MetaToken: Detecting Hallucination in Image Descriptions by Meta Classification

Laura Fieback[1,2][a], Jakob Spiegelberg[1][b] and Hanno Gottschalk[2][c]

[1]*Volkswagen AG, Berliner Ring 2, 38440 Wolfsburg, Germany*

[2]*Mathematical Modeling of Industrial Life Cycles, Institute of Mathematics, TU Berlin, Berlin, Germany*

Keywords: Hallucination Detection, Large Vision Language Models, Multimodal Language Models, Meta Classification.

Abstract: Large Vision Language Models (LVLMs) have shown remarkable capabilities in multimodal tasks like visual question answering or image captioning. However, inconsistencies between the visual information and the generated text, a phenomenon referred to as hallucinations, remain an unsolved problem with regard to the trustworthiness of LVLMs. To address this problem, recent works proposed to incorporate computationally costly Large (Vision) Language Models in order to detect hallucinations on a sentence- or subsentence-level. In this work, we introduce **MetaToken**, a lightweight binary classifier to detect hallucinations on token-level at negligible cost. Based on a statistical analysis, we reveal key factors of hallucinations in LVLMs. MetaToken can be applied to any open-source LVLM without any knowledge about ground truth data providing a calibrated detection of hallucinations. We evaluate our method on four state-of-the-art LVLMs outperforming baseline methods by up to $46.50pp$ in terms of area under precision recall curve values.

1 INTRODUCTION

LVLMs have demonstrated impressive visual-language understanding skills by aligning text and visual features. However, besides their remarkable zero-shot performance on visual downstream tasks, LVLMs suffer from the problem of hallucinations (Li et al., 2023b; Liu et al., 2024b; Rohrbach et al., 2018) inherited from the underlying Large Language Models (LLMs) or caused by faulty interpretation of the image input by the vision branch. In the context of LVLMs, hallucination refers to the problem of inconsistencies between the generated text and the visual input (Liu et al., 2024b) diminishing the trustworthiness of these models. Especially in safety-critical applications like autonomous driving (Gao et al., 2024; Tian et al., 2024) or medicine (Jiang et al., 2024; Li et al., 2023a), the reliability of the underlying model is indispensable for decision making. In order to address this problem, recent works (Liu et al., 2024a; Gunjal et al., 2023; Zhao et al., 2024; Dai et al., 2023b; Xing et al., 2024) have proposed additional instruction tuning datasets and pre-training strategies to detect and mitigate hallucinations on a sentence- or subsentence-level. Another common strategy comprises stacked L(V)LMs to post-hoc detect and rectify hallucinations (Wu et al., 2024; Yin et al., 2023; Jing et al., 2023).

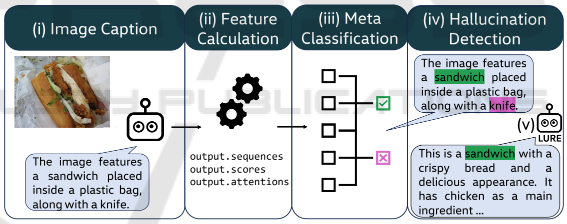

Figure 1: MetaToken. Based on generated image captions (i), we calculate our proposed input features (ii) (see Sec. 3.2). Afterwards, we apply the trained meta classifier (iii) to detect hallucinated and true objects (iv). Moreover, (v) MetaToken can be easily integrated into LURE (Zhou et al., 2023) to improve the hallucination mitigation.

In this work, we tackle the problem of object hallucination in image captions. To this end, we introduce MetaToken, a lightweight hallucination detection method which can be applied to any open-source LVLM. MetaToken builds up on the idea of meta classification (Lin and Hauptmann, 2003; Hendrycks and Gimpel, 2017; Chen et al., 2019; Rottmann et al., 2020; Fieback et al., 2023) to detect hallucinated ob-

[a] https://orcid.org/0009-0003-3456-6766
[b] https://orcid.org/0000-0002-6550-0087
[c] https://orcid.org/0000-0003-2167-2028

Fieback, L., Spiegelberg, J. and Gottschalk, H.
MetaToken: Detecting Hallucination in Image Descriptions by Meta Classification.
DOI: 10.5220/0013165700003912

In *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2025) - Volume 2: VISAPP*, pages 126-137
ISBN: 978-989-758-728-3; ISSN: 2184-4321

Table 1: Related Work on Hallucination Detection. A comparison of existing approaches on hallucination detection with respect to computational efficiency, i.e., whether the respective method can be implemented without an additional dataset, fine-tuning or prompting an L(V)LM. ✓ indicates 'yes', ✗ indicates 'no'.

method	w/o add. dataset	w/o fine-tuning	w/o prompting
LogicCheckGPT (Wu et al., 2024)	✓	✓	✗
Woodpecker (Yin et al., 2023)	✓	✓	✗
M-HalDetect (Gunjal et al., 2023)	✗	✗	✗
HaELM (Wang et al., 2023)	✗	✗	✗
FAITHSCORE (Jing et al., 2023)	✓	✓	✗
UNIHD (Chen et al., 2024)	✓	✓	✗
Ours	✓	✓	✓

jects on token-level based on the model output only. Fig. 1 depicts our approach. In contrast to existing methods, our approach neither requires an additional dataset, fine-tuning an L(V)LM nor cost-intensive L(V)LM prompting. Within a comprehensive statistical analysis, we investigate a broad set of input features which are indicative of hallucinations providing deep insights into the sources of this specific type of model errors. We evaluate our method on four state-of-the-art (SOTA) LVLMs (Dai et al., 2023a; Ye et al., 2023; Zhu et al., 2023; Huang et al., 2023) achieving area under receiver operator characteristic curve values (Davis and Goadrich, 2006) of up to 92.12% and area under precision recall curve values (Davis and Goadrich, 2006) of up to 84.01%. Moreover, we show that our method can be incorporated into the LVLM Hallucination Revisor (LURE) mitigation method (Zhou et al., 2023). While the initial LURE method reduces hallucinations by up to 52.98%, we achieve a hallucination reduction by up to 56.62% through the superior precision-recall-ratio of MetaToken. Our main contributions are as follows:

- We propose and investigate a comprehensive set of statistics as potential factors of object hallucinations.
- Based on these statistics, we introduce MetaToken, a lightweight binary classifier to detect object hallucinations as a post-hoc method. MetaToken can be applied to any open-source LVLM without any knowledge about the ground truth data.
- We show that MetaToken can be easily integrated into the LURE mitigation method, outperforming the initial LURE results through a superior precision-recall-ratio.

The remainder of this work is structured as follows: An overview over related work in the field of LVLM hallucination and meta classification is provided in Sec. 2. In Sec. 3, we introduce MetaToken, which comprises a formal definition of meta classification and our proposed input features followed by the experimental details in Sec. 4. Finally, we present our numerical results in Sec. 5 and discuss limitations of our work in Sec. 6.

2 RELATED WORK

2.1 Hallucinations in LVLMs

Hallucinations in LVLMs can occur on different semantic levels, where coarse-grained object hallucination (Rohrbach et al., 2018) refers to objects generated in the language output, which are not depicted in the input image, whereas fine-grained hallucination describes inconsistencies with respect to object attributes or relations between objects (Li et al., 2023b; Liu et al., 2024b). For a comprehensive survey on hallucinations in LVLMs, we refer to (Liu et al., 2024b).

