Evaluating Hallucination in Large Vision-Language Models based on Context-Aware Object Similarities

Shounak Datta[*1], **Dhanasekar Sundararaman**[*2]
[1]Indian Statistical Institute
[2]Duke University
shounak.jaduniv@gmail.com, ds448@duke.edu

Abstract

Despite their impressive performance on multi-modal tasks, large vision-language models (LVLMs) tend to suffer from hallucinations. An important type is object hallucination, where LVLMs generate objects that are inconsistent with the images shown to the model. Existing works typically attempt to quantify object hallucinations by detecting and measuring the fraction of hallucinated objects in generated captions. Additionally, more recent work also measures object hallucinations by directly querying the LVLM with binary questions about the presence of likely hallucinated objects based on object statistics like top-k frequent objects and top-k co-occurring objects. In this paper, we present Context-Aware Object Similarities (CAOS), a novel approach for evaluating object hallucination in LVLMs using object statistics as well as the generated captions. CAOS uniquely integrates object statistics with semantic relationships between objects in captions and ground-truth data. Moreover, existing approaches usually only detect and measure hallucinations belonging to a predetermined set of in-domain objects (typically the set of all ground-truth objects for the training dataset) and ignore generated objects that are not part of this set, leading to under-evaluation. To address this, we further employ language model–based object recognition to detect potentially out-of-domain hallucinated objects and use an ensemble of LVLMs for verifying the presence of such objects in the query image. CAOS also examines the sequential dynamics of object generation, shedding light on how the order of object appearance influences hallucinations, and employs word embedding models to analyze the semantic reasons behind hallucinations. By providing a systematic framework to identify and interpret object hallucinations, CAOS aims to offer a nuanced understanding of both the hallucination tendencies of LVLMs and the factors contributing to object hallucinations.

Introduction

Large language models (LLMs) like LLaMA, Gemini, and Mixtral (Touvron et al. 2023a,b; Team et al. 2023; Jiang et al. 2024) have demonstrated remarkable capabilities in natural language processing tasks. Building upon this success, researchers have focused on integrating powerful LLMs with visual encoders to create large vision-language models (LVLMs) (Liu et al. 2024a; Gong et al. 2023; Zhu et al. 2023; Dai et al. 2024; Ye et al. 2023). These LVLMs leverage the language understanding abilities of LLMs while enhancing their capabilities to process and interpret visual information seamlessly. LVLMs typically utilize the visual encoders to analyze image data, while replacing the original language encoders with state-of-the-art LLMs. Through a combination of vision-language pretraining and fine-tuning on visual instructions (Wang et al. 2021), LVLMs have demonstrated impressive performance on complex tasks that require integrating visual and linguistic information.

LVLMs exhibit robust proficiency across various vision-language tasks, showcasing their versatility and adaptability. For instance, they excel in tasks such as image captioning (Herdade et al. 2019), generating descriptive textual representations of visual content to bridge the semantic gap between images and language. Additionally, LVLMs demonstrate proficiency in visual question answering (Antol et al. 2015; Wu et al. 2017), comprehending and responding to queries posed in natural language based on visual input.

The development of LVLMs marks a significant milestone in the convergence of vision and language modalities, opening up new avenues for research and application in multimodal models. However, LLMs, and similarly LVLMs, often suffer from the issue of hallucination (Rawte, Sheth, and Das 2023; Liu et al. 2024b; Xu, Jain, and Kankanhalli 2024), where the generated content contains information that is inconsistent or unfaithful to the provided input data. Hallucination refers to the phenomenon where the generated content contains nonsensical, contradictory or factually incorrect information that violates the input instructions or prompts. A common form of hallucination in LVLMs is object hallucination (Rohrbach et al. 2018; Li et al. 2023; Zhou et al. 2023), where the model describes objects or entities that are not present in the visual input. Recent studies have shown that LVLMs suffer from severe object hallucination issues (Li et al. 2023; Zhou et al. 2023), often generating descriptions that include objects inconsistent with the inputs. Hallucinations can be influenced by the visual instructions or prompts, as objects frequently occurring in the instructions or co-occurring with image objects are more prone to being hallucinated.

Hallucination in both LLMs and LVLMs can be problematic, as it can lead to the generation of unreliable or misleading information. This issue undermines the reliability of

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

[*]These authors contributed equally to this work

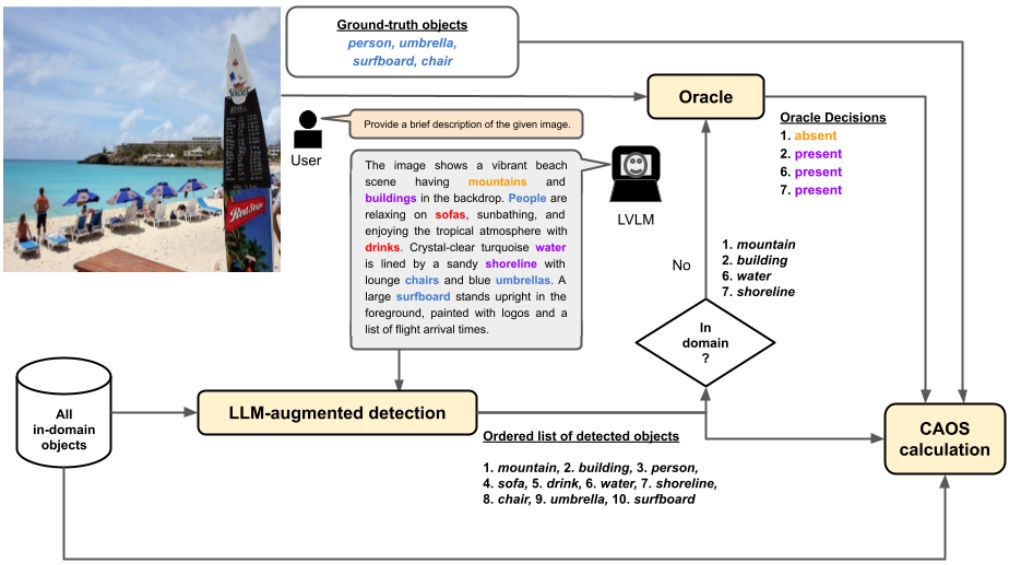


Figure 1: Overview of the CAOS framework: The CAOS framework evaluates LVLMs by generating captions for images with known ground-truth object annotations. The captions may include both in-domain and out-of-domain objects, which could be real or hallucinated. Specific color coding identifies object types: blue for real in-domain objects, red for hallucinated in-domain objects, purple for real out-of-domain objects, and orange for hallucinated out-of-domain objects. The key components of the framework are highlighted in yellow: constructing an ordered list of objects in the caption using an LLM-augmented identification module, querying an oracle (an ensemble of LVLMs) to confirm the presence or absence of out-of-domain objects (isolated on the basis of known in-domain objects in the training dataset) as ground-truth annotations are unavailable, and finally calculating CAOS scores (best viewed in color).

