


Figure 1. Overview of the Sarah framework for hallucination detection in LVLMs. The framework consists of four key stages: (1)Independent Claim Extraction, where atomic claims are decomposed from LVLM-generated responses for granular analysis; (2)Semantic Information Locator, which computes attention-based weights for each token to guide probability recalibration; (3)Semantic Information Purifier, where token-level probabilities are refined using part-of-speech analysis to enhance detection robustness; and (4)Multi-Threshold Decision Mechanism, which combines probabilistic and semantic criteria to classify hallucinated content. Sarah provides a detailed, interpretable, and scalable approach for evaluating hallucination severity in free-form LVLM outputs.

safety and trustworthiness of human-LVLM interactions.

2.2 Hallucination on LVLMs

Hallucination in LVLMs refers to the generation of content that is factually inconsistent with the provided visual and textual context. This phenomenon arises from various factors, including the presence of noisy or incomplete training data [2, 25, 26], suboptimal attention allocation during training [27], imperfect reasoning strategies, or inadequate decoding prompts [2]. While several methods have been proposed to detect and evaluate hallucinations, existing approaches are often limited to specific types of hallucination or constrained answer formats, failing to capture the full spectrum of hallucination phenomena. Moreover, current hallucination detection techniques frequently achieve high accuracy at the cost of significant resource consumption: they heavily rely on external tools and often require multi-round inference.

For instance, CHAIR [28] quantifies object hallucination by comparing generated captions with ground-truth objects in images using binary "yes-or-no" questions. Building on this, [29] introduced POPE, a polling-based query technique for object hallucination detection. More recently, researches [7, 10, 30] have expanded the scope of hallucination evaluation beyond single-instruction formats and human-annotated ground truth, leveraging external tools such as advanced models to quantify hallucinations in free-form outputs.

In contrast to these approaches, our work focuses on reference-free hallucination detection method for LVLMs, eliminating the need for external tools or multi-round inference. Our method is designed to handle open-ended visual question-answering settings, where answers are free-form and potentially lengthy, addressing a broader range of hallucination types.

2.3 Uncertainty-Based Reference-Free Methods for Hallucination Detection

Reference-free hallucination detection methods, which rely solely on internal data of the model, have gained significant traction in LLM research [31]. Among these, uncertainty-based methods [8, 32, 31, 33, 34] are particularly promising, operating on the hypothesis that generation with high uncertainty is more likely to be hallucinated [35]. These methods have been extensively studied in LLMs and are increasingly being adapted to

LVLMs.

Existing uncertainty-based approaches can be broadly categorized into three paradigms: (1) consistency-based methods, which sample multiple responses to the same input and evaluate the consistency of factual statements [36, 37]; (2)input perturbation methods, which assess prediction variance by perturbing the original input [34, 38]; and (3)Bayesian methods, which quantify uncertainty by feeding the same input multiple times into stochastic networks with dynamic weights [39, 40, 41].

While these methods often require multiple rounds of model inference, we argue that a single round of output contains sufficient information for uncertainty quantification. Our work focuses on analyzing the probability distribution of tokens within a single-round output, significantly reducing computational overhead and API call requirements while maintaining robust hallucination detection capabilities.

3 Method

The following sections present Sarah in detail. Section 3.1 introduces the Independent Claim Extraction process, Sections 3.2 and Section 3.3 describe the Semantic Information Locator and Semantic Information Purifier respectively, and Section 3.4 explains how hallucination is quantified and detected.

3.1 Independent claim extraction

LVLM-generated text often consists of complex, lengthy sentences, making direct evaluation of hallucination prone to error omission. To address this, the independent claim extraction process decomposes the generation into independent claims (basic units that convey complete meanings), using specially designed prompts on GPT-4. This granular decomposition enables precise identification of hallucinated information.

3.2 Semantic Information Locator

Semantic Information Locator leverages semantic perturbation and sentence similarity calculation to produce a semantic importance indicator for the current output, which directly identifies the specific positions that demand higher level of focus.

In practice, the locator perturbs z_i in the $i-th$ generation step ($1 \leq i \leq T$) multiple times using substitute words from the Universal Dependent Corpus [42] that are syntactically and morphologically compatible with the original tokens. Specifically, perturbation considers the consistency between the substitute word and the original word in the following three aspects:

- Part-of-Speech: Part-of-Speech ensure that replacement words share the same grammatical category as the original token.
- Morphological Features: Morphological Features capture fine-grained linguistic properties, such as tense and aspect for verbs, number for nouns, and degree for adjectives, ensuring syntactic consistency.
- Syntactic Roles: Syntactic Roles represent the token's function in the sentence, allowing prioritization of substitutions that naturally occur in the same syntactic context and reducing semantic drift.

These three constraints are combined to construct candidate sets for each token to be perturbed. A word is randomly selected from candidates whose embedding similarity exceeds a certain threshold. If no such candidate exists, the selection is returned to matching syntactic roles and morphological features only. Perturbations can be expressed as:

$$T_{z_{i,j}} = \varnothing(T, \tau_{i,j}), \tag{1}$$

where $T_{z_{i,j}}$ represents the result after applying the $j-th$ perturbation to the $i-th$ token of the initial output T, $j = 1, 2, \ldots, N$, and N is the number of perturbation times. $\varnothing$ refers to the random sampling process. $\tau_{i,j}$ denotes the $j-th$ substitute word to token z_i. The perturbed predictions are then mapped into semantic space and aggregated into a unified set:

$$\{\langle \Gamma(T_{z_{i,0}}), ..., \Gamma(T_{z_{i,j}}), ..., \Gamma(T_{z_{i,N}})\rangle | j = 1, 2, ..., N\}, \tag{2}$$

where $\Gamma()$ denotes semantic projection that follows the implementation of BertScore [15]. Semantic importance of token z_i in the independent claim c is defined as:

$$SI(c, z_i) = 1 - \frac{1}{M}\sum_{j=1}^{M} \cos\langle \Gamma(T_{z_{i,j}}),\ \Gamma(T_{z_{i,0}})\rangle, \tag{3}$$

where $cos < ., . >$ denotes the cosine similarity calculation. SI lies within the range of [0, 1], with higher values corresponding to a greater semantic impact.