The problem of hallucination mitigation is mainly tackled by either retraining the model with an instruction tuning dataset (Liu et al., 2024a; Gunjal et al., 2023), rectifying image captions as a post-processing step or incorporating new pre-training or generation strategies. LURE (Zhou et al., 2023) serves as a post-hoc method to rectify object hallucinations by training an LVLM-based revisor to reconstruct less hallucinatory descriptions. MARINE (Zhao et al., 2024) enriches the visual context of LVLMs by incorporating object grounding features into the LLM input. In (Dai et al., 2023b), a new pre-training objective is introduced to mitigate object hallucinations by improving the object-level image-text alignment. Since the reliability of the generated language output still remains an unsolved problem, many studies focus on the problem of hallucination detection. In (Wu et al., 2024), the problem of hallucination detection and mitigation is solved simultaneously by raising logical correlated questions and checking for logical consistency throughout the generated answers afterwards. Similarly, Woodpecker (Yin et al., 2023) serves as a post-processing hallucination detection and correction method incorporating visual knowledge validation for both instance- and attribute-level hallucinations. A human-labeled dataset is published in (Gunjal et al., 2023), which is used to train an LLM-based classifier to classify between accurate and inaccurate sentences. In (Wang et al., 2023), a hallucination evaluation framework is introduced by training an LLM to distinguish between hallucinated and hallucination-free image captions. Both, (Jing et al., 2023) and (Chen et al., 2024), propose a pipeline consisting of several LVLMs and LLMs to verify each claim contained in the generated language output. In this work, we tackle the problem of object hallucination detection using meta classification. In contrast to existing methods, our approach neither requires an additional dataset, fine-tuning an L(V)LM nor cost-intensive L(V)LM prompting for claim verification (see Tab. 1).

2.2 Meta Classification

In classical machine learning, meta classification refers to the problem of how to best combine predictions from an ensemble of classifiers (Lin and Hauptmann, 2003). In terms of deep learning, this concept has been transferred to the classification whether a prediction is true or false based on uncertainty features (Hendrycks and Gimpel, 2017). Several works have applied this idea to natural language processing (Vasudevan et al., 2019; Liu et al., 2022; Gui et al., 2024), image classification (Chen et al., 2019), semantic segmentation (Rottmann et al., 2020; Rottmann and Schubert, 2019; Maag et al., 2020; Fieback et al., 2023), video instance segmentation (Maag et al., 2021) and object detection (Schubert et al., 2021; Kowol et al., 2020). We are the first to transfer the idea of meta classification to the problem of hallucination detection for LVLMs. Based on a statistical analysis of key factors of hallucinations in LVLMs, we identify input features outperforming classical uncertainty-based statistics.

2.3 Hallucination Evaluation

Since different studies (Rohrbach et al., 2018; Dai et al., 2023b) have shown that standard image captioning metrics like BLEU (Papineni et al., 2002), METEOR (Lavie and Agarwal, 2007), CIDEr (Vedantam et al., 2015) and SPICE (Anderson et al., 2016) are not capable of measuring hallucinations properly, a common hallucination evaluation method is the Caption Hallucination Assessment with Image Relevance (CHAIR) metric (Rohrbach et al., 2018). The CHAIR metric measures the proportion of hallucinated objects in an image caption by matching the objects in the generated text against the ground truth annotations (Lin et al., 2014). Further datasets and evaluation methods have been proposed to evaluate the performance of LVLMs across multiple multimodal tasks (Fu et al., 2023; Gunjal et al., 2023; Lovenia et al., 2024; Liu et al., 2023). While some of the proposed evaluation methods ask LLMs to output quality-related scores (Liu et al., 2025; Liu et al., 2024a; Yu et al., 2024) or measure the image-text similarity (Hessel et al., 2021), other methods (Li et al., 2023b; Fu et al., 2023; Wang et al., 2024) use a prompt template to query hallucination-related questions and force the model to answer either *'yes'* or *'no'*. However, the results in (Li et al., 2023b) and (Fu et al., 2023) have shown that LVLMs tend to answer *'yes'*, which results in a low recall for hallucinated objects. Moreover, the LLM- and similarity-based scores (Liu et al., 2025; Liu et al., 2024a; Yu et al., 2024; Hessel et al., 2021) evaluate the entire image caption in terms of continuous scores instead of providing a binary label for each generated object. Thus, we rely on the CHAIR metric to evaluate hallucinations.

3 METHOD

The aim of our method is to detect hallucinations in the text output of LVLMs leveraging the idea of meta classification. To this end, we build input features based on the model output that have been shown to correlate with hallucinations. These features are used to train a lightweight binary meta model to classify between hallucinated and true objects. At inference time, we can detect hallucinations by computing the proposed features and applying the trained meta model afterwards (see Fig. 1). A formal definition of meta classification is provided in Sec. 3.3.

3.1 Notation

Typically, LVLMs generate language output in an auto-regressive manner by predicting the probability distribution of the next token over the entire vocabulary $\mathcal{V}$ given the input image x, the provided prompt q as well as the already generated tokens. For this purpose, the image x as well as the prompt q are tokenized into $u+1$ image tokens $t_{x_0},\dots,t_{x_u}$ and $v+1$ prompt tokens $t_{q_0},\dots,t_{q_v}$, respectively.

We denote the sequence of generated output tokens by $s=(t_0,\dots,t_K)$ with sequence length $K+1$. Moreover, let $s_i=(t_0,\dots,t_i)$ denote the generated output sequence at generation step i. The probability of generating token $t_{i+1} \in \mathcal{V}$ at generation step $i+1$ given the input image x, the provided prompt q and the already generated tokens s_i can be formulated as $p(t_{i+1}|x,q,s_i)$. For a shorter notation, we define $\hat{p}_{i+1} = p(t_{i+1}|x,q,s_i)$. Furthermore, let p_{i+1} denote the probability distribution at generation step $i+1$ over the dictionary $\mathcal{V}$ and $|\mathcal{V}|$ the cardinality of $\mathcal{V}$.

Given the language output s, we extract all objects contained in the generated text. We denote the set of objects contained in the sequence s by $O_s = \{o_0,\ldots,o_z\}$. Since the generated string of an object might consist of several tokens, we define for every object $o_j \in O_s$ the start token $t_{o_{j,s}}$ at position $0 \le o_{j,s} \le K$ as well as the end token $t_{o_{j,e}}$ at position $0 \le o_{j,e} \le K$.

3.2 Input Features

Recent works (Rohrbach et al., 2018; Wang et al., 2023) have investigated influencing factors of object hallucinations. First, the results in (Wang et al., 2023) indicate that LVLMs often generate true segments at the beginning while the risk of hallucinations increases at the later part of the generated responses. Thus, we take account of the relative position (Eq. (1)) of a generated object and the absolute occurrence (Eq. (2)) of the object in the generated text. Second, to account for the over-reliance of LVLMs on language priors during the generation process (Rohrbach et al., 2018; Wang et al., 2023), we consider the mean absolute attention on the image tokens (Eq. (3)). Finally, we regard the model uncertainty through different dispersion measures (Eq. (4)-(11)) which have been shown to correlate with model errors in different fields (Rottmann and Schubert, 2019; Schubert et al., 2021; Vasudevan et al., 2019) including the sequence score[1] calculated during the LVLM generation process. For a generated object $o_j \in O_s$ from the output sequence $s = (t_0,\ldots,t_K)$, we define