LVLM outputs despite their semantic coherence. Moreover, it also poses potential risks in real-world applications where accurate interpretation of visual content is of paramount importance. Therefore, it has prompted researchers to develop evaluation benchmarks and mitigation techniques to detect and mitigate hallucination in these models. Existing works like (Li et al. 2023; Pham and Schott 2024) use object statistics such as top-k objects and frequently co-occurring objects to determine the causes of hallucination and belong to a family of methods which ensure LVLMs respond to a set of predefined questions to evaluate hallucinations (Lovenia et al. 2023). Others works may focus on a particular aspects of object hallucinations, like numbers (Zhang, Zhang, and Wan 2024). Moreover, the object statistics mentioned above can also then be used to mitigate hallucinations by post-hoc rectification (Zhou et al. 2023). In addition to the object statistics, the generated captions also contain additional information about an LVLM's tendency to hallucinate, in the form of semantic relationships between the objects in the generated text. However, existing methods do not attempt to investigate the effect that the interplay between object statistics and the semantics of the objects present either in the query image or the already generated context can have on object hallucinations during the remainder of the generation process. Consequently, in this paper, we present a novel approach named context-aware object similarities (CAOS) for systematically evaluating object hallucination in LVLMs using the generated captions. By focusing on the interplay between generated objects (hallucinated or otherwise), their position in the generated captions, ground-truth objects as well as the object statistics of the training data, CAOS attempts to offers a more nuanced understanding of object hallucination dynamics and encompasses the the following key improvements over existing works:

- **Detect hallucinations of out-of-domain objects:** Unlike existing methods which rely either on rule-based parsing of known in-domain objects (Rohrbach et al. 2018) or on known occurrence statistics of these objects from the training data (Li et al. 2023), we propose a novel way to augment object detection from generated captions using LLMs and to verify the existence of candidate out-of-domain objects in the images using an oracle consisting of an ensemble of LVLMs.
- **Sequential generation dynamics:** Recognizing the sequential nature of caption generation, CAOS investigates how the order of object appearance influences hallucination. By delving into this, our approach sheds light on how the dynamics of object generation impacts object hallucinations in addition to other critical factors, namely ground-truth objects and frequently occurring objects in the training dataset.
- **Semantic reasons behind hallucinations:** Unlike traditional evaluation metrics, CAOS employs word embedding models to scrutinize the semantic relation between hallucinated objects and ground-truth objects, other objects in the generated captions and frequent objects from the training dataset.

With an aim to foster the development of more robust and reliable LVLMs for diverse real-world applications, CAOS intend to enrich the existing literature on evaluating object hallucinations by offering a framework for better identifying object hallucinations in LVLMs, quantifying the semantic reasons behind such hallucinations, and for interpreting these results (potentially in tandem with existing metrics such as CHAIR (Rohrbach et al. 2018) and POPE (Li et al. 2023)) to obtain a meaningful ranking not just based on an LVLM's affinity to hallucinate but also the factors influencing such hallucinations.

Related Works

There are a number of related works on evaluating object hallucinations in LVLMs. Rohrbach et al. (2018) proposed the CHAIR family of metrics which uses rule-based parsing to identify in-domain objects in an LVLM-generated annotation and then calculates the fraction of hallucinated objects per caption and the fraction of annotations with hallucinated objects in a set of generated annotations. Another relevant precursor to our work is POPE (Li et al. 2023). POPE measures the tendency of an LVLM to hallucinate objects by asking yes/no questions to the model about whether certain objects exist in the image, instead of directly evaluating the generated annotations. The choice of objects to be used for such queries can be random, or based on object statistics of the pre-training datasets, such as objects known to co-occur with ground-truth objects, or most frequent objects in the dataset. H-POPE (Pham and Schott 2024) is an extension of POPE that assesses hallucinations in object existence as well as attributes by hierarchically refining the yes/no questions from coarse-to-fine-grained, progressively probing about the attributes of the objects in the image. Lovenia et al. (2023) proposed NOPE, that uses a static set of negative pronouns to determine if a model hallucinates. There are a couple of works that detect object hallucination and perform post-hoc tuning to generate captions that do not contain hallucinations (Zhou et al. 2023; Dai et al. 2022).

Background

Large Vision-Language Models

An LVLM architecture comprises of a vision encoder, a language encoder (i.e., an LLM), and a cross-modal alignment network. The vision backbone and cross-modal alignment networks are usually used to convert an input image I into a set of n visual tokens $X_{1:n}$, whereas the accompanying text query is converted to text tokens $Y_{1:q}$ and the subsequent response by the LVLM, parameterized by weights θ, is generated based on the probabilities

$$p(Y_{q+1:q+r}) = \prod_{t=q+1}^{q+r} p_\theta(Y_t|Y_{<t}, X_{1:n}). \quad (1)$$

The training pipeline of LVLMs entails several crucial steps:

Pre-training on Unimodal Data: Initially, the vision encoder and the language encoder undergo pre-training on large-scale unimodal datasets, encompassing image and text data, respectively.

Image-Text Alignment Pre-training: Subsequently, these encoders are aligned through image-text alignment pre-training. This alignment enables LVLMs to generate coherent and meaningful captions for given images by leveraging the synergy between visual and textual modalities.

Fine-tuning on Image-Text Instructions: The aligned LVLM model undergoes further fine-tuning on image-text instructions to refine its ability to generate satisfactory answers to natural language questions related to specific images. This fine-tuning process enhances the model's performance on diverse multimodal tasks.