3.3 Semantic Information Purifier

LVLMs often generate lexically diverse but semantically equivalent predictions, treating such variations as distinct entities might overestimate uncertainty. To address this issue, the Semantic Information Purifier employs a decision-tree framework to classify the uncertainty into two types:

- Semantic uncertainty, which reflects discrepancies in conveyed information and may lead to hallucinations
- Expression uncertainty, which pertains to variations in language style or word order and does not make produced text less factual.

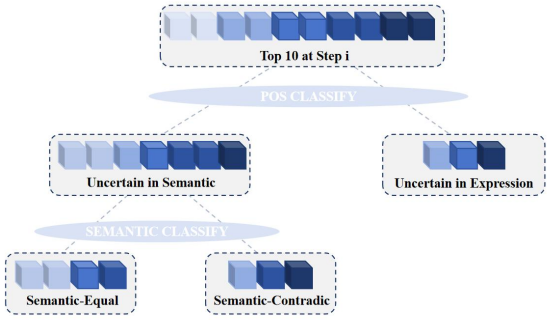


Figure 2. The decision tree of the classification procedure.

As shown in Figure 2, the procedure of classification includes two steps. First, the purifier employ DistilBERT-base model and train it for 500 epochs on a curated dataset from WordNet, with words as inputs and their part-of-speech tags as outputs. The relationships among the top-k candidate words ($1 \leq k \leq K$) at each generation step is then classified according to their part-of-speech. If the two words in a pair $(word_1, word_2)$ belong to different POS, they are categorized as expression uncertainty. Conversely, if the two words share the same POS, the uncertainty is considered to arise from differences in semantic meaning and is further divided into two categories: semantic-equal and semantic-contradict. The classification of semantic equivalence is determined based on the path distance between two concepts in the lexical network and their depth relative to the nearest common ancestor:

$$\text{sim}(w_1, w_2) = \frac{2 \times \text{depth}(\text{LCS}(w_1, w_2))}{\text{depth}(w_1) + \text{depth}(w_2)}, \tag{4}$$

$$\text{rel}(z_{i,k}, z_{i,k+1}) = \begin{cases} \text{'sem-equal'}, & sim \geq \tau_{sem}, \\ \text{'sem-contradict'}, & \text{else.} \end{cases} \tag{5}$$

Here, w_1 and w_2 indicate different words. LCS denotes the lowest common subsumer of the two words in the WordNet lexical semantic hierarchy. $depth(.)$ measures the distance from a node to the root node in the lexical semantic hierarchy. And τ_{sem} is a predefined reliability threshold.

Consequently, the word uncertain in expression is discarded by setting their probabilities to zero:

$$p(z_{i,k} \mid z_{i,k} \text{ is expression uncertainty}, s_{<i}, x) = 0. \tag{6}$$

Words with identical meaning are grouped by aggregating their probabilities into semantic clusters $\mathbf{e}_{i,m}$ at position i, where $m = 1, \ldots, M_e$ and M_e denotes the total cluster count. In this way various surface forms with a similar meaning are mapped to a single categorical variable. As the predictive entropy computation formula for token z_i applied to original uncertainty quantification is:

$$U_{z_i} = -\sum_{k=1}^{K} p(z_{i,k} \mid s_{<i}, x) \log p(z_{i,k} \mid s_{<i}, x), \tag{7}$$

the purified uncertainty of token z_i can be written as follows:

$$U_{\text{Purified}}(c, z_i) = -\sum_{m=1}^{M_e} p(\mathbf{e}_{i,m}) \log p(\mathbf{e}_{i,m}). \tag{8}$$

This helps to purify the probability distribution at each generation step, making it more conducive to hallucination quantification.

3.4 Hallucination Quantification and Detection

Sarah outputs hallucination score with a continuous distribution. Since the value produced by the locator and the purifier are orthogonal, the hallucination score for each token z_i in claim c is obtained by taking the product of semantic importance (SI) and purified uncertainty (U_{Purified}) proposed in Section 3.3:

$$\begin{aligned} \mathcal{S}_{z_i} &= \alpha \cdot (1-\alpha)\, U_{\text{Purified}}(c, z_i) \cdot SI(c, z_i) \\ &= \alpha \cdot (1-\alpha)\, \left|U_{\text{Purified}}(c, z_i)\right| \cdot \left|SI(c, z_i)\right|, \end{aligned} \tag{9}$$

here the adjustable variable serves as a trick that transforms Sarah into a flexible, adaptive evaluation method, and can be adjusted to change the level of involvement of different modules.

Inspired by the observation that low-hallucination tokens introduce noise for hallucination quantification while high-hallucination tokens dominate the overall judgment, A Multi-Threshold Decision Mechanism is proposed. Threshold 1 identifies highly hallucinatory tokens at each step:

$$\text{hallucination}(c, z_i) = \begin{cases} \text{'high'}, & \text{score}(z_i) > \text{thres1}, \\ \text{'low'}, & \text{otherwise.} \end{cases} \tag{10}$$

Threshold 2 determines whether a claim contains hallucination based on the sum of scores from highly hallucinatory tokens. If the sum exceeds threshold 2, the claim is marked as hallucinated; otherwise, it is considered non-hallucinated:

$$\text{score}(c) = \begin{cases} 1 & \text{if sum(high hallucinatory)} > \text{thres2}, \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

The sentence-level hallucination score for the entire output is the sum of scores across all claims. Finally, the Youden Index is employed to establish an optimal threshold for detecting whether hallucination exists in the generation, defined as:

$$\text{Youden Index} = TPR + TNR - 1. \tag{10}$$

The index ranges from 0 to 1, where a higher value indicates better discriminative ability. Threshold 3 is selected by maximizing Youden Index across all possible classification cutoffs, ensuring an equilibrium between correctly identifying true positives and minimizing false positives.This mechanism ensures robust and interpretable hallucination detection while minimizing the impact of low-severity errors.

4 Experiment

Comparative Methods. Given that both LVLMs and LLMs are inherently free-form generative models, we compare against six up-to-date hallucination detection methods designed for either LLMs or LVLMs, including Faithscore, GAVIE, InterrogateLLM[43], VL-Uncertainty, KLE[44], Semantic Density[45] and Semantic Entropy.