- the **relative position**
$$P_{o_j} = o_{j,s}/(K+1), \tag{1}$$
- the **absolute occurrence** of object o_j in s
$$N_{o_j} = \sum_{l=0}^{z} \mathbb{1}_{\{o_l=o_j\}}, \tag{2}$$
- for every attention head $g = 0,\ldots,G-1$, the **mean absolute attention** of the start token $t_{o_{j,s}}$ on the image tokens $t_{x_0},\ldots,t_{x_u}$
$$A^g_{o_j} = \frac{1}{u+1}\sum_{r=0}^{u} |\text{Attention}_{t_{o_{j,s}}}(t_{x_r})|, \tag{3}$$
where $\text{Attention}_{t_i}(t_n)$ denotes the attention of the generated token t_i at generation step i on the input token t_n,
- the **log probability**
$$L_{o_j} = \sum_{i=o_{j,s}}^{o_{j,e}} \log \hat{p}_i, \tag{4}$$
- the **cumulated log probability**
$$C_{o_j} = \sum_{i=0}^{o_{j,e}} \log \hat{p}_i, \tag{5}$$
- the sequence **score** with *length_penalty* parameter l_p
$$S_{o_j} = \frac{1}{(o_{j,e})^{l_p}} \sum_{i=0}^{o_{j,e}} \log \hat{p}_i, \tag{6}$$
- the **variance**
$$V_{o_j} = \frac{1}{|\mathcal{V}|}\sum_{t\in\mathcal{V}} (\log p_{o_{j,s}}(t) - \mu)^2 \tag{7}$$
with $\mu = \frac{1}{|\mathcal{V}|}\sum_{t\in\mathcal{V}} \log p_{o_{j,s}}(t)$,
- the **entropy** (Shannon, 1948)
$$E_{o_j} = -\frac{1}{\log|\mathcal{V}|}\sum_{t\in\mathcal{V}} p_{o_{j,s}}(t)\log p_{o_{j,s}}(t), \tag{8}$$
- the **variation ratio**
$$R_{o_j} = 1 - p_{o_{j,s}}(t_{\max}), \quad t_{\max} = \max_{t\in\mathcal{V}} p_{o_{j,s}}(t), \tag{9}$$
- the **probability margin**
$$M_{o_j} = R_{o_j} + \max_{t\in\mathcal{V}\setminus\{t_{\max}\}} p_{o_{j,s}}(t), \quad \text{and}, \tag{10}$$
- the **probability difference**
$$D_{o_j} = \log p_{o_{j,s}}(t_{\max}) - \log \hat{p}_{o_{j,s}}. \tag{11}$$

Finally, for an object $o_j \in O_s$, we define the total set of input features as
$$\begin{aligned}\mathcal{M}_{o_j} = \{&P_{o_j}, N_{o_j}, A^0_{o_j},\ldots,A^{G-1}_{o_j}, L_{o_j},\\ &C_{o_j}, S_{o_j}, V_{o_j}, E_{o_j}, R_{o_j}, M_{o_j}, D_{o_j}\}\end{aligned} \tag{12}$$
with cardinality $|\mathcal{M}_{o_j}| = 10 + G$.

3.3 Hallucination Detection Using Meta Classification

Let $\mathcal{M}$ denote a set of input features with cardinality $|\mathcal{M}|$. The idea of meta classification consists of training a lightweight binary meta model based on the input features $\mathcal{M}$ to classify between true and false predictions, i.e., to detect true and hallucinated objects in the generated output s. To this end, let
$$f : \mathbb{R}^{|\mathcal{M}|} \to \{0,1\} \tag{13}$$

[1] https://huggingface.co/docs/transformers/main/en/main_classes/text_generation\#transformers.GenerationMixin.compute_transition_scores

Table 2: BDD100K Synonyms. A list of synonyms for the BDD100K object categories (Yu et al., 2020).

BDD Object	Synonyms
person	human, man, woman, driver, people, someone, somebody, citizen, human being, walker, pedestrian
rider	cyclist, bicyclist, bike rider, biker, motorcyclist, motor biker, motorbike user, motorcycle user
car	automobile, vehicle, auto, suv, motorcar, ride, roadster, taxi
bus	coach, minibus, shuttle, omnibus, motorbus, passenger vehicle, trolleybus, school bus, tour bus
truck	lorry, pickup, van, semi-truck, rig, dump truck, cargo truck, delivery truck, garbage truck
bike	bicycle, cycle, pedal bike, road bike, mountain bike, velocipede
motor	motorcycle, scooter
traffic light	stoplight, signal light, traffic signal, red light, green light, traffic control signal, road signal, semaphore, stop light
traffic sign	direction sign, railroad crossing sign, road sign, signpost, traffic marker, stop sign
train	metro, tram

denote the binary classifier. We denote the set of training captions by $\mathcal{S}^{\text{train}}$ and the corresponding set of generated objects by

$$O_{\mathcal{S}^{\text{train}}} = \bigcup_{s \in \mathcal{S}^{\text{train}}} O_s. \tag{14}$$

For every generated $o_j \in O_{\mathcal{S}^{\text{train}}}$ we build an input vector $m_{o_j} \in \mathbb{R}^{|\mathcal{M}_{o_j}|}$ representing the feature set $\mathcal{M}_{o_j}$ (Eq. (12)) and define the label $y_{o_j} \in \{0,1\}$ according to the CHAIR evaluation (see Sec. 4.2). After standardizing the inputs, we use the set

$$\{(m_{o_j}, y_{o_j}) \quad | \quad j = 0, \dots, |O_{\mathcal{S}^{train}}| - 1\} \tag{15}$$

to train the classifier f.

Given a generated caption s at inference time, we calculate the input vector $m_{o_j} \in \mathbb{R}^{|\mathcal{M}_{o_j}|}$ for every object $o_j \in O_s$ and apply the trained binary meta classifier f to detect hallucinated objects. Note that the input vector m_{o_j} can be calculated in an automated manner based on the model output only, without any knowledge of the ground truth data.

4 EXPERIMENTAL SETTINGS

4.1 Datasets

We evaluate our method on the MSCOCO and BDD100K datasets. The MSCOCO dataset (Lin et al., 2014) is a large-scale dataset for object detection, segmentation, and image captioning comprising more than $200K$ labeled images. The BDD100K dataset (Yu et al., 2020) consists of $100K$ labeled street scene images including labels for object detection, semantic segmentation and instance segmentation. We randomly sample $5{,}000$ images from the validation sets and produce image captions s for four SOTA LVLMs. We use 80% of the generated captions as training set $O_{\mathcal{S}^{\text{train}}}$ and validate our method on the remaining 20% denoted as $O_{\mathcal{S}^{\text{val}}}$. In our experiments, we average our results over ten randomly sampled training-validation splits. The corresponding standard deviations are given in parentheses.

4.2 Hallucination Evaluation

The CHAIR metric (Rohrbach et al., 2018) is an automated hallucination evaluation method which has been introduced to measure hallucinations for the MSCOCO dataset (Lin et al., 2014). By matching the generated text s against the ground truth objects, CHAIR provides a binary label for every generated object category and a wide range of corresponding synonyms (Lu et al., 2018) indicating whether the object $o_j \in O_s$ is true, i.e., contained in the image, or hallucinated. We follow the same methodology to evaluate hallucinations on the BDD100K dataset (Yu et al., 2020). For every BDD100K object category, we create a comprehensive list of synonyms (see Tab. 2) and match the LVLM output against the ground truth labels of the BDD100K object detection dataset where true objects are encoded as 0 and hallucinated objects are encoded as 1. Finally, the proportion of hallucinated objects in an image caption is defined as

$$\text{CHAIR}_i = \frac{|\{\text{hallucinated objects}\}|}{|\{\text{all objects mentioned}\}|}. \tag{16}$$

4.3 Large Vision Language Models

We evaluate our approach on four SOTA open-source LVLMs, i.e., InstructBLIP (Vicuna-7B) (Dai et al., 2023a), mPLUG-Owl (LLaMA-7B) (Ye et al., 2023), MiniGPT-4 (Vicuna-7B) (Zhu et al., 2023), and LLaVa 1.5 (Vicuna-7B) (Huang et al., 2023), all of them using $G = 32$ attention heads. We use nucleus sampling (Holtzman et al., 2020) and the prompt

"Describe all objects in the image."