Object Hallucination

Object hallucination occurs when a LVLM references objects that are not actually present in the image it is describing. This can produce unexpected outcomes, particularly in applications like visual question answering, image captioning, or event detection, where accurate identification of objects is crucial. Several factors can contribute to these hallucinations:

Common Objects in Training Data: LVLMs might hallucinate objects that are frequently represented in their training datasets. Examples of such objects include "person," "dining table," and "chair," which are prevalent in visual instruction datasets.

Co-occurring Objects: Hallucinations may also arise when LVLMs predict objects that commonly appear together with the actual objects in the image, based on patterns observed in the training data.

Impact of Visual Instructions: The type of visual instructions used during training plays a significant role in hallucination tendencies. For instance, LVLMs like InstructBLIP, which are trained on diverse public datasets with concise instructions, are more likely to generate accurate, albeit brief responses. In contrast, models trained with extended synthetic instructions from unimodal language models may be prone to hallucinating due to inconsistencies or excessive detail in the synthetic data.

Influence from the already generated context: In addition to the above factors which are known to impact object hallucinations, we also postulate that the objects already mentioned in the prior context at any given point during generation may also cause hallucinations in a manner similar to the co-occurrence statistics of the ground-truth objects mentioned above. On probing LLaVA (Liu et al. 2024a) and mPLUG-Owl (Ye et al. 2023) with the instruction *"Provide a brief description of the given image."* for a subset of 2000 images from the MSCOCO (Lin et al. 2014) validation set (identical to that used by POPE (Li et al. 2023)), we find that respectively 20% and 16% of the hallucinated objects happen to be the most frequent object to have co-occurred in the training dataset with at least one preceding objects in the already generated part of the annotation. Thus, the past objects in the already generated part of an annotation can also influence object hallucinations in the remainder of the annotation which is yet to be generated. While it is straightforward to measure the fraction of hallucinated objects that

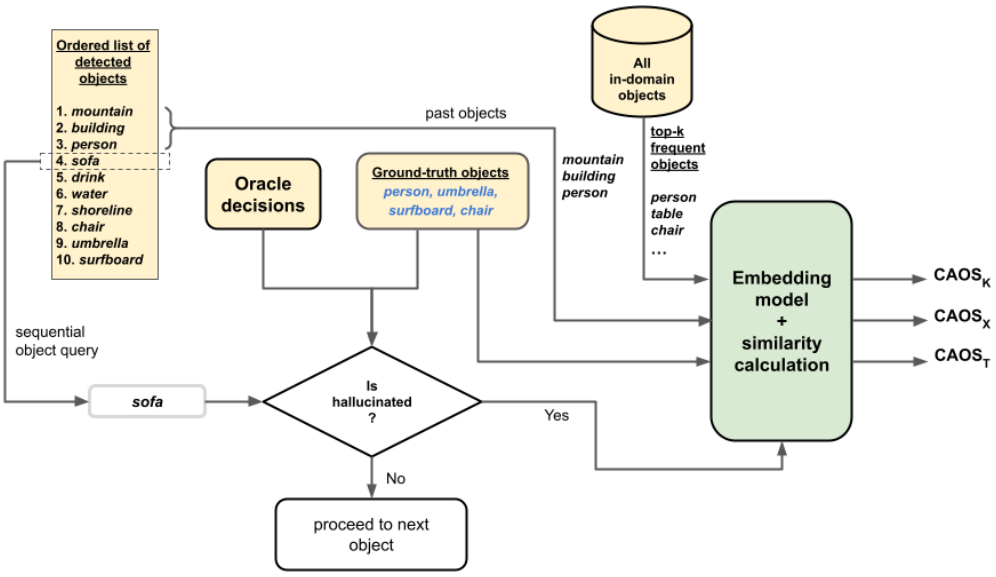


Figure 2: CAOS scores are calculated for in-domain and out-of-domain hallucinated objects which are respectively identified from the ordered list of generated objects using ground-truth annotations and oracle decisions (about presence or absence in the image). CAOS$_T$, CAOS$_X$, and CAOS$_K$ are calculated as the maximum cosine similarity between the embeddings of the hallucinated object and those of ground-truth objects, preceding objects in the generated caption, and top-k frequent objects in the training dataset, respectively. Inputs to the CAOS calculation module are highlighted in yellow for clarity (best viewed in color).

fall in this category for known in-domain objects, we can not do the same for out-of-domain objects (i.e. objects not belonging to any of the labeled classes in the training data), due to the lack of co-occurrence information.

CAOS: Context-Aware Object Similarities

To account for all the different factors that can influence object hallucinations, we propose a set of evaluation metrics based on context-aware object similarities, called CAOS, to holistically measure how the contents of the image, the sequential ordering of generated objects, as well as the dominant contents of the training dataset influences hallucination. The proposed group of measures consists of CAOS$_T$, CAOS$_X$, CAOS$_K$, CAOS$_{T/X}$, CAOS$_{X/K}$, and CAOS$_{avg}$. For a given word embedding model E, the CAOS$_T$, CAOS$_X$ and CAOS$_K$ scores measure the maximum cosine similarity between the embeddings of a hallucinated object and a set of other objects. In particular, CAOS$_T$ measures the maximum cosine similarity for a given hallucinated object with all the ground-truth objects present in the image. CAOS$_X$, on the other hand, measures the maximum cosine similarity between a hallucinate object and all past objects appearing before itself in the generated caption. In practice, we also consider all ground-truth objects, irrespective of their position in the generated caption, to be valid past objects for all hallucinated objects, since the ground-truth objects may always influence generation due to their presence in the image. Finally, since it is well-known that the most frequent objects present in the training datasets can also influence hallucinations (Li et al. 2023), CAOS$_K$ measures the maximum cosine similarity between the embeddings of the hallucinated object and the top-k frequent objects in the training dataset (MSCOCO in our experiments).

Furthermore, since it is often relatively more tolerable for hallucinations to be semantically related to the ground-truth objects known to be present in the image than to other objects in the generated caption, we calculate CAOS$_{T/X}$ to be the ratio between CAOS$_T$ and CAOS$_X$. A high CAOS$_{T/X}$ signifies that the hallucinations are mostly influenced by ground-truth objects. Similarly, it may be relatively more tolerable for the hallucinations to be related to past objects in the generated caption (including all ground-truth objects) than to frequent objects in the training dataset which may not have any relation to the contents of the image being described. Therefore, a high value of CAOS$_{T/K}$, which is the ratio between CAOS$_X$ and CAOS$_K$, may be desirable. Lastly, we also calculate CAOS$_{avg}$ which is the mean of CAOS$_T$, CAOS$_X$, and CAOS$_K$. We argue that a high value of CAOS$_{avg}$ is also desirable in most cases. This is because high CAOS$_{avg}$ values denote that the hallucinations can be accounted for by the mentioned factors and is less likely to have been caused by unknown eccentricities of the LVLM.