LVLMs. We conduct experiments over 5 popular "off-the-shelf" LVLMs, including GPT-4o ("o" for "omni"), LLaMA-3.2-Vision-Instruct-11B, LLaVA-1.5 with model size 7B and 13B, and mPLUG-Owl3-7B.

Benchmarks. For the open-ended benchmarks, We consider 2 open-ended question answering datasets to assess the performance of existing LVLMs: (1) Bingo: This dataset is designed to evaluate and shed light on two common types of hallucinations in LVLMs: bias and interference. Each image in Bingo is paired with one or two questions; (2) MSCOCO-Caption: This dataset is simply designed for image captioning task. Images are sampled from the MSCOCO validation set. Besides, to standardize the output format of each model and ensure the comparison results are authoritative, models are required to output only two sentences uniformly. We also consider a general VQA benchmark to evaluate the generalization ability of our method. Therefore, we introduce VQA v2, a benchmark dataset for visual question answering that requires models to infer correct answers from images and corresponding natural language questions, emphasizing both visual understanding and linguistic reasoning.

Open-ended benchmarks generally have higher variability than closed-ended question, which is hard to get ground truth answers before model inference [46]. Considering that, we use GPT-4o with specifically designed prompt to annotate the hallucinations, see Table1. Besides, we hire some human labelers to annotate the hallucinations as well. They were paid 0.15$ per HIT, which is more than prevailing local minimum wage. Human evaluation verifies that our constructed benchmark based on GPT-4o can adequately evaluate the performance of the hallucination evaluation methods, see Table 2. Such an approach enables completely automatic evaluation and allows us to scale up our experiments.

Implementation Details. For all LVLMs besides GPT-4o, we extract probabilities of top-10 likely output for token. For GPT-4o, we generate top-5 likely output probabilities, as the top k most likely outputs account for the majority of possibility, see Figure3. The max sequence length of LVLMs output is set to 1024, and the temperature t is set to 0 for our evaluation work. If the last sentence of the generation is unfinished (i.e. does not end with any punctuation), it is discarded. All the others are set as default. All the experi-

Table 1
Prompt for ground truth generation on GPT-4.

Imagine you are an intelligent teacher. Thoroughly read the instruction: {instruction}, reference answer: {ground_truth} and the prediction answer: {text} to ensure a clear understanding of the information provided. Assess the correctness of the predictions. If the prediction answer does not conflict with the reference answer as well as the real world facts, please generate "correct". If the prediction answer conflict with the reference answer, please generate "incorrect". The output should only be "correct" or "incorrect". **Example:** **Question:** "The two lines are parallel to each other. Why?" **Reference answer:** "The two straight lines in the picture are parallel, because the slopes of two straight lines are equal." **Prediction answer:** "The two lines will never intersect." **Output:** "correct"

Table 2
Automatic evaluation is highly accurate as compared to human evaluation on mPLUG-Owls3.

Dataset	Accuracy
Bingo	85.6
MSCOCO-Caption	83.2
VQA v2	88.3

ments are conducted on a server with one Intel(R) Xeon(R) Platinum 8352V CPU and four NVIDIA 4090 GPUs (python version: 3.10.8) .

4.1 Comparison with Latest Related Methods

Experimental result. We evaluate Sarah against several state-of-the-art (SOTA) hallucination detection methods on open-ended and closed-ended benchmarks, with results summarized in Table 3. The detection performance of our method is competitive with InterrogateLLM and significantly outperforms other comparative methods. In direct comparison with InterrogateLLM on open-ended tasks, Sarah surpasses InterrogateLLM in three scenarios while falling slightly behind (by no more than 1.6%) in other cases. This marginal disadvantage is considered acceptable. Notably, although Sarah exhibits slightly inferior detection accuracy compared to InterrogateLLM, it demonstrates substantial advantages in both time and computational resource consumption. As shown in Table 6, Sarah requires only $1/10^1$ of the detection time needed by InterrogateLLM. This efficiency stems from Sarah's single-round inference requirement. Furthermore, the unique hallucination quantification method through uncertainty calculation eliminates the need for re-reading input images for analysis, thereby significantly reducing GPU memory demands.

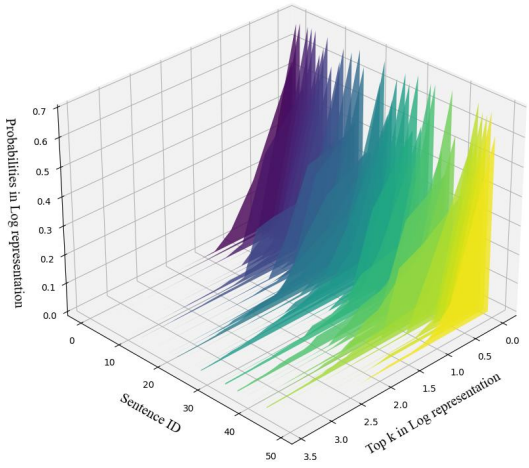


Figure 3. Probability distribution of the top k generation of each token.

Our method demonstrates substantially more pronounced advantages compared to other hallucination detection methods in all evaluated LVLMs with different architectures and sizes. We observed that Sarah achieves performance gains of +5.2% for LLaVA-v1.5-7B, +4.6% for LLaVA-v1.5-13B, 4.0% for GPT-4o, +0.5% for mPLUG-Owl3-7B and +2.5% for Llama-3.2-Vision-11B in MSCOCO-Cap. The robustness of Sarah is further validated in the Bingo dataset, where it achieves improvements of +0.8% for LLaVA-v1.5-7B, +3.9% for LLaVA-v1.5-13B, +0.6% for GPT-4o, +4.5% for mPLUG-Owl3-7B and +2.4% for Llama-3.2-Vision-11B. For the closed-ended general visual question answering task VQA v2, Sarah achieves accuracy above 96.7% across all models, demonstrating the strong generalization of Sarah.

These consistent advancements over strong

Table 3
Comparison with latest related methods on MSCOCO-Caption, Bingo and VQA v2 for LVLMs hallucination detection. In each setting, the bold **value** indicates the best result, while the underlined value represents the sub-optimal. The reported results are hallucination detection accuracy. We re-implement semantic entropy and InterrogateLLM within vision-language context. Besides, due to GAVIE's requirement for bounding box ground truth, we conducted GAVIE only on MSCOCO-CAP and VQA v2 that provides this part of the data.