Table 3: LVLM Performance. Evaluation results of SOTA LVLMs with respect to the average number of generated objects per image (#obj.) and CHAIR_i (in %). The best results in each block are highlighted.

	MSCOCO		BDD100K	
Model	#obj.	$\text{CHAIR}_i \downarrow$	#obj.	$\text{CHAIR}_i \downarrow$
InstructBLIP	5.6	**10.4**	5.8	**22.07**
mPLUG-Owl	6.9	30.2	5.8	33.44
MiniGPT-4	4.4	13.6	3.7	29.12
LLaVa	7.3	18.7	7.1	29.72

Table 4: Classifier Configurations. The configurations applied in our experiments for the sklearn.linear_model.LogisticRegression (LR) and sklearn.ensemble.GradientBoostingClassifier (GB) classifier.

	scikit-learn	random_state	solver	max_iter	tol
LR	1.3.2	0	saga	1000	$1e-3$
GB	1.3.2	0	-	-	-

for all image caption generations. The performance of the LVLMs considered with respect to the average number of generated objects per image as well as the hallucination rate in terms of CHAIR_i (Eq. (16)) is summarized in Tab. 3.

4.4 Evaluation Metrics and Meta Models

We evaluate our method based on the accuracy ACC, the area under receiver operator characteristic curve $AUROC$ (Davis and Goadrich, 2006) and the area under precision recall curve $AUPRC$ (Davis and Goadrich, 2006). The receiver operator characteristic curve illustrates the performance of a binary classifier by plotting the true positive rate against the false positive rate at various decision thresholds indicating the ability to distinguish between both classes. The precision recall curve plots precision values against the recall at various decision thresholds accounting for imbalance in the underlying dataset. Since we observe imbalanced data with respect to object instance hallucinations (see Tab. 3), the main focus in our evaluation is on the $AUROC$ and $AUPRC$ value. We compare two binary meta models, i.e., a classifier based on a logistic regression (LR) and a gradient boosting (GB) meta model (see Tab. 4 for the configuration details).

4.5 Baseline

We use the reference-free token-level algebraic confidence TLC-A (Petryk et al., 2023) as our baseline. We consider the log probability-based token-level confidence L (see Eq. (4)) and the entropy-based confidence E (see Eq. (8)). For both confidence measures,

Table 5: Computational Time. The average time for feature calculation per image (feature), classifier training using 4,000 image captions (train) and inference on 1,000 image captions (predict).

	feature (sec.)	train (sec.)	predict (sec.)
LR	0.07174	0.47187	0.00198
GB	0.07174	54.00010	0.00385

Table 6: Expected Calibration Error. The ECE for the LR and GB MetaToken classifier.

	ECE (in %) $\downarrow$	
Model	LR	GB
InstructBLIP	$0.81^{(\pm 3.2e-2)}$	$1.29^{(\pm 4.6e-2)}$
mPLUG-Owl	$1.36^{(\pm 1.3e-1)}$	$2.01^{(\pm 6.9e-2)}$
MiniGPT-4	$1.35^{(\pm 9.5e-2)}$	$1.43^{(\pm 8.8e-2)}$
LLaVa	$1.05^{(\pm 2.0e-2)}$	$1.38^{(\pm 8.1e-2)}$

we train a baseline classifier (LR and GB) in the one-dimensional space, i.e.,

$$f^{\text{baseline}} : \mathbb{R} \rightarrow \{0,1\} \tag{17}$$

with training set $\{(L_{o_j}, y_{o_j}) | j = 0, \dots, |O_{\mathcal{S}^{\text{train}}}| - 1\}$ and $\{(E_{o_j}, y_{o_j}) | j = 0, \dots, |O_{\mathcal{S}^{\text{train}}}| - 1\}$, respectively.

Note that a direct comparison of our approach to the detection methods listed in Tab. 1 is not possible. While our method tackles the problem of token-level object hallucination, the listed methods either evaluate hallucinations on a sentence- or subsentence-level on their own human-labeled dataset (Gunjal et al., 2023; Wang et al., 2023) or based on atomic facts extracted from the image captions using LLMs, which are then labeled using LVLMs (Wu et al., 2024; Yin et al., 2023; Jing et al., 2023; Chen et al., 2024). Thus, the proposed methods (Gunjal et al., 2023) and (Wang et al., 2023) do not provide any information on which specific word of the respective sentence or subsentence is hallucinated. Similarly, (Wu et al., 2024; Yin et al., 2023; Jing et al., 2023; Chen et al., 2024) are based on LLM-generated atomic facts which neither allow for a token-level evaluation nor are reproducible. Moreover, the analysis in (Jing et al., 2023) shows that the LLM-based atomic fact extraction already induces errors propagating through the detection and evaluation pipeline. To overcome these issues, we rely on the automated and reproducible CHAIR evaluation method (see Sec. 4.2).

5 RESULTS

5.1 Hallucination Detection

In this section, we discuss the performance of our proposed method on four SOTA LVLMs. Tab. 5 summa-

Table 7: Experimental Results. Hallucination detection results on four SOTA LVLMs. Ours refers to the feature set $\mathcal{M}$. The best results in each block are highlighted.