Given an image, the associated ground-truth object labels, and a caption generated by the LVLM for the image, we begin by identifying all objects in the captions. We then identify which of the objects are hallucinated based on the ground-truth object labels (or using an oracle for out-of-domain objects). In order of their appearance in the generated caption, we calculate the CAOS scores for all the hallucinated objects. Consequently, for a given caption containing multiple hallucinated objects, we report average values

Algorithm 1: Calculating CAOS

Inputs: LVLM p_θ to be evaluated.
Image I with ground-truth object labels.
Set G of known objects from fine-tuning dataset labels
Set K of top-k frequent objects from fine-tuning dataset
Rule-based object parser P
LLM q_ϕ for object identification and oracle O
Embedding model E
Cosine similarity operator S_{cos}
Outputs: Scores CAOS_T, CAOS_X, CAOS_K, $\text{CAOS}_{T/X}$, $\text{CAOS}_{X/K}$, and CAOS_{avg}.

Generate caption for I using p_θ.
Identify list L_1 of objects in the generated caption belonging to G using P.
Identify list L_2 of objects in the generated caption not in G, using q_ϕ.
Combine lists $L = L_1 \cup L_2$ preserving the order of the objects in the generated caption.
Construct map $M : L \rightarrow \{0, 1\}$ labeling genuine objects as 1 and hallucinated objects as 0 using ground-truth labels for $l \in L_1$ and oracle O for $l \in L_2$.
Initialize lists T and X with ground-truth objects.
Expand sets $T = T \cup T'$ and $X = X \cup T'$, where $T' = \{l \in L_2 | M(l) = 1\}$.
Initialize empty set H, T_S, X_S, and K_S.
for all $l \in L$ in order **do**
    if $M(l) = 0$ **then**
        Add l to set H.
        Add $\max_{o \in T} S_{cos}(E(l), E(o))$ to set T_S.
        Add $\max_{o \in X} S_{cos}(E(l), E(o))$ to set X_S.
        Add $\max_{o \in K} S_{cos}(E(l), E(o))$ to set K_S.
    end if
    Add l to set X.
end for
Calculate $\text{CAOS}_T = \sum T_S / |H|$.
Calculate $\text{CAOS}_X = \sum X_S / |H|$.
Calculate $\text{CAOS}_K = \sum K_S / |H|$.
Calculate $\text{CAOS}_{T/X} = \text{CAOS}_T / \text{CAOS}_X$.
Calculate $\text{CAOS}_{X/K} = \text{CAOS}_X / \text{CAOS}_K$.
Calculate $\text{CAOS}_{avg} = (\text{CAOS}_T + \text{CAOS}_X + \text{CAOS}_K)/3$.

for all the CAOS scores over all hallucinated objects in the caption.

Augmenting object identification using LLM We observe that the standard rule-based parsing which is used by existing methods like CHAIR (Rohrbach et al. 2018) to identify objects in a caption fails to identify objects beyond the set of known objects in the training dataset. Moreover, we observed that the rule-based approach may sometimes fail to even identify the ground-truth objects present in the caption. This can lead to lower recall scores for the LVLM being evaluated when ground-truth objects are missed. On the other hand, this can also lead to under-evaluation of object hallucinations when hallucinated objects are not detected. Therefore, we augmented the rule-based method with an additional list of objects identified by an LLM (LLaMA-2-7B (Touvron et al. 2023b) in our experiments), based on the generated caption, aided by in-context learning on a handful of examples (5-shot examples in our experiments). We also tally the detected objects in this augmented list with the actual words in the caption to weed-out any potential hallucinations by the LLM.

We also evaluate the efficacy of the LLM-augmented object detection by manually extracting objects from the captions generated by LLaVA on a randomly chosen subset of 100 MSCOCO (Lin et al. 2014) validation images and comparing these with the objects detected by the rule-based approach and that of the LLM-augmented approach. Due to its rule-based nature, the former approach has perfect precision (i.e. it never detects objects that are not present in the caption). However, we found that the rule-based approach returns an empty list of objects for two captions. On the flip side, this approach is prone to missing objects present in the generated captions, resulting in a recall of 59%. The LLM-augmented approach, on the other hand, has perfect recall and almost perfect precision at 97%. The slight dip in precision occurs due to a handful of instances where the LLM-augmented detection method mistook adjectives to be objects, such as "night" in "night scene". However, such rare misdetections are further corrected for as a side-effect of the next stage of our process, where we use an oracle to ascertain whether a detected object actually exists in the given image.

Oracle using an ensemble of LVLMs Since we have no ground-truth reference to identify whether the additional out-of-domain objects identified using the LLM are actually present in the image, we further label these identified objects as either genuine or hallucinated based on an oracle consisting of an ensemble of LVLMs. Given an object and the original image, we query each of the LVLMs in the ensemble with a query of the form *"Does the image contain <object>? Please respond with only Present or Absent."*, to force the model to vote on the presence or absence of the object in the image. For our experiments, we use an ensemble of InstructBLIP (Dai et al. 2024), LLaVA-7B (Liu et al. 2024a), mPLUG-Owl (Ye et al. 2023), and MiniGPT-4 (Zhu et al. 2023), and break ties in favor of absence to minimize false positives and penalize potential hallucinations.