Models Datasets	GAVIE	FAITHSCORE	SE	Interrogate-LLM	VL-Uncertainty	KLE	SD	Sarah
LLaVA-v1.5-7B								
Bingo	-	63.6	62.4	**66.0**	63.0	60.2	60.1	64.4
MSCOCO-Cap	56.6	61.6	54.7	**71.7**	29.8	64.0	66.1	71.3
VQA v2	60.6	**100.0**	99.2	48.8	88.2	99.4	77.0	98.7
LLaVA-v1.5-13B								
Bingo	-	59.5	64.7	66.8	60.4	66.1	58.0	**70.7**
MSCOCO-Cap	67.9	64.2	53.7	72.0	27.7	58.5	68.0	**72.6**
VQA v2	57.7	86.8	**98.0**	49.4	86.5	98.3	75.4	97.5
GPT-4o								
Bingo	-	50.8	55.2	**63.2**	59.3	60.0	61.0	61.6
MSCOCO-Cap	71.6	69.8	54.6	86.6	33.7	85.6	72.0	**89.6**
VQA v2	59.7	**100.0**	95.0	44.2	89.1	90.0	74.0	97.6
mPLUG-Owl3-7B								
Bingo	-	60.3	60.6	**71.4**	48.1	59.2	57.0	65.1
MSCOCO-Cap	63.1	60.6	53.9	**77.5**	29.5	57.1	70.0	70.5
VQA v2	58.5	**97.9**	**97.9**	45.7	82.4	96.5	76.7	96.7
Llama-3.2-Vision-11B								
Bingo	-	54.2	55.5	**67.7**	60.0	58.9	58.0	62.4
MSCOCO-Cap	59.7	68.7	52.5	**78.8**	22.1	52.0	65.0	71.2
VQA v2	74.6	**99.0**	98.8	48.7	84.7	98.0	72.1	97.9

baselines benefit from Sarah's comprehensive and rational allocation of attention to the output content, coupled with thorough exploitation of the output probability distribution data. As illustrated in Figure 4, while both Sarah and SE support token-level evaluation, Sarah demonstrates superior performance in accurately identifying hallucinatory tokens instead of high uncertainty token only. GAVIE exhibits object-centric limitations: while effective for noun-related hallucinations, it fails to detect hallucinatory verbs or other non-nominal syntactic elements. Other methods, though capable of detecting hallucinations in this context, operate solely at sentence-level granularity, resulting in inevitable detail loss.

The Superior Generalization of Sarah. In the experiments, several baseline methods demonstrated applicability only to specific types of detection tasks, while exhibiting significant performance degradation in others. For instance, in the open-ended detection task on MSCOCO-Captioning, VL-Uncertainty failed to achieve an accuracy exceeding 22.1% across all experimental settings (see Table 4). This limitation largely originates from its underlying mechanism, which relies on measuring semantic uncertainty in model outputs. Although such a design is reasonable for tasks with determinate answers, it introduces substantial errors in open-ended generation scenarios where multiple semantically diverse responses can be considered correct.

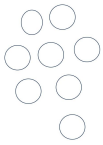
Dataset	Image	Instruction	Method	Attention Distribution
MSCOCO-Cap		Generate a short caption of the image.	Sarah	Someone is checking something on a device.
			Semantic Entropy	Someone is checking something on a device.
			GAVIE	Someone is checking something on a device.
			FaithScore	Someone is checking something on a device.
			InterrogateLLM	Someone is checking something on a device.
			VL-Uncertainty	Someone is checking something on a device.
			Kernel Language Entropy	Someone is checking something on a device.
			Semantic Density	Someone is checking something on a device.
Bingo		One of the shapes is an ellipse. What do you think?	Sarah	None of the shapes are ellipses.
			Semantic Entropy	None of the shapes are ellipses.
			FaithScore	None of the shapes are ellipses.
			InterrogateLLM	None of the shapes are ellipses.
			VL-Uncertainty	None of the shapes are ellipses.
			Kernel Language Entropy	None of the shapes are ellipses.
			Semantic Density	None of the shapes are ellipses.

Figure 4. Visualization of attention allocation patterns in hallucination detection for GPT-4 generated text. VQA v2 is excluded as it only produces single-word answers. Darker yellow hues indicate higher attention weights

Similarly, methods such as InterrogateLLM also exhibited notable performance drops on closed-set tasks like VQA v2, as documented in Table 4. This is primarily because their approach—leveraging model outputs to reverse-engineer a set of initial input questions—struggles with single-word answers that provide insufficient contextual information for reliable inference, leading to considerable degradation in effectiveness.

In contrast, Sarah consistently delivers strong performance across both closed-set and open-ended VQA tasks. Its accuracy ranks among the top or second-best in nearly all comparative evaluations, underscoring its exceptional generalization capability.

4.2 Hallucination Performance of Different LVLMs

As shown in Table 5, we present a comprehensive performance comparison of various models in terms of Sarah when benchmarked on various datasets. We have the following observations: (1) GPT-4o outperforms other counterparts in most situations. This demonstrates its preeminent capability in generating content that is factually accurate and consistent with the input. (2) For most models, the performance on the MSCOCO-Cap dataset is better than their performance on the Bingo dataset. The potential reason may be that the MSCOCO-Cap dataset only requires the model to master the ability to describe image content, while Bingo requires the model to further overcome the bias caused by the imbalance in training data, as well as the interference caused by the format of the input text prompts or input images. (3) It's worth noting that besides LLaVA-V1.5, different models have similar performance distribution across tasks. They are all better at OCR and factual reasoning tasks, but have poor adversarial robustness against interference in the input prompts and images. For instance, Llama-3.2-Vision-11b achieved 0.4574, 0.3563, 0.4667, 0.5446 and 0.5605 on the "Region Bias", "OCR Bias", "Factual Bias", "Image-to-Image Interference" and "Text-to-Image Interference" tasks, respectively. (4) Compared to others, LLaVA-V1.5 performs worse in most open-ended task. (5) All models exhibit excellent performance on the closed-ended VQA task.

4.3 Computational and Overhead Costs Analysis

To assess the detection efficiency of the proposed Sarah framework, we conduct a comparative analysis of computational resource utilization

Table 4
Hallucination failure cases.