		ACC (in %) ↑		*AUROC* (in %) ↑		*AUPRC* (in %) ↑	
MSCOCO (Lin et al., 2014)		LR	GB	LR	GB	LR	GB
InstructBLIP	*L*	$89.46^{(\pm 1.4e-1)}$	$89.46^{(\pm 1.3e-1)}$	$73.51^{(\pm 8.7e-1)}$	$73.16^{(\pm 9.0e-1)}$	$27.07^{(\pm 2.1e-0)}$	$25.6^{(\pm 2.5e-0)}$
	E	$89.49^{(\pm 1.6e-1)}$	$89.48^{(\pm 1.6e-1)}$	$65.49^{(\pm 1.3e-0)}$	$66.23^{(\pm 1.5e-0)}$	$15.38^{(\pm 5.7e-1)}$	$17.68^{(\pm 7.0e-1)}$
	Ours	$91.34^{(\pm 1.7e-1)}$	$\mathbf{91.49}^{(\pm 1.8e-1)}$	$\mathbf{89.93}^{(\pm 8.9e-1)}$	$\mathbf{89.93}^{(\pm 7.3e-1)}$	$56.07^{(\pm 1.2e-0)}$	$\mathbf{56.71}^{(\pm 7.6e-0)}$
mPLUG-Owl	*L*	$72.42^{(\pm 4.3e-1)}$	$72.48^{(\pm 4.6e-1)}$	$71.75^{(\pm 9.4e-1)}$	$71.86^{(\pm 9.3e-1)}$	$51.21^{(\pm 1.2e-0)}$	$50.65^{(\pm 1.1e-0)}$
	E	$70.06^{(\pm 4.9e-1)}$	$70.77^{(\pm 2.9e-1)}$	$66.01^{(\pm 6.3e-1)}$	$68.33^{(\pm 6.1e-1)}$	$40.09^{(\pm 8.2e-1)}$	$45.54^{(\pm 1.2e-0)}$
	Ours	$82.90^{(\pm 1.9e-1)}$	$\mathbf{83.26}^{(\pm 2.6e-1)}$	$88.41^{(\pm 3.9e-1)}$	$\mathbf{88.90}^{(\pm 2.8e-1)}$	$75.94^{(\pm 6.2e-1)}$	$\mathbf{77.04}^{(\pm 5.8e-1)}$
MiniGPT-4	*L*	$86.91^{(\pm 3.6e-1)}$	$86.85^{(\pm 3.9e-1)}$	$67.26^{(\pm 2.1e-0)}$	$67.01^{(\pm 2.1e-0)}$	$26.25^{(\pm 1.7e-0)}$	$25.41^{(\pm 1.2e-0)}$
	E	$86.84^{(\pm 3.6e-1)}$	$86.82^{(\pm 3.6e-1)}$	$60.78^{(\pm 1.8e-0)}$	$63.19^{(\pm 1.2e-0)}$	$15.77^{(\pm 6.7e-1)}$	$18.98^{(\pm 1.3e-0)}$
	Ours	$88.92^{(\pm 3.5e-1)}$	$\mathbf{89.27}^{(\pm 4.9e-1)}$	$88.16^{(\pm 1.5e-0)}$	$\mathbf{89.74}^{(\pm 1.3e-0)}$	$54.90^{(\pm 6.5e-0)}$	$\mathbf{57.25}^{(\pm 5.7e-0)}$
LLaVa	*L*	$81.57^{(\pm 1.4e-1)}$	$81.49^{(\pm 1.6e-1)}$	$70.53^{(\pm 8.7e-1)}$	$70.73^{(\pm 6.6e-1)}$	$37.53^{(\pm 2.0e-0)}$	$36.59^{(\pm 1.7e-0)}$
	E	$81.28^{(\pm 2.8e-1)}$	$81.26^{(\pm 2.9e-1)}$	$62.73^{(\pm 9.0e-1)}$	$64.63^{(\pm 7.7e-1)}$	$23.85^{(\pm 6.3e-1)}$	$27.52^{(\pm 4.6e-1)}$
	Ours	$87.25^{(\pm 2.0e-1)}$	$\mathbf{87.78}^{(\pm 3.0e-1)}$	$90.05^{(\pm 4.0e-1)}$	$\mathbf{91.01}^{(\pm 4.3e-1)}$	$70.15^{(\pm 1.0e-0)}$	$\mathbf{72.58}^{(\pm 1.3e-0)}$
BDD100K (Yu et al., 2020)		LR	GB	LR	GB	LR	GB
InstructBLIP	*L*	$77.54^{(\pm 3.3e-1)}$	$77.73^{(\pm 4.3e-1)}$	$63.30^{(\pm 8.2e-1)}$	$63.63^{(\pm 1.1e-0)}$	$31.80^{(\pm 1.1e-0)}$	$31.94^{(\pm 1.2e-0)}$
	E	$77.86^{(\pm 3.6e-1)}$	$77.86^{(\pm 3.7e-1)}$	$54.32^{(\pm 5.8e-1)}$	$56.71^{(\pm 6.3e-1)}$	$23.28^{(\pm 7.6e-1)}$	$26.06^{(\pm 8.9e-1)}$
	Ours	$84.07^{(\pm 2.0e-1)}$	$\mathbf{84.40}^{(\pm 3.1e-1)}$	$87.75^{(\pm 2.8e-1)}$	$\mathbf{88.78}^{(\pm 3.3e-1)}$	$66.04^{(\pm 1.6e-0)}$	$\mathbf{69.55}^{(\pm 2.0e-0)}$
mPLUG-Owl	*L*	$66.52^{(\pm 4.4e-1)}$	$66.25^{(\pm 5.3e-1)}$	$63.34^{(\pm 6.7e-1)}$	$63.40^{(\pm 6.4e-1)}$	$44.44^{(\pm 1.1e-0)}$	$43.86^{(\pm 7.5e-1)}$
	E	$66.42^{(\pm 4.8e-1)}$	$66.39^{(\pm 5.4e-1)}$	$57.83^{(\pm 4.8e-1)}$	$60.54^{(\pm 2.9e-1)}$	$37.20^{(\pm 5.4e-1)}$	$40.82^{(\pm 6.9e-1)}$
	Ours	$78.63^{(\pm 6.4e-1)}$	$\mathbf{79.85}^{(\pm 6.3e-1)}$	$85.06^{(\pm 6.2e-1)}$	$\mathbf{87.36}^{(\pm 6.7e-1)}$	$69.43^{(\pm 1.6e-0)}$	$\mathbf{74.79}^{(\pm 2.1e-0)}$
MiniGPT-4	*L*	$70.92^{(\pm 3.3e-1)}$	$71.08^{(\pm 2.1e-1)}$	$63.76^{(\pm 3.4e-1)}$	$63.44^{(\pm 4.4e-1)}$	$37.47^{(\pm 5.2e-1)}$	$37.18^{(\pm 2.1e-0)}$
	E	$71.30^{(\pm 2.7e-1)}$	$71.20^{(\pm 2.7e-1)}$	$64.04^{(\pm 5.4e-1)}$	$63.47^{(\pm 5.9e-1)}$	$38.03^{(\pm 5.0e-1)}$	$37.51^{(\pm 3.0e-1)}$
	Ours	$85.83^{(\pm 2.1e-1)}$	$\mathbf{86.05}^{(\pm 4.9e-1)}$	$90.75^{(\pm 5.1e-1)}$	$\mathbf{92.12}^{(\pm 4.2e-1)}$	$79.57^{(\pm 1.1e-0)}$	$\mathbf{84.01}^{(\pm 1.3e-0)}$
LLaVa	*L*	$69.92^{(\pm 2.4e-1)}$	$70.02^{(\pm 3.0e-1)}$	$61.25^{(\pm 9.4e-1)}$	$61.23^{(\pm 7.2e-1)}$	$37.29^{(\pm 1.2e-0)}$	$37.03^{(\pm 1.1e-0)}$
	E	$70.05^{(\pm 2.8e-1)}$	$70.03^{(\pm 2.8e-1)}$	$56.57^{(\pm 4.5e-1)}$	$57.88^{(\pm 6.2e-1)}$	$32.58^{(\pm 7.2e-1)}$	$34.65^{(\pm 7.6e-1)}$
	Ours	$80.93^{(\pm 4.6e-1)}$	$\mathbf{82.39}^{(\pm 3.3e-1)}$	$87.46^{(\pm 3.8e-1)}$	$\mathbf{89.56}^{(\pm 2.1e-1)}$	$70.39^{(\pm 1.5e-0)}$	$\mathbf{76.59}^{(\pm 8.0e-1)}$

rizes the computational time for calculating the input features (Sec. 3.2), training the meta model and the prediction (hallucination detection) time for the LR and GB meta model. As shown in Tab. 6, both models provide a calibrated classification between true and hallucinated objects reflected by a small expected calibration error (*ECE*) (Pakdaman Naeini et al., 2015). Tab. 7 summarizes our detection results. We achieve an *ACC* of up to 91.49%, *AUROC* values of up to 92.12%, and *AUPRC* values of up to 84.01% which clearly outperforms the TLC-A baselines L and E (Petryk et al., 2023) due to the additional information obtained from our features. More precisely, we outperform the TLC-A baseline by $14.85pp$ in terms of *ACC*, $28.65pp$ in terms of *AUROC* and $46.50pp$ in terms of *AUPRC*. A detailed analysis of our features including the TLC-A baselines L and E is provided in Sec. 5.2. While the GB classifier outperforms the linear model for our method in all experiments, this result does not hold for the one-dimensional baselines L and E. Especially for the log probability-based token-level confidence L, the linear model is superior to the GB classifier in most of the experiments.

Moreover, we observe better detection results with respect to *AUPRC* on the BDD100K dataset than on the MSCOCO data. This behavior is expected since the MSCOCO dataset is widely used as an instruction tuning dataset for pre-trained LVLMs leading to lower hallucination rates on MSCOCO (see Tab. 3). Thus, the LVLMs induce less positive (hallucinated) training samples when generating image captions, which makes the problem of learning the lightweight classifier f more challenging. Simultaneously, we achieve higher *ACC* values on the MSCOCO dataset indicating the insufficiency of the *ACC* as an evaluation metric for imbalanced datasets. While we state the performance of our method with respect to *ACC* for the sake of completeness, we emphasize the superior interpretability of the *AUROC* and *AUPRC* values for imbalanced datasets (Davis and Goadrich, 2006).