We also conduct a manual inspection of a randomly chosen subset of 100 MSCOCO (Lin et al. 2014) validation images and the corresponding captions generated by MultimodalGPT (Gong et al. 2023) (which is not part of the ensemble) to evaluate the accuracy of the "Present"/"Absent" decisions made by the ensemble as well as the individual models. We find that the ensemble-based oracle is able to correctly identify the presence or absence of 93.43% of the 259 non-MSCOCO objects in these images. Among the individual models, InstructBLIP was found to have the highest accuracy at 89.57%, followed closely by LLaVA at 88.42%, both MiniGPT-4 and mPLUG-Owl perform slightly worse at 84.94%. We also illustrate the performance of the oracle with some visual examples in the Appendix. While this may seem similar to querying the model in POPE (Li et al. 2023), it should be noted that the ensemble is likely to misdetect the presence or absence of a smaller number of objects than the individual model being evaluated in POPE. However, these rare misdetections of non-MSCOCO objects may slightly perturb the average CAOS scores because out-

Model	InstructBLIP	LLaVA	mPLUG-Owl	MiniGPT-4	Multimodal-GPT
Precision	**0.98**	0.85	0.79	0.92	0.88
Recall	0.62	**0.85**	0.74	0.78	0.67
# Objects	2.22	4.97	4.49	4.91	3.26
$CHAIR_S$	**0.04**	0.51	0.56	0.33	0.32
POPE-F1	**0.84**	0.68	0.67	0.74	0.67
$CAOS_T$–GloVe	0.30	0.38	0.37	0.40	0.40
$CAOS_X$–GloVe	0.32	0.41	0.41	0.45	0.42
$CAOS_K$–GloVe (k=3)	0.52	0.50	0.51	0.47	0.47
$CAOS_{T/X}$–GloVe	0.94	0.93	0.90	0.89	**0.95**
$CAOS_{X/K}$–GloVe	0.62	0.82	0.80	**0.96**	0.89
$CAOS_{avg}$–GloVe	0.38	0.43	0.43	**0.44**	0.43
$CAOS_T$–MiniLM-L6	0.26	0.40	0.37	0.35	0.42
$CAOS_X$–MiniLM-L6	0.27	0.43	0.40	0.40	0.45
$CAOS_K$–MiniLM-L6 (k=3)	0.52	0.54	0.53	0.45	0.51
$CAOS_{T/X}$–MiniLM-L6	**0.96**	0.93	0.92	0.88	0.93
$CAOS_{X/K}$–MiniLM-L6	0.52	0.80	0.75	**0.89**	0.88
$CAOS_{avg}$–MiniLM-L6	0.35	**0.46**	0.43	0.40	**0.46**

Table 1: Average Precision, Recall, the number of objects per generated caption, $CHAIR_S$, POPE-F1, and CAOS scores using GloVe as well as MiniLM-L6 embeddings across 5 different LVLMs.

of-domain objects tend to have higher $CAOS_T$ and $CAOS_X$ values and lower $CAOS_K$ values compared to known in-domain MSCOCO objects (see Figure 5).

The complete method is detailed in Algorithm 1 while Figure 1 illustrates the overall framework and Figure 2 explains the flow of CAOS score calculation.

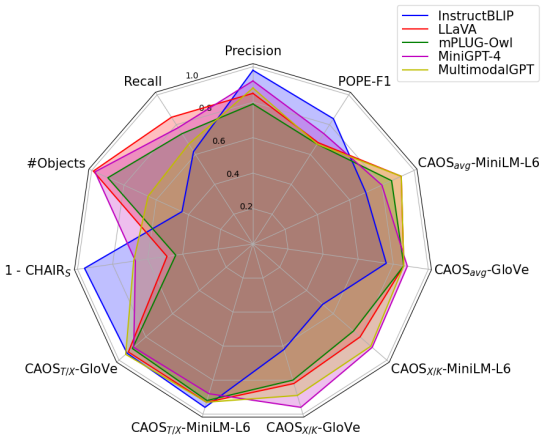
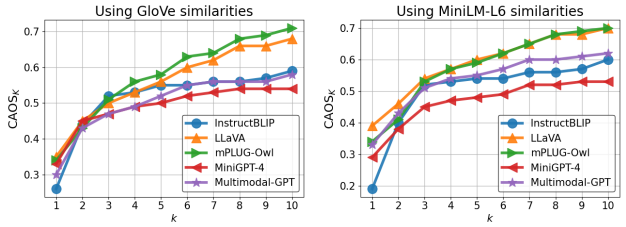


Figure 3: Comparison between all evaluated LVLMs on Precision, Recall, the average number of objects per generated caption (normalized with division by 5), 1 - $CHAIR_S$ (greater is better), POPE-F1, and $CAOS_{T/X}$, $CAOS_{X/K}$, as well as $CAOS_{avg}$ scores (normalized using multiplication by 2) using both GloVe and MiniLM-L6 embeddings (best viewed in color).

Results

We conduct all our experiments on the subset of 2000 images from the MSCOCO (Lin et al. 2014) validation set identical to that used by POPE (Li et al. 2023). For each of these 2000 images, we use the instructions *"Provide a brief description of the given image."* and *"Question: Generate a short caption of the image. Answer: "* to probe 5 LVLMs, namely InstructBLIP (Dai et al. 2024), LLaVA (Liu et al. 2024a), mPLUG-Owl (Ye et al. 2023), MiniGPT-4 (Zhu et al. 2023), and MultimodalGPT (Gong et al. 2023) to generate captions for the image. A detailed comparison of the LVLMs, in terms of model sizes and training recipes is shown in the Appendix.

In Table 1, we report all 6 CAOS scores, namely $CAOS_T$, $CAOS_X$, $CAOS_K$, $CAOS_{T/X}$, $CAOS_{X/K}$, and $CAOS_{avg}$, using two different word embedding models, GloVe (Pennington, Socher, and Manning 2014) and MiniLM-L6 (Wang et al. 2020; all-MiniLM-L6-v2). We choose these embedding models as they form embeddings based on slightly different objectives. GloVe aims to assign similar embeddings to objects which co-occur is a corpus of documents, while MiniLM-L6 assigns semantically meaningful embeddings based on pretraining on a large number of language tasks such as question answering, natural language inference, etc. For the $CAOS_K$ scores, we choose $k = 3$ based on the trends of $CAOS_K$ scores across k values (see a full discussion in the subsequent section). Additionally, we also report precision (i.e., the fraction of detected objects which are not hallucinations) and recall (which is the fraction of actual ground-truth objects from the query image that are mentioned in the generated caption). Due to the relative instability of the precision measure (related to $CHAIR_I$ (Rohrbach et al. 2018)), we also report POPE-F1 scores for all the models (Li et al. 2023). We also report the fraction of captions having hallucinated objects, which is equivalent to the $CHAIR_S$ metric proposed by Rohrbach et al. (2018). LVLMs which are prone to generating shorter captions with fewer objects consequently have less scope for hallucina-



(a) (b)

Figure 4: Trend of CAOS$_K$ scores with k varying from 1 to 10 (best viewed in color).

tion. However, such models also have limited ability to generate faithful descriptive captions for the images. Ideally, a model should have the ability to generate as many objects as needed to generate a particular image while minimizing hallucinations. Therefore, we have also reported the average number of objects generated per caption, as a loose proxy for the capacity of an LVLM to generate articulate captions.