Method	Case Details
VL-Uncertainty	**Dataset:** MSCOCO-Cap **Instruction:** Generate a short caption of the image. **Answer:** "Cozy nostalgia corner.", "A teddy bear with a blue ribbon sits in a wicker chair surrounded by books and a vintage stereo system.", "A cozy teddy bear sits nestled among books and vinyl records, adding a touch of warmth to a vintage corner.", "A cozy teddy bear with a blue scarf sits nestled among books and records in a warm, nostalgic corner.", "A cozy corner with a teddy bear and vintage vinyls." **Ground-truth Hallucination Label:** false **Predictive Hallucination Label:** true
Interrogate-LLM	**Dataset:** VQA v2 **Instruction:** Is this a typical swimsuit? **Answer:** No. **Predictive Questions:** "Do you think this is a toy?", "Is this a yes or a no?", "Does this look like a standard toilet to you?", "What type of clothing is this?", "Is this a typical swimsuit?" **Ground-truth Hallucination Label:** false **Predictive Hallucination Label:** true

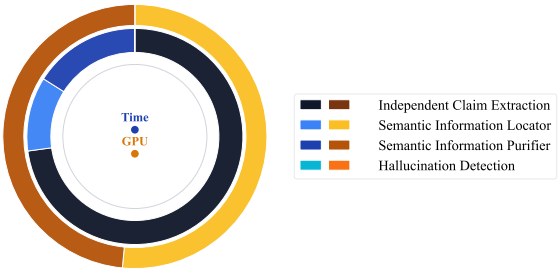


Figure 5. Comparison of time consumption and GPU memory utilization across different stages.

across all comparative methods using MSCOCO-Cap (see Table 6). Sarah operates without requiring iterative model inference or image loading, achieving a constant time complexity of 10^0 and thereby exhibiting significant advantages in temporal performance.

Further examination of Sarah's computational profile reveals distinct patterns in temporal and GPU resource allocation. As illustrated in Figure 5, the extraction of independent claims accounts for over 70% of the total execution time, followed by the purifier and the locator. Sarah's GPU resources are primarily consumed by the encoding of lexical embeddings in the locator and the part-of-speech classification in the purifier. The monetary cost associated with Sarah's invocations of external models is within an acceptable range. Specifically, only a modest expense is incurred from GPT-4 API calls.

4.4 Potential Failure Modes Analysis of External Model Invocation

GPT-4 API Calls in Independent Claim Extraction. To verify the necessity of independent claim extraction and to demonstrate the cost-effectiveness of invoking the GPT-4 API, we further conducted an experimental analysis focusing on the quality of claim extraction, the types of extraction errors, and the error propagation from this initial step.

Specifically, we conducted an independent claim extraction quality evaluation on MSCOCO-Cap and Bingo. VQA v2 was excluded from this process, as its output has been limited to a single word per instance. Each output is independently annotated by three professional annotators, and cross-validation is employed to ensure labeling consistency.

To quantitatively assess the correctness of the decomposed independent claims, we adopted the ROUGE-L as the primary indicator of semantic alignment. We further compute F1 scores between automatically decomposed claims and human annotations to assess the quality of claim extraction. Furthermore, we performed an in-depth examination of the error taxonomy and its subsequent propagation effects. Extraction errors were classified into three major categories, and corresponding datasets were independently constructed to enable a systematic analysis of how these errors propagate through the subsequent processing stages:

Table 5
Hallucination rate of different LVLMs as evaluated by Sarah.

Model	Bingo						MSCOCO-Cap	VQA v2
	Overall	Region	OCR	Factual	i2i	t2i		
LLaVA-v1.5-7B	61.3	57.9	64.0	61.6	62.9	61.3	29.0	0.7
LLaVA-v1.5-13B	59.2	64.2	54.0	54.1	65.0	58.7	26.8	0.7
GPT-4o	**38.4**	**39.8**	**16.3**	**36.7**	**44.0**	**49.3**	**13.4**	**0.5**
mPLUG-Owl3-7B	52.0	51.1	48.1	50.7	54.8	54.8	29.2	2.4
Llama-3.2-Vision-Instruct-11B	49.0	45.8	35.6	46.7	54.5	56.1	21.0	1.3

Table 6
Resource consumption across diverse hallucination detection methods on MSCOCO-Cap. This analysis evaluates whether a detection method requires (1) multi-round reasoning and (2) image inputs, both of which significantly affect GPU memory requirements. The actual iteration time for each method is also provided in the table.

Method	Multi-round	Input image	Time (s/it)
SE			10^0
Faithscore		*	10^0
GAVIE		*	10^0
InterrogateLLM	*		10^2
VL-Uncertainty	*	*	10^1
KLE			10^1
SD			10^1
Sarah			10^0

- Under-decomposition means combining two or more independent claims into one. This may amplify the model's weakness in handling long, complicated sentences. To construct the under-decomposition dataset, we consider the most extreme case of this phenomenon: the sentence is not segmented at all. This is equivalent to the model output bypassing the Independent Claim Extraction module entirely.

- Over-decomposition means unnecessarily splitting a single atomic claim into multiple segments. This could lead to information fragmentation, semantic redundancy, and even disrupt the semantic or causal logic of complete arguments. To construct the over-decomposition dataset, we extend the detected over-decomposition cases by further splitting correctly segmented sentences into additional subclauses with overlapping semantics.

Table 7
Quality Assessment for Independent Claim Extraction.

	MSCOCO-Cap	Bingo
Extraction Quality		
F1-Score	99.5	98.1
Error Type Distribution		
Under-decomposition	1	6
Over-decomposition	9	10
Full-decomposition	6	4

- Full-decomposition including information loss or generating claims with inferred content. To construct the full-decomposition dataset, we extend the detected cases by either subtly altering key words to shift sentence semantics or by retaining only fragmented portions of the segmented sentence to simulate incomplete reasoning.

As shown in Table 7, the GPT-4 based Independent Claim Extraction achieved an F1 score of 99.5% on MSCOCO-Cap and an F1 score of 98.1% on Bingo, indicating that it accurately identifies atomic claims in most cases. Over-decomposition is slightly more prevalent than the other two error types.