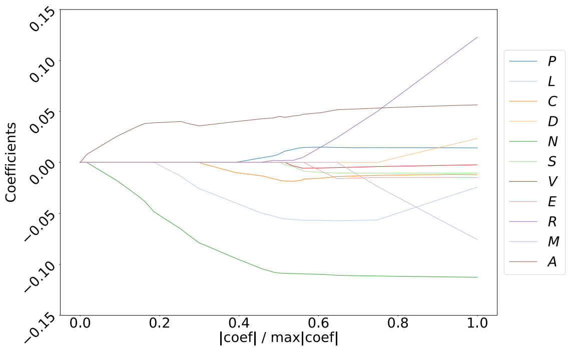

Figure 2: LASSO Path. LASSO path for $\mathcal{M}$. A denotes the maximum of the absolute values of all G weight coefficients for the attention features $A^g, g = 0, \dots, G-1$.

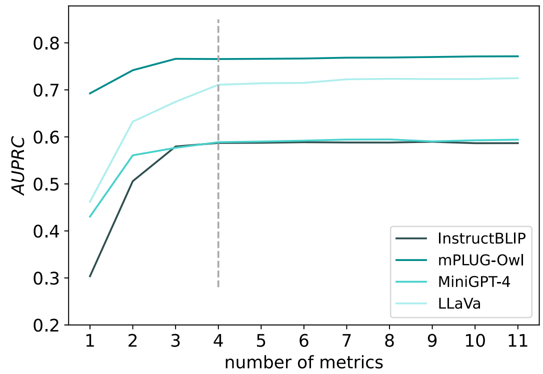

Figure 3: $AUPRC$ as a Function of the Number of Features. The classification performance of MetaToken in terms of $AUPRC$ as a function of the number of features for different LVLMs. The features are selected along the LASSO path of the respective LVLM.

Table 8: Feature Rank. The average rank of the features $\mathcal{M}$ in the LASSO paths of four SOTA art LVLMs.

avg. rank	feature	feature name
1.375	N	absolute occurrence (Eq. (2))
2.125	A	mean absolute attention (Eq. (3))
3.625	C	cumulated log probability (Eq. (5))
5.125	L	log probability (Eq. (4))
6.875	R	variation ratio (Eq. (9))
6.875	V	variance (Eq. (7))
7.250	E	entropy (Eq. (8))
7.625	D	probability difference (Eq. (11))
8.000	S	score (Eq. (6))
8.375	M	probability margin (Eq. (10))
8.750	P	relative position (Eq. (1))

5.2 Feature Analysis

In this section, we investigate the information contained in our proposed input features introduced in Sec. 3.2. We make use of the least absolute shrinkage and selection operator (LASSO) algorithm (Efron et al., 2004; Tibshirani, 2018) to analyze the predictive power of the input features considered. The LASSO method performs a variable selection for a linear regression including the estimation of the corresponding coefficients ranking the most informative features. For the attention features (Eq. (3)), we use the maximum of the absolute values of all G weight coefficients. Fig. 2 shows the LASSO path for mPLUG-Owl. Our proposed attention features $A^g, g = 0, \dots, G-1$ (Eq. (3)) are selected first, closely followed by the absolute occurrence N (Eq. (2)), the log probability L (Eq. (4)) as well as the cumulated log probability C (Eq. (5)). Moreover, the LASSO path indicates a minor relevance of the sequence score S (Eq. (6)) and the variance V (Eq. (7)) indicated by vanishing coefficients. We obtain similar results independently from the underlying LVLM or dataset. Tab. 8 lists the average rank of all features contained in the feature set $\mathcal{M}$ (Eq. (12)) during the LASSO selection with the TLC-A baselines L and E selected at position 4 and 7, respectively. This analysis underlines the improved information content from our features compared to the TLC-A baseline. Moreover, while most of the features are selected during the LASSO paths indicated through non-zero coefficients, Fig. 3 shows that four features are usually enough to achieve high $AUPRC$ values. Further features only add minor additional information to the classifier.

Finally, we refer to Fig. 4 to emphasize the importance of our statistical analysis based on the LASSO algorithm. While the relative position P (Eq. (1)) and probability margin M (Eq. (10)) might look like proper features to classify between hallucinated and true objects, our analysis shows that these features only add minor information to the classifier reflected by an average rank of 8.750 and 8.375, respectively (see Tab. 8).

5.3 MetaToken and Revision of Image Descriptions

In this section, we investigate MetaToken as a substitute for the LURE detection on the MSCOCO dataset. LURE (Zhou et al., 2023) serves as a hallucination mitigation method using a MiniGPT-4-based revisor to rectify image captions. To this end, LURE applies thresholds on the log probability L (Eq. (4)) and the relative position P (Eq. (1)) to detect possible object hallucinations and replaces them by the *I-don't-know* string "[IDK]". The resulting image caption and the input image are fed into the revisor afterwards to rectify the detected tokens. In our experiments, we replace the threshold-based LURE detection with our

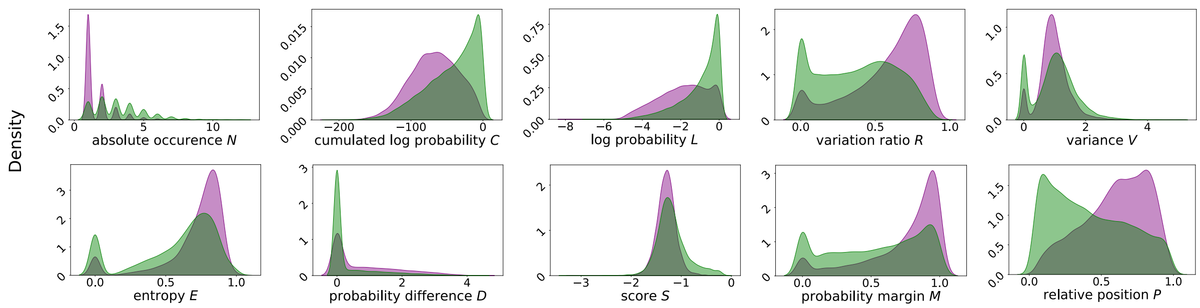

Figure 4: Features. Visualization of a selection of our input features defined in Sec. 3.2 for true and hallucinated objects.

proposed MetaToken method (see Sec. 3).

First of all note that as in (Zhao et al., 2024), we are not able to reproduce the results in (Zhou et al., 2023) with respect to InstructBLIP. While we achieve a hallucination rate of 10.4% for InstructBLIP captions (see Tab. 3), the rectified image captions by LURE include 10.9% hallucinations, even increasing the amount of hallucinated objects (see Tab. 9). We believe that this observation results from the fact that MiniGPT-4 has a higher hallucination rate than InstructBLIP (see Tab. 3). Since the LURE detection has a false positive rate (FPR) of 42.2% (see Tab. 9), i.e., detects true objects as hallucinations, we guess that the revisor based on MiniGPT-4 replaces true objects by hallucinated objects, even though the revisor is fine-tuned to mitigate hallucinations. However, applying our detection method, we are able to mitigate hallucinations achieving CHAIR$_i$ values of 9.8%. To this end, note that we can control the precision-recall-ratio in our method by varying the decision threshold of our lightweight meta classifier. For a recall of 70%, we observe a FPR of 9.4% only. Thus, we prevent the revisor from including additional hallucinations by replacing false positives, that is, true objects. The results in Tab. 9 confirm our assumption on InstructBLIP: The higher the FPR, the higher the number of hallucinations induced by the revisor.