Overall, the CAOS scores trend similarly across the LVLMs for GloVe and MiniLM-L6. All of the LVLMs exhibit higher CAOS$_K$ scores relative to the CAOS$_T$ and CAOS$_X$ scores, implying that all of these models have a high tendency to hallucinate verbatim the frequent objects from the training dataset (i.e. MSCOCO). We also observe a trade-off between the CAOS$_T$ scores (or consequently the related CAOS$_X$ scores) and the CAOS$_K$ scores. In other words, LVLMs like InstructBLIP, LLaVA and mPLUG-Owl which have higher CAOS$_K$ scores tend to have lower CAOS$_T$ (and CAOS$_X$) scores. This hints at the fact that these models have a relatively higher tendency to hallucinate common objects from the training dataset while the remainder of the models, viz. MiniGPT-4 and Multimodal-GPT, have a higher tendency for hallucinations related to the ground-truth objects and the preceding objects in the generated response.

In order to better compare the performance of the different LVLMs, we also create a radar plot in Figure 3, excluding the CAOS$_T$, CAOS$_X$, and CAOS$_K$ scores (since those scores do not offer a straightforward way to compare the models, which can be done instead with the other CAOS scores). InstructBLIP exhibits the highest 1-CHAIR$_S$ (or equivalently the lowest CHAIR$_S$) value as well as the highest precision and POPE-F1 scores, implying that it hallucinates less than the other models. However, this is in part due to its tendency to generate a lower number of objects per caption which also results in low recall. Moreover, InstructBLIP also has low CAOS$_{X/K}$ scores. On the other hand, MiniGPT-4 appears to overall have competitive performance across most of the axes, suggesting that it hallucinates less compared to most of the other contenders and that a relatively greater fraction of it's hallucinations are related to the ground-truth objects actually present in the image (a somewhat tolerable property for hallucinations that do happen).

Effect of varying k on CAOS$_K$

The choice of k for CAOS$_K$ can affect the CAOS scores. Therefore, we vary k to observe how CAOS$_K$ is impacted for the LVLMs. The CAOS$_K$ scores for all the LVLMs are shown in Figure 4. We observe that the CAOS$_K$ scores for all models saturate to some extent at $k = 3$, suggesting that the top-3 most frequent objects in MSCOCO disproportionately appear as hallucinations for all the models. This suggests that a choice of $k = 3$ can capture most of the information about how the frequent in-domain objects can affect object hallucinations. Additionally, despite the diminishing rate of impact beyond $k = 3$, we also observe that LLaVA and mPLUG-Owl continue to be impacted more by frequent objects beyond the top 3.

Comparison between different subsets of objects

In Figure 5, we investigate how the CAOS$_T$, CAOS$_X$, and CAOS$_K$ scores vary across different subsets of hallucinated objects. Since our proposed LLM-augmented object detection is meant to uncover the hallucination of out-of-domain objects, we inspect what the CAOS scores look like for just the hallucinated in-domain MSCOCO objects and just the out-of-domain (denoted as non-MSCOCO) objects. The CAOS scores for both groups largely follow a similar trend to that of all objects, but the in-domain MSCOCO objects seems to have a more pronounced influence from the frequently occurring objects, as implied by the elevated CAOS$_K$ scores. Conversely, the out-of-domain objects have a lower albeit non-negligible impact from the frequent MSCOCO objects. This suggests that even the hallucinated out-of-domain objects are semantically influence by the frequent MSCOCO objects to a certain extent. Further, due to the higher impact from the top-3 frequently occurring objects, we also investigate how the CAOS scores change when the hallucinated instances of only the 3 most common

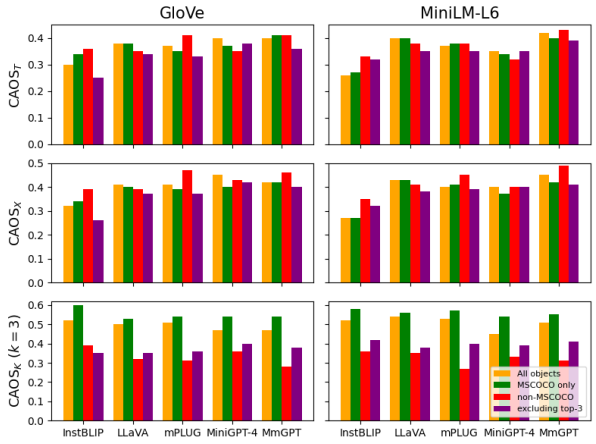


Figure 5: Comparison of CAOS$_T$, CAOS$_X$, and CAOS$_K$ scores for all hallucinated objects, only MSCOCO in-domain hallucinated objects, non-MSCOCO hallucinated objects, and all objects barring the top-3 most frequent MSCOCO in-domain objects (best viewed in color).

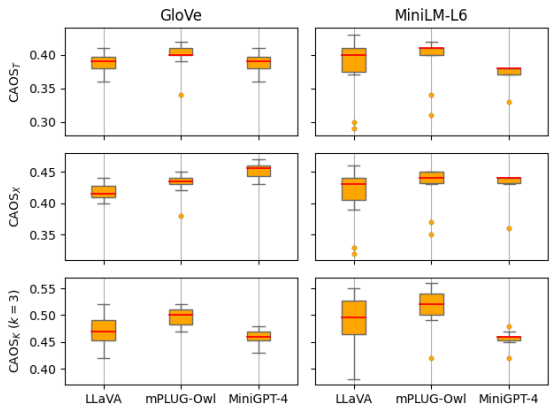


Figure 6: Variability of CAOS scores for LLaVA, mPLUG-Owl, and MiniGPT-4 across 14 different instructions with averages shown in red (best viewed in color).