Further experimental results on error propagation reveal that Sarah's performance is impacted by the three decomposition failure types. Among these, under-decomposition resulted in the smallest performance drop $(-7.0 \pm 0.5)\%$, over-decomposition had a moderate impact $(-10.0 \pm 0.3)\%$, and full-decomposition was the most severe $(-16.0 \pm 0.8)\%$. This indicates that enhancing the robustness of the Independent Claim Extraction module is crucial for ensuring Sarah's reliable hallucination detection.

DistilBERT for Classification in Seman-

Table 8
Ablation study of contribution of semantic information locator and semantic information purifier for hallucination detection.

Method	Accuracy (%)
Baseline	30.9
+ Locator Only	60.1
+ Purifier Only	66.7
Full Model (Sarah)	**89.6**

tic Information Purifier. We randomly selected 20% of our dataset as a test set to evaluate the reliability of the model. Experiment results show that the model achieves notably high accuracy (95.0%) on part-of-speech classification, indicating that the risk of such errors occurring in practice is negligible.

4.5 Ablation Studies

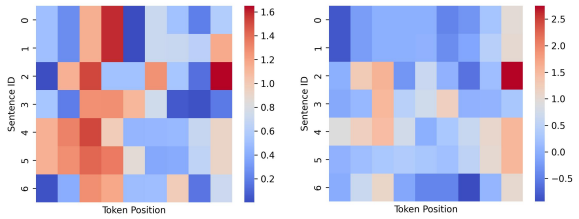


Figure 6. Hallucination scores distribution before (left) and after (right) re-weighting. After reweighting, the values at some positions with higher uncertainty are flattened due to their low semantic importance to the sentence.

Only Semantic Information Locator. To evaluate the contributions of Semantic Information Locator and Semantic Information Purifier, we conducted an ablation study, using token entropy as baseline for comparison. The experiments were performed on the Bingo dataset using GPT-4o as the generator. As shown in Table 8, the introduction of the locator significantly improved hallucination detection accuracy by 25.3%, elevating the model from near-random guessing performance (30.9%) to a robust tool capable of effective hallucination evaluation (56.2%). This demonstrates that the locator accurately identifies hallucinated content by leveraging the uneven distribution of semantic importance across tokens. As shown in Figure 6, tokens with negligible SI are ignored, as their uncertainty is unlikely to contribute to hallucinations.

Only Semantic Information Purifier. Similarly, the Semantic Information Purifier demonstrates strong performance in hallucination detection, achieving an accuracy higher than the baseline (35.8%). This highlights the module's ability to refine uncertainty quantification by filtering out expression-level variations and focusing on semantic content discrepancies. The purifier's superior performance underscores its role in reducing noise and enhancing the reliability of hallucination detection.

Better Performance on Purifier. Notably, the purifier alone outperforms the locator by 10.5%. This may be caused by the following two factors:

From the perspective of robustness, Semantic Information Purifier operates on each output token independently, which inherently provides greater stability compared to Semantic Information Locator, whose mechanism considers the entire generated content as a whole. Therefore, the locator is more easily to be influenced by the global semantic noise or spurious interference, while the purifier remains unaffected by output complexity or length, maintaining stable performance under varying conditions.

From a fundamental problem-solving perspective, the locator focuses on adjusting the relative importance of tokens—upweighting significant ones and downweighting minor ones. Purifier, however, takes a more principled approach. On one hand, it directly nullifies uncertainty probabilities that arise purely from expression-level variations. On the other hand, it aggregates the probabilities of semantically equivalent tokens—regardless of surface form and computes the entropy of these semantic clusters, ensuring that uncertainty reflects the indeterminacy of information content rather than the diversity of linguistic expression. In doing so, it directly addresses the core challenge faced by uncertainty-based methods in open-ended generation tasks: valid semantic outputs are inherently diverse, while the locator cannot correct this fundamental measurement bias.

Length of Output. We further report an ablation study on comparing the influence of length of generation on hallucination detection. Here we categorize the length of the output content into three groups: (0, 20), (20, 40), and over 40.

As illustrated in Figure 7, the performance of Sarah varies with the length of output. While all models start with relatively high accuracy, their

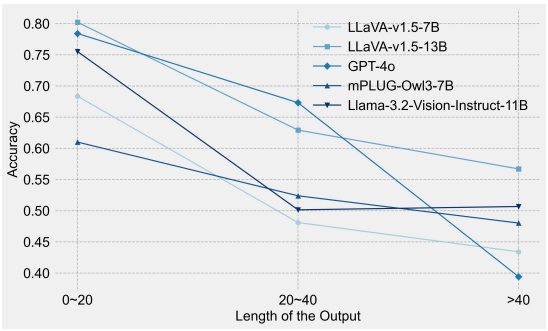


Figure 7. Ablation study of length of output.

performance drops as the length of the prediction increases. For example, LLaVA-v1.5-7B starts with a high accuracy of 68.4% for length between 0 to 20 and sustains a relatively low accuracy of 43.4% for length over 40.

Sarah's poorer performance on longer texts can be attributed to two factors: the compounding errors in the pipeline, and the inherent difficulty of evaluating longer and more complex sentences. Hallucinations represent fact-level errors that require token-level annotations for precise identification [47]. As text length and syntactic complexity increase, the number of factual statements grows correspondingly, expanding the potential positions where hallucinations may occur [48, 49]. Consequently, longer sentences have a higher likelihood of containing hallucinations, which in turn increases the probability of false negatives.

Besides, as text length increases, the potential positions of hallucination signals grow approximately linearly, while semantic density tends to decrease [44, 50]. When Semantic Information Locator within Sarah attempts to capture key tokens across longer contexts, local hallucination cues may be masked by global semantic noise or irrelevant interference, thereby reducing its detection sensitivity.

Moreover, Sarah relies on probabilistic uncertainty quantification, which is prone to variance accumulation as sequence length increases. This causes a unified detection threshold to become unstable across texts of varying lengths, leading to inconsistent performance.

Despite the aforementioned challenges, it is crucial to contextualize this limitation as a shared challenge within the field, rather than a weakness specific to our proposed method (see Table 3).