For mPLUG-Owl, LURE reduced the number of hallucinations by 52.98%, i.e. from 30.2% to 14.2% with a FPR of 46.4%. Since the revisor induces substantially less hallucinations than mPLUG-Owl, the correction of true positives outweighs the potential introduction of new hallucinations by replacing false positives. In fact, we observe from Tab. 9 that higher recall values, and thus, higher FPRs, lead to consistently lower hallucination rates for mPLUG-Owl. Note that the superior precision-recall-ratio from our approach again outperforms the LURE results: For a recall of 80% (which is closest to the LURE detection of 78.6%), we reduce the proportion of hallucinations by 56.62% to 13.1%, that is, 3.64% less hallucinations compared to the LURE baseline.

Table 9: Integration of MetaToken into LURE. Results of MetaToken (Ours) plugged into the LURE mitigation method (Zhou et al., 2023) in %. The superscripts denote the hallucination recall values for the respective method. PR and FPR denote the hallucination precision and false positive rate, respectively. The best results are highlighted, the second best results are underlined.

	Method	CHAIR$_i$ ↓	CHAIR$_s$ ↓	PR ↑	FPR ↓
InstructBLIP	LURE$^{76.5}$	10.9	29.4	17.5	42.2
	Ours^{70}	**9.8**	**28.2**	**46.5**	**9.4**
	Ours^{80}	<u>10.3</u>	<u>28.5</u>	<u>37.0</u>	<u>15.9</u>
	Ours^{90}	10.6	30.3	26.3	29.5
	Ours^{100}	11.6	29.4	10.5	100
mPLUG-Owl	LURE$^{78.6}$	14.2	37.1	42.4	46.4
	Ours^{70}	14.4	36.3	**71.9**	**11.9**
	Ours^{80}	13.1	33.5	<u>64.8</u>	<u>18.8</u>
	Ours^{90}	<u>12.2</u>	<u>31.1</u>	55.1	31.6
	Ours^{100}	**12.1**	**30.2**	30.3	100

6 LIMITATIONS

Due to the lack of automated and reproducible token-level evaluation methods for attribute-level hallucinations, MetaToken is currently restricted to the problem of object hallucination detection, while the detection of attribute-level hallucinations remains an unsolved problem we will tackle in future work. Moreover, while the automated CHAIR method (Rohrbach et al., 2018) relies on ground truth labels, it still leads to mismatches due to misinterpretations of the generated language output of LVLMs.

7 CONCLUSION

In this paper, we introduce MetaToken, a novel lightweight hallucination detection technique for LVLMs based on meta classification. Inspired by recently discovered causes of hallucinations, we propose and analyze a broad set of potential factors for hallucinations in LVLMs. Based on a comprehensive statistical analysis of these factors, we reveal key indicators of hallucinations. We evaluate our method on four SOTA LVLMs achieving $AUROC$ values of up to 92.12% and $AUPRC$ values of up to 84.01%. Moreover, we show that our lightweight classifier detects hallucinations inducing an ECE between 0.81% and 2.01%. Finally, we demonstrate that MetaToken can be easily integrated into the LURE mitigation method reducing the hallucination rate by up to 56.62%, i.e., 3.64% less hallucinations than the LURE baseline. As future work, we will tackle the problem of attribute-level hallucination detection for general visual question answering tasks.

DISCLAIMER

The results, opinions and conclusions expressed in this publication are not necessarily those of Volkswagen Aktiengesellschaft.

REFERENCES

Anderson, P., Fernando, B., Johnson, M., and Gould, S. (2016). Spice: Semantic propositional image caption evaluation. In Leibe, B., Matas, J., Sebe, N., and Welling, M., editors, *Computer Vision – ECCV 2016*, pages 382–398, Cham. Springer International Publishing.

Chen, T., Navratil, J., Iyengar, V., and Shanmugam, K. (2019). Confidence scoring using whitebox meta-models with linear classifier probes. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 1467–1475. PMLR.

Chen, X., Wang, C., Xue, Y., Zhang, N., Yang, X., Li, Q., Shen, Y., Liang, L., Gu, J., and Chen, H. (2024). Unified hallucination detection for multimodal large language models. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3235–3252, Bangkok, Thailand. Association for Computational Linguistics.

Dai, W., Li, J., LI, D., Tiong, A., Zhao, J., Wang, W., Li, B., Fung, P. N., and Hoi, S. (2023a). Instructblip: Towards general-purpose vision-language models with instruction tuning. In Oh, A., Neumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., editors, *Advances in Neural Information Processing Systems*, volume 36, pages 49250–49267. Curran Associates, Inc.

Dai, W., Liu, Z., Ji, Z., Su, D., and Fung, P. (2023b). Plausible may not be faithful: Probing object hallucination in vision-language pre-training. In Vlachos, A. and Augenstein, I., editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2136–2148, Dubrovnik, Croatia. Association for Computational Linguistics.

Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 233–240, New York, NY, USA. Association for Computing Machinery.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *The Annals of Statistics*, 32(2):407–499.

Fieback, L., Dash, B., Spiegelberg, J., and Gottschalk, H. (2023). Temporal performance prediction for deep convolutional long short-term memory networks. In Ifrim, G., Tavenard, R., Bagnall, A., Schaefer, P., Malinowski, S., Guyet, T., and Lemaire, V., editors, *Advanced Analytics and Learning on Temporal Data*, pages 145–158, Cham. Springer Nature Switzerland.

Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Qiu, Z., Lin, W., Yang, J., Zheng, X., Li, K., Sun, X., and Ji, R. (2023). Mme: A comprehensive evaluation benchmark for multimodal large language models. *ArXiv*, abs/2306.13394.

Gao, H., Li, Y., Long, K., Yang, M., and Shen, Y. (2024). A survey for foundation models in autonomous driving. *ArXiv*, abs/2402.01105.

Gui, Y., Jin, Y., and Ren, Z. (2024). Conformal alignment: Knowing when to trust foundation models with guarantees. *ArXiv*.

Gunjal, A., Yin, J., and Bas, E. (2023). Detecting and preventing hallucinations in large vision language models. *ArXiv*, abs/2308.06394.

Hendrycks, D. and Gimpel, K. (2017). A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *Proceedings of International Conference on Learning Representations*.

Hessel, J., Holtzman, A., Forbes, M., Le Bras, R., and Choi, Y. (2021). Clipscore: A reference-free evaluation metric for image captioning. In Moens, M.-F., Huang, X., Specia, L., and Yih, S. W.-t., editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Holtzman, A., Buys, J., Du Li, Forbes, M., and Choi, Y. (2020). The curious case of neural text degeneration. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Huang, J., Zhang, J., Jiang, K., Qiu, H., and Lu, S. (2023). Visual instruction tuning towards general-purpose multimodal model: A survey. *ArXiv*, abs/2312.16602.

Jiang, Y., Omiye, J. A., Zakka, C., Moor, M., Gui, H., Alipour, S., Mousavi, S. S., Chen, J. H., Rajpurkar, P., and Daneshjou, R. (2024). Evaluating general vision-language models for clinical medicine. *medRxiv*.

Jing, L., Li, R., Chen, Y., Jia, M., and Du Xinya (2023). Faithscore: Evaluating hallucinations in large vision-language models. *ArXiv*, abs/2311.01477.

Kowol, K., Rottmann, M., Bracke, S., and Gottschalk, H. (2020). Yodar: Uncertainty-based sensor fusion for vehicle detection with camera and radar sensors. In *International Conference on Agents and Artificial Intelligence*.

