TLDR: TOKEN-LEVEL DETECTIVE REWARD MODEL FOR LARGE VISION LANGUAGE MODELS

Deqing Fu[1,2] **Tong Xiao**[1] **Rui Wang**[1] **Wang Zhu**[1,2]
Pengchuan Zhang[1] **Guan Pang**[1] **Robin Jia**[2] **Lawrence Chen**[1]
[1]Meta [2]University of Southern California
deqingfu@usc.edu, lawrencechen@meta.com

ABSTRACT

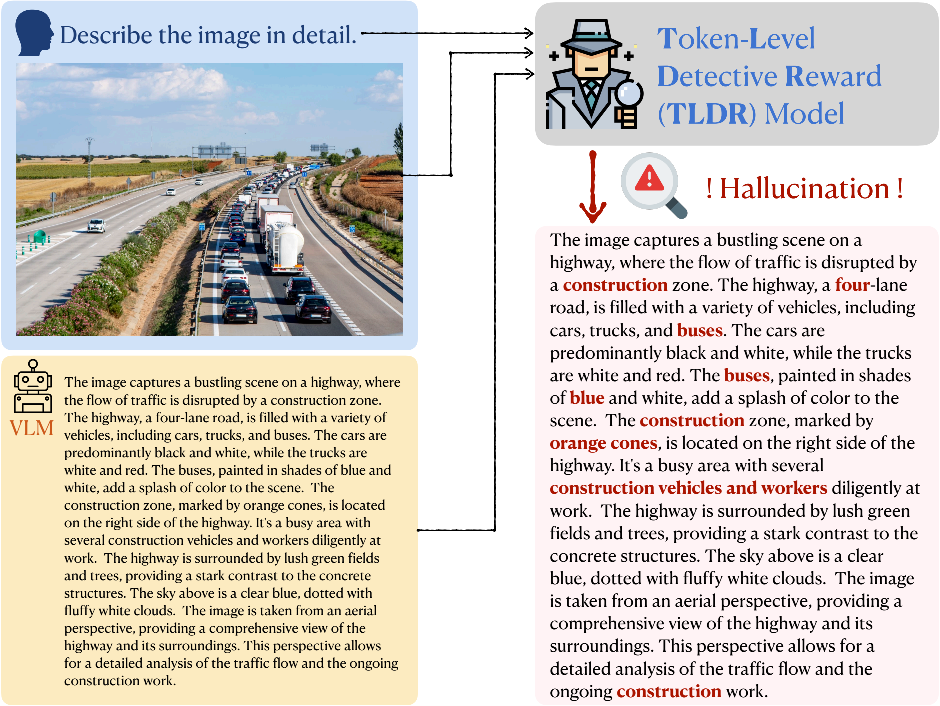
Although reward models have been successful in improving multimodal large language models, the reward models themselves remain brutal and contain minimal information. Notably, existing reward models only mimic human annotations by assigning only one binary feedback to any text, no matter how long the text is. In the realm of multimodal language models, where models are required to process both images and texts, a naive reward model may learn implicit biases toward texts and become less grounded in images. In this paper, we propose a **T**oken-**L**evel **D**etective **R**eward Model (**TLDR**) to provide fine-grained annotations to each text token. We first introduce a perturbation-based method to generate synthetic hard negatives and their token-level labels to train TLDR models. Then we show the rich usefulness of TLDR models both in assisting off-the-shelf models to self-correct their generations, and in serving as a hallucination evaluation tool. We show that TLDR automatically trains a token-level likelihood optimization, and can improve the base model's performance significantly. Finally, we show that TLDR models can significantly speed up human annotation by 3 times to acquire a broader range of high-quality vision language data.

1 INTRODUCTION

Large vision language models (VLMs) are becoming increasingly powerful in generating human-like text, exemplified by models like GPT-4 family (OpenAI, 2024), Gemini and PaliGemma (Google, 2023; Beyer et al., 2024), LLaVA model family (Liu et al., 2024), and Llama 3 Vision models (Meta, 2024a). However, they are far from perfect and still suffer from generating hallucinated texts that are not grounded to the reference image. The need for accurate and interpretable reward models (RMs) to highlight the mistakes becomes increasingly critical. Traditional RMs, which are often binary classification models to provide one single score to evaluate entire outputs, have limitations in terms of interpretability and granularity. These models obscure the decision-making process of the model, making it challenging for humans to diagnose