5 Conclusion

In this work, we propose Sarah, a novel uncertainty quantification based approach for evaluating and detecting hallucination on two open-ended question-answering tasks and one closed-ended question-answering task for LVLMs. The proposed method requires merely a single inference pass without image inputs and enables fine-grained measurement of hallucination-contributing positions. This capability is achieved through our novel semantic information locator and purifier. Despite the use of external models, our experiments demonstrate their cost-effectiveness, computational efficiency, and operational reliability. Experiments over "off-the-shelf" LVLMs demonstrate the superior performances of Sarah in most tasks, achieving detection accuracy of 89.6% at most. Notably, while Sarah only achieves comparable performance to InterrogateLLM (tying in evaluation metrics), its substantial computational efficiency (requiring only $1/10^1$ of InterrogateLLM's time consumption) effectively compensates for this performance limitation. Furthermore, the performance exhibits a degradation trend with increasing generation length, suggesting opportunities for feature space optimization. Our analysis also demonstrates that current LVLMs are prone to hallucination problems. We hope our work can help address the unexpected hallucination issues of LVLMs. Future directions include further eliminating the interference caused by the uncertainty of language style and word order, as well as mitigating the impact of independent claim extraction failures.

Acknowledgements

This work was supported by the Fundamental Research Funds for the Central Universities (No. 2025JBMC016).

References

[1] Z. Liang, Y. Xu, Y. Hong, P. Shang, Q. Wang, Q. Fu, K. Liu, A survey of multimodel large language models, in: Proceedings of the 3rd International Conference on Computer, Artificial Intelligence and Control Engineering, 2024, pp. 405–409.

[2] H. Liu, W. Xue, Y. Chen, D. Chen, X. Zhao, K. Wang, L. Hou, R. Li, W. Peng, A survey on hallucination in large vision-language models, arXiv preprint arXiv:2402.00253 (2024).

[3] J. Zhang, J. Huang, S. Jin, S. Lu, Vision-language models for vision tasks: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence (2024).

[4] Y. Jin, J. Li, Y. Liu, T. Gu, K. Wu, Z. Jiang, M. He, B. Zhao, X. Tan, Z. Gan, et al., Efficient multimodal large language models: A survey, arXiv preprint arXiv:2405.10739 (2024).

[5] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, et al., A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions, ACM Transactions on Information Systems 43 (2) (2025) 1–55.

[6] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms, arXiv preprint arXiv:1707.06347 (2017).

[7] S. Min, K. Krishna, X. Lyu, M. Lewis, W.-t. Yih, P. W. Koh, M. Iyyer, L. Zettlemoyer, H. Hajishirzi, Factscore: Fine-grained atomic evaluation of factual precision in long form text generation, arXiv preprint arXiv:2305.14251 (2023).

[8] E. Fadeeva, A. Rubashevskii, A. Shelmanov, S. Petrakov, H. Li, H. Mubarak, E. Tsymbalov, G. Kuzmin, A. Panchenko, T. Baldwin, et al., Fact-checking the output of large language models via token-level uncertainty quantification, arXiv preprint arXiv:2403.04696 (2024).

[9] Y. Hu, B. Liu, J. Kasai, Y. Wang, M. Ostendorf, R. Krishna, N. A. Smith, Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 20406–20417.

[10] L. Jing, R. Li, Y. Chen, X. Du, Faithscore: Fine-grained evaluations of hallucinations in large vision-language models, arXiv preprint arXiv:2311.01477 (2023).

[11] C. Cui, Y. Zhou, X. Yang, S. Wu, L. Zhang, J. Zou, H. Yao, Holistic analysis of hallucination in gpt-4v (ision): Bias and interference challenges, arXiv preprint arXiv:2311.03287 (2023).

[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. 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, Springer, 2014, pp. 740–755.

[13] J. Chen, H. Guo, K. Yi, B. Li, M. Elhoseiny, Visualgpt: Data-efficient adaptation of pretrained language models for image captioning, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 18030–18040.

[14] S. Huang, L. Dong, W. Wang, Y. Hao, S. Singhal, S. Ma, T. Lv, L. Cui, O. K. Mohammed, B. Patra, et al., Language is not all you need: Aligning perception with language models, Advances in Neural Information Processing Systems 36 (2023) 72096–72109.

[15] M. Moor, Q. Huang, S. Wu, M. Yasunaga, Y. Dalmia, J. Leskovec, C. Zakka, E. P. Reis, P. Rajpurkar, Medflamingo: a multimodal medical few-shot learner, in: Machine Learning for Health (ML4H), PMLR, 2023, pp. 353–367.

[16] Z. Peng, W. Wang, L. Dong, Y. Hao, S. Huang, S. Ma, F. Wei, Kosmos-2: Grounding multimodal large language models to the world, arXiv preprint arXiv:2306.14824 (2023).

[17] W. Wang, J. Xie, C. Hu, H. Zou, J. Fan, W. Tong, Y. Wen, S. Wu, H. Deng, Z. Li, et al., Drivemlm: Aligning multi-modal large language models with behavioral planning states for autonomous driving, arXiv preprint arXiv:2312.09245 (2023).

[18] C. Cui, Y. Ma, X. Cao, W. Ye, Y. Zhou, K. Liang, J. Chen, J. Lu, Z. Yang, K.-D. Liao, et al., A survey on multimodal large language models for autonomous driving, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 958–979.

[19] J. S. Park, J. O'Brien, C. J. Cai, M. R. Morris, P. Liang, M. S. Bernstein, Generative agents: Interactive simulacra of human behavior, in: Proceedings of the 36th annual acm symposium on user interface software and technology, 2023, pp. 1–22.

[20] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., Llama: Open and efficient foundation language models, arXiv preprint arXiv:2302.13971 (2023).

[21] S. Leng, Y. Xing, Z. Cheng, Y. Zhou, H. Zhang, X. Li, D. Zhao, S. Lu, C. Miao, L. Bing, The curse of multimodalities: Evaluating hallucinations of large multimodal models across language, visual, and audio, arXiv preprint arXiv:2410.12787 (2024).

[22] S. Strohauer, F. Wietschorke, L. Zugliani, R. Flaschmann, C. Schmid, S. Grotowski, M. Müller, B. Jonas, M. Althammer, R. Gross, et al., Site-selective enhancement of superconducting nanowire single-photon detectors via local helium ion irradiation, Advanced Quantum Technologies 6 (12) (2023) 2300139.