Lavie, A. and Agarwal, A. (2007). Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 228–231, USA. Association for Computational Linguistics.

Li, C., Wong, C., Zhang, S., Usuyama, N., Liu, H., Yang, J., Naumann, T., Poon, H., and Gao, J. (2023a). Llava-med: Training a large language-and-vision assistant for biomedicine in one day. In Oh, A., Neumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., editors, *Advances in Neural Information Processing Systems*, volume 36, pages 28541–28564. Curran Associates, Inc.

Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, X., and Wen, J.-R. (2023b). Evaluating object hallucination in large vision-language models. In Bouamor, H., Pino, J., and Bali, K., editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305, Singapore. Association for Computational Linguistics.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T., editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.

Lin, W.-H. and Hauptmann, A. (2003). Meta-classification: Combining multimodal classifiers. In Zaïane, O. R., Simoff, S. J., and Djeraba, C., editors, *Mining Multimedia and Complex Data*, pages 217–231, Berlin, Heidelberg. Springer Berlin Heidelberg.

Liu, F., Guan, T., Wu, X., Li, Z., Chen, L., Yacoob, Y., Manocha, D., and Zhou, T. (2023). Hallusionbench: You see what you think? or you think what you see? an image-context reasoning benchmark challenging for gpt-4v(ision), llava-1.5, and other multi-modality models. *ArXiv*, abs/2310.14566.

Liu, F., Lin, K., Li, L., Wang, J., Yacoob, Y., and Wang, L. (2024a). Mitigating hallucination in large multi-modal models via robust instruction tuning. In *International Conference on Learning Representations*.

Liu, H., Xue, W., Chen, Y., Chen, D., Zhao, X., Wang, K., Hou, L., Li, R.-Z., and Peng, W. (2024b). A survey on hallucination in large vision-language models. *ArXiv*, abs/2402.00253.

Liu, T., Zhang, Y., Brockett, C., Mao, Y., Sui, Z., Chen, W., and Dolan, B. (2022). A token-level reference-free hallucination detection benchmark for free-form text generation. In Muresan, S., Nakov, P., and Villavicencio, A., editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6723–6737, Dublin, Ireland. Association for Computational Linguistics.

Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., Chen, K., and Lin, D. (2025). Mmbench: Is your multi-modal model an all-around player? In Leonardis, A., Ricci, E., Roth, S., Russakovsky, O., Sattler, T., and Varol, G., editors, *Computer Vision – ECCV 2024*, pages 216–233, Cham. Springer Nature Switzerland.

Lovenia, H., Dai, W., Cahyawijaya, S., Ji, Z., and Fung, P. (2024). Negative object presence evaluation (NOPE) to measure object hallucination in vision-language models. In Gu, J., Fu, T.-J. R., Hudson, D., Celikyilmaz, A., and Wang, W., editors, *Proceedings of the 3rd Workshop on Advances in Language and Vision Research (ALVR)*, pages 37–58, Bangkok, Thailand. Association for Computational Linguistics.

Lu, J., Yang, J., Batra, D., and Parikh, D. (2018). Neural baby talk. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7219–7228.

Maag, K., Rottmann, M., and Gottschalk, H. (2020). Time-dynamic estimates of the reliability of deep semantic segmentation networks. In *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 502–509.

Maag, K., Rottmann, M., Varghese, S., Hüger, F., Schlicht, P., and Gottschalk, H. (2021). Improving video instance segmentation by light-weight temporal uncertainty estimates. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Pakdaman Naeini, M., Cooper, G., and Hauskrecht, M. (2015). Obtaining well calibrated probabilities using bayesian binning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1).

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In Isabelle, P., Charniak, E., and Lin, D., editors, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Petryk, S., Whitehead, S., Gonzalez, J., Darrell, T., Rohrbach, A., and Rohrbach, M. (2023). Simple token-level confidence improves caption correctness. *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5730–5740.

Rohrbach, A., Hendricks, L. A., Burns, K., Darrell, T., and Saenko, K. (2018). Object hallucination in image captioning. In Riloff, E., Chiang, D., Hockenmaier, J., and Tsujii, J., editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4035–4045, Brussels, Belgium. Association for Computational Linguistics.

Rottmann, M., Colling, P., Paul Hack, T., Chan, R., Hüger, F., Schlicht, P., and Gottschalk, H. (2020). Prediction error meta classification in semantic segmentation: Detection via aggregated dispersion measures of softmax probabilities. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–9.

Rottmann, M. and Schubert, M. (2019). Uncertainty measures and prediction quality rating for the semantic segmentation of nested multi resolution street scene images. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1361–1369.

Schubert, M., Kahl, K., and Rottmann, M. (2021). Metadetect: Uncertainty quantification and prediction quality estimates for object detection. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423.

Tian, X., Gu, J., Li, B., Liu, Y., Hu, C., Wang, Y., Zhan, K., Jia, P., Lang, X., and Zhao, H. (2024). Drivevlm: The convergence of autonomous driving and large vision-language models. *ArXiv*, abs/2402.12289.

Tibshirani, R. (2018). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Vasudevan, V. T., Sethy, A., and Ghias, A. R. (2019). Towards better confidence estimation for neural models. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7335–7339.

Vedantam, R., Zitnick, C. L., and Parikh, D. (2015). Cider: Consensus-based image description evaluation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575.

Wang, J., Zhou, Y., Xu, G., Shi, P., Zhao, C., Xu, H., Ye, Q., Yan, M., Zhang, J., Zhu, J., Sang, J., and Tang, H. (2023). Evaluation and analysis of hallucination in large vision-language models. *ArXiv*, abs/2308.15126.

Wang, L., He, J., Li, S., Liu, N., and Lim, E.-P. (2024). Mitigating fine-grained hallucination by fine-tuning large vision-language models with caption rewrites. In Rudinac, S., Hanjalic, A., Liem, C., Worring, M., Jónsson, B. Þ., Liu, B., and Yamakata, Y., editors, *MultiMedia Modeling*, pages 32–45, Cham. Springer Nature Switzerland.

Wu, J., Liu, Q., Wang, D., Zhang, J., Wu, S., Wang, L., and Tan, T. (2024). Logical closed loop: Uncovering object hallucinations in large vision-language models. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Findings of the Association for Computational Linguistics ACL 2024*, pages 6944–6962, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Xing, S., Zhao, F., Wu, Z., An, T., Chen, W., Li, C., Zhang, J., and Dai, X. (2024). Efuf: Efficient fine-grained unlearning framework for mitigating hallucinations in multimodal large language models. *ArXiv*, abs/2402.09801.

Ye, Q., Xu, H., Xu, G., Ye, J., Yan, M., Zhou, Y., Wang, J., Hu, A., Shi, P., Shi, Y., Li, C., Xu, Y., Chen, H., Tian, J., Qi, Q., Zhang, J., and Huang, F. (2023). mplug-owl: Modularization empowers large language models with multimodality. *ArXiv*, abs/2304.14178.

Yin, S., Fu, C., Zhao, S., Xu, T., Wang, H., Sui, D., Shen, Y., Li, K., Sun, X., and Chen, E. (2023). Woodpecker: Hallucination correction for multimodal large language models. *ArXiv*, abs/2310.16045.

Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., and Darrell, T. (2020). Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yu, W., Yang, Z., Li, L., Wang, J., Lin, K., Liu, Z., Wang, X., and Wang, L. (2024). MM-vet: Evaluating large multimodal models for integrated capabilities. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F., editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 57730–57754. PMLR.

Zhao, L., Deng, Y., Zhang, W., and Gu, Q. (2024). Mitigating object hallucination in large vision-language models via classifier-free guidance. *ArXiv*, abs/2402.08680.

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*, abs/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*, abs/2304.10592.