Model	LLaVA	mPLUG-Owl	MiniGPT-4
CAOS$_T$–GloVe	0.39	0.40	0.39
CAOS$_X$–GloVe	0.42	0.43	0.45
CAOS$_K$–GloVe (k=3)	0.47	0.50	0.46
CAOS$_{T/X}$–GloVe	**0.93**	**0.93**	0.87
CAOS$_{X/K}$–GloVe	0.89	0.86	**0.98**
CAOS$_{avg}$–GloVe	0.43	**0.44**	0.43
CAOS$_T$–MiniLM-L6	0.39	0.40	0.37
CAOS$_X$–MiniLM-L6	0.41	0.43	0.43
CAOS$_K$–MiniLM-L6 (k=3)	0.49	0.52	0.46
CAOS$_{T/X}$–MiniLM-L6	**0.95**	0.93	0.86
CAOS$_{X/K}$–MiniLM-L6	0.84	0.83	**0.93**
CAOS$_{avg}$–MiniLM-L6	0.43	**0.45**	0.42

Table 2: Average results for LLaVA, mPLUG-Owl, and MiniGPT-4 across 14 different instructions.

MSCOCO objects are excluded. A similar dip in CAOS$_K$ is seen for hallucinated objects excluding the top 3, signifying that the top-3 objects disproportionately appear as hallucinations for all models.

Consistency across different prompt styles

To analyze the sensitivity of CAOS scores to changes in the text prompt, we rerun our experiments on LLaVA, mPLUG-Owl, and MiniGPT-4 with 12 new instructions in addition to the 2 instructions already used in Table 1. The list of all instructions is shown in the Appendix. We find that the average results over all 14 instructions, detailed in Table 2, are overall quite similar to those in Table 1, indicating that CAOS scores are largely stable to changes in instructions. We also illustrate how CAOS$_T$, CAOS$_X$ and CAOS$_K$ values vary across the instructions in Figure 6. For a given model and specific CAOS score, we observe some variability across instructions, which is to be expected. Furthermore, while the CAOS scores have different ranges across the different LVLMs, the ranges maintain an ordering similar to that exhibited by the mean CAOS scores (shown in red), further indicating stability across instructions. Finally, the CAOS scores with MiniLM-L6 embeddings look slightly different than those using GloVe. CAOS scores calculated using MiniLM-L6 embeddings seem to be slightly more prone to having outliers than their corresponding GloVe counterparts and CAOS scores for LLaVA have higher variance than the other two models with MiniLM-L6 embeddings.

Conclusion and Limitations

Existing methods do not investigate the interplay between object statistics and the semantics of objects in query images or generated context, leaving a gap in understanding object hallucinations during the generation process. To address this, we propose the novel CAOS framework for systematically evaluating object hallucination in LVLMs using generated captions. CAOS focuses on the interaction between generated objects (hallucinated or otherwise), their positional dynamics, ground-truth objects, and object statistics from training data to offer a deeper understanding of hallucination dynamics. Our key contributions include detecting out-of-domain hallucinated objects using LLMs and an oracle based on an ensemble of LVLMs, analyzing sequential generation dynamics, and employing word embedding models to explore the semantic relationships behind hallucinations. We conduct experiments with several diverse LVLMs and find that CAOS effectively identifies hallucinations and provides insights into trade-offs, such as the tendency of certain models to hallucinate frequent objects from training datasets versus ground-truth-related objects. Notably, MiniGPT-4 demonstrates competitive performance across metrics, suggesting that it tends to hallucinate fewer and more contextually relevant objects. In summary, CAOS provides a systematic and nuanced framework for understanding hallucination dynamics, supporting the development of more reliable and robust LVLMs.

It is important to note the limitations of our study despite the extensive exploration undertaken. We focus only on object hallucination in LVLMs, leaving out other performance aspects such as the ability to generate more articulate or contextually coherent responses. Moreover, we use a partial validation set of 2000 MSCOCO images due to computational constraints, which could potentially skew our results. However, for consistency with existing works, we retained the same subset of images employed by Li et al. (2023). Additionally, our reliance on rule-based object detection, augmented by an LLM and an oracle for verification, may occasionally lead to inaccuracies due to errors in any of these components, though such cases are likely rare. Finally, our analysis considers only a small subset of state-of-the-art LVLMs, excluding some newer or closed-source models. Nevertheless, we view these findings as a step forward in developing more reliable and human-aligned LVLMs. Future work could extend the CAOS framework to encompass other types of hallucinations, such as spatial, relational, or numerical inconsistencies, offering a holistic evaluation of an LVLM's multimodal understanding.

References

all-MiniLM-L6-v2. 2021. https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2. [Online; accessed 10-Dec-2024].

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Zitnick, C. L.; and Parikh, D. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, 2425–2433.

Dai, W.; Li, J.; Li, D.; Tiong, A. M. H.; Zhao, J.; Wang, W.; Li, B.; Fung, P. N.; and Hoi, S. 2024. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36.

Dai, W.; Liu, Z.; Ji, Z.; Su, D.; and Fung, P. 2022. Plausible may not be faithful: Probing object hallucination in vision-language pre-training. *arXiv preprint arXiv:2210.07688*.

Gong, T.; Lyu, C.; Zhang, S.; Wang, Y.; Zheng, M.; Zhao, Q.; Liu, K.; Zhang, W.; Luo, P.; and Chen, K. 2023. Multimodal-gpt: A vision and language model for dialogue with humans. *arXiv preprint arXiv:2305.04790*.

Herdade, S.; Kappeler, A.; Boakye, K.; and Soares, J. 2019. Image captioning: Transforming objects into words. *Advances in neural information processing systems*, 32.

Jiang, A. Q.; Sablayrolles, A.; Roux, A.; Mensch, A.; Savary, B.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Hanna, E. B.; Bressand, F.; et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Li, Y.; Du, Y.; Zhou, K.; Wang, J.; Zhao, W. X.; and Wen, J.-R. 2023. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*.

Lin, T.; Maire, M.; Belongie, S. J.; Bourdev, L. D.; Girshick, R. B.; Hays, J.; Perona, P.; Ramanan, D.; Doll'a r, P.; and Zitnick, C. L. 2014. Microsoft COCO: Common Objects in Context. *CoRR*, abs/1405.0312.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2024a. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Liu, H.; Xue, W.; Chen, Y.; Chen, D.; Zhao, X.; Wang, K.; Hou, L.; Li, R.; and Peng, W. 2024b. A survey on hallucination in large vision-language models. *arXiv preprint arXiv:2402.00253*.

Lovenia, H.; Dai, W.; Cahyawijaya, S.; Ji, Z.; and Fung, P. 2023. Negative object presence evaluation (nope) to measure object hallucination in vision-language models. *arXiv preprint arXiv:2310.05338*.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543.

Pham, N.; and Schott, M. 2024. H-POPE: Hierarchical Polling-based Probing Evaluation of Hallucinations in Large Vision-Language Models. *arXiv preprint arXiv:2411.04077*.

Rawte, V.; Sheth, A.; and Das, A. 2023. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*.

Rohrbach, A.; Hendricks, L. A.; Burns, K.; Darrell, T.; and Saenko, K. 2018. Object hallucination in image captioning. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Team, G.; Anil, R.; Borgeaud, S.; Wu, Y.; Alayrac, J.-B.; Yu, J.; Soricut, R.; Schalkwyk, J.; Dai, A. M.; Hauth, A.; et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

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

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

Wang, W.; Wei, F.; Dong, L.; Bao, H.; Yang, N.; and Zhou, M. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33: 5776–5788.

Wang, Z.; Yu, J.; Yu, A. W.; Dai, Z.; Tsvetkov, Y.; and Cao, Y. 2021. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*.

Wu, Q.; Teney, D.; Wang, P.; Shen, C.; Dick, A.; and Van Den Hengel, A. 2017. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, 163: 21–40.

Xu, Z.; Jain, S.; and Kankanhalli, M. 2024. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*.

Ye, Q.; Xu, H.; Xu, G.; Ye, J.; Yan, M.; Zhou, Y.; Wang, J.; Hu, A.; Shi, P.; Shi, Y.; et al. 2023. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

Zhang, H.; Zhang, J.; and Wan, X. 2024. Evaluating and Mitigating Number Hallucinations in Large Vision-Language Models: A Consistency Perspective. *arXiv preprint arXiv:2403.01373*.

Zhou, Y.; Cui, C.; Yoon, J.; Zhang, L.; Deng, Z.; Finn, C.; Bansal, M.; and Yao, H. 2023. Analyzing and mitigating object hallucination in large vision-language models. *arXiv preprint arXiv:2310.00754*.

Zhu, D.; Chen, J.; Shen, X.; Li, X.; and Elhoseiny, M. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

List of Instructions

The list of all instructions used for our experiments is detailed in Table 3.

Sl. No.	Instruction
1.	"Provide a brief description of the given image."
2.	"Question: Generate a short caption of the image. Answer: "
3.	"Create a short textual summary for the image."
4.	"Generate a concise description for the image."
5.	"Write a succinct summary capturing the essence of the image."
6.	"Craft a brief narrative that encapsulates the scene depicted in the image."
7.	"Summarize the image with a few descriptive words."
8.	"Compose a short, evocative caption for the image."
9.	"Describe the image using minimal words but maximum impact."
10.	"Formulate a concise and descriptive caption for the image."
11.	"Write a short, impactful description for the image."
12.	"Sum up the image in a few words, capturing its essence effectively."
13.	"Craft a brief but descriptive caption for the image."
14.	"Write a concise summary that encapsulates the image's message or mood."

Table 3: List of all instructions used for our experiments.

Details of models used

All the evaluated LVLMs used in our experiments consist of three main parts: a vision encoder (VE), a large language model (LLM), and lastly an alignment model (AN) meant to serve as a common mapping between the VE and the LLM. We report a comparison between the LVLMs in Table 4, comparing the size of the models, and the respective pretraining and finetuning recipes (on visual instruction data).

Model	VE	AN	LLM	Pre-training			Fine-tuning		
				VE	AN	LLM	VE	AN	LLM
InstructBLIP	ViT-G/14	Q-Former	Vicuna_{13B}	Frozen	Trained	Frozen	Frozen	Trained	Frozen
LLaVA	ViT-L/14	Linear	LLaMA_{13B}	Frozen	Trained	Frozen	Frozen	Trained	Trained
mPLUG-Owl	ViT-L/14	Attention	LLaMA_{7B}	Trained	Trained	Frozen	Frozen	Frozen	LoRA
MiniGPT-4	ViT-G/14	Linear	Vicuna_{13B}	Frozen	Trained	Frozen	Frozen	Trained	Frozen
Multimodal-GPT	ViT-L/14	Attention	LLaMA_{7B}	Frozen	Trained	Frozen	Frozen	Frozen	LoRA

Table 4: Comparison of the evaluated LVLMs: "Trained" denotes full pretraining or finetuning, while "LoRA" denotes that low-rank adapters were trained with the backbone frozen.

Out-of-domain object verification using the Oracle

We further illustrate the performance of the oracle on captions generated by MultimodalGPT (which is not part of the ensemble of LVLMs used as the oracle) for a few example images in Table 5. The illustration consists of successful, ambiguous (such as "sun" in the 3rd example which can be interpreted as both the star, which is absent, or sunlight, which is present) as well as erroneous decisions ("apron" in the 5th example image) made by the oracle.

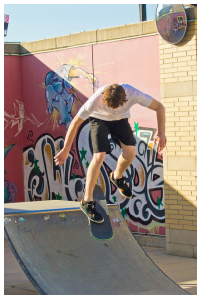
Image	Detected out-of-domain objects (with actual tags for presence/absence)	Oracle decisions
	lettuce: **absent** ingredients: **present** tomato: **present** onion: **absent**	lettuce: **absent** ingredients: **present** tomato: **present** onion: **absent**
	stall: **present** clipboard: **absent** shirt: **present** pants: **present** barn: **present**	stall: **present** clipboard: **absent** shirt: **present** pants: **present** barn: **present**
	jeans: **absent** wall: **present** t-shirt: **present**	jeans: **absent** wall: **present** t-shirt: **present**
	sun: **ambiguous** beach: **present** rock: **absent**	sun: **present** beach: **present** rock: **absent**
	apron: **absent** pineapple: **absent**	apron: **present** pineapple: **absent**

Table 5: Examples of out-of-domain object verification by the oracle.