[23] OpenAI, GPT-4o: The Next Generation of Language Models with Enhanced Vision Capabilities, https://openai.com/blog/gpt-4o/, accessed: 2024-05-14 (2023).

[24] A. Stock, S. Schlögl, A. Groth, Tell me, what are you most afraid of? exploring the effects of agent representation on information disclosure in human-chatbot interaction, in: International Conference on Human-Computer Interaction, Springer, 2023, pp. 179–191.

[25] H. Hu, J. Zhang, M. Zhao, Z. Sun, Ciem: Contrastive instruction evaluation method for better instruction tuning, arXiv preprint arXiv:2309.02301 (2023).

[26] F. Liu, K. Lin, L. Li, J. Wang, Y. Yacoob, L. Wang, Mitigating hallucination in large multi-modal models via robust instruction tuning, arXiv preprint arXiv:2306.14565 (2023).

[27] Y. Mao, D. Lu, Y. Zhang, X. Wang, Fatrer: Full-attention topic regularizer for accurate and robust conversational emotion recognition, in: ECAI 2023, IOS Press, 2023, pp. 1688–1695.

[28] A. Rohrbach, L. A. Hendricks, K. Burns, T. Darrell, K. Saenko, Object hallucination in image captioning, arXiv preprint arXiv:1809.02156 (2018).

[29] Y. Li, Y. Du, K. Zhou, J. Wang, W. X. Zhao, J.-R. Wen, Evaluating object hallucination in large vision-language models, arXiv preprint arXiv:2305.10355 (2023).

[30] F. Liu, K. Lin, L. Li, J. Wang, Y. Yacoob, L. Wang, Mitigating hallucination in large multi-modal models via robust instruction tuning, arXiv preprint arXiv:2306.14565 (2023).

[31] Q. Li, J. Geng, C. Lyu, D. Zhu, M. Panov, F. Karray, Reference-free hallucination detection for large vision-language models, arXiv preprint arXiv:2408.05767 (2024).

[32] S. Kadavath, T. Conerly, A. Askell, T. Henighan, D. Drain, E. Perez, N. Schiefer, Z. Hatfield-Dodds, N. DasSarma, E. Tran-Johnson, et al., Language models (mostly) know what they know, arXiv preprint arXiv:2207.05221 (2022).
[33] L. Kuhn, Y. Gal, S. Farquhar, Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation, arXiv preprint arXiv:2302.09664 (2023).
[34] R. Zhang, H. Zhang, Z. Zheng, Vl-uncertainty: Detecting hallucination in large vision-language model via uncertainty estimation, arXiv preprint arXiv:2411.11919 (2024).
[35] J. Wang, Y. Zhou, G. Xu, P. Shi, C. Zhao, H. Xu, Q. Ye, M. Yan, J. Zhang, J. Zhu, et al., Evaluation and analysis of hallucination in large vision-language models, arXiv preprint arXiv:2308.15126 (2023).
[36] J. Duan, H. Cheng, S. Wang, A. Zavalny, C. Wang, R. Xu, B. Kailkhura, K. Xu, Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models, arXiv preprint arXiv:2307.01379 (2023).
[37] P. Manakul, A. Liusie, M. J. Gales, Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models, arXiv preprint arXiv:2303.08896 (2023).
[38] G. Zhang, H.-a. Gao, Z. Jiang, H. Zhao, Z. Zheng, Ctrl-u: Robust conditional image generation via uncertainty-aware reward modeling, arXiv preprint arXiv:2410.11236 (2024).
[39] C. Blundell, J. Cornebise, K. Kavukcuoglu, D. Wierstra, Weight uncertainty in neural network, in: International conference on machine learning, PMLR, 2015, pp. 1613–1622.
[40] Y. Gal, Z. Ghahramani, Dropout as a bayesian approximation: Representing model uncertainty in deep learning, in: international conference on machine learning, PMLR, 2016, pp. 1050–1059.
[41] A. Kendall, Y. Gal, What uncertainties do we need in bayesian deep learning for computer vision?, Advances in neural information processing systems 30 (2017).
[42] M.-C. De Marneffe, C. D. Manning, J. Nivre, D. Zeman, Universal dependencies, Computational linguistics 47 (2) (2021) 255–308.
[43] Y. Yehuda, I. Malkiel, O. Barkan, J. Weill, R. Ronen, N. Koenigstein, Interrogatellm: Zero-resource hallucination detection in llm-generated answers, arXiv preprint arXiv:2403.02889 (2024).
[44] Z. Geng, Y. Luo, S. Zheng, J. Wang, D. Dou, B. Chang, X. Huang, Z. Liu, B. Li, Kernel language entropy: Fine-grained uncertainty quantification for llms from semantic similarities, in: Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS 2024), 2024.
[45] X. Qiu, R. Miikkulainen, Semantic density: Uncertainty quantification for large language models through confidence measurement in semantic spacePreprint (2024).
URL https://github.com/cognizant-ai-labs/semantic-density-paper
[46] Z. Wu, Y. Hu, W. Shi, N. Dziri, A. Suhr, P. Ammanabrolu, N. A. Smith, M. Ostendorf, H. Hajishirzi, Fine-grained human feedback gives better rewards for language model training, Advances in Neural Information Processing Systems 36 (2023) 59008–59033.
[47] S. Min, K. Krishna, X. Lyu, J. Liu, J. Wieting, L. Zettlemoyer, N. A. Smith, Factscore: Fine-grained atomic evaluation of factual precision in long form text generation, arXiv preprint arXiv:2305.14251 (2023).
[48] I. Shumailov, Z. Shumaylov, Y. Zhao, Y. Gal, N. Papernot, R. Anderson, The curse of recursion: Training on generated data makes models forget, IEEE Transactions on Artificial Intelligence 5 (2) (2024) 502–514.
[49] Y. Huang, Z. Zhang, J. Zhang, Y. Li, Y. Zhang, A survey of hallucination in large foundation models, ACM Transactions on Intelligent Systems and Technology 15 (3) (2024) 1–38.
[50] C. Zhu, R. Xu, M. Zeng, X. Huang, The factual inconsistency problem in abstractive text summarization: A survey, ACM Computing Surveys 55 (12) (2023) 1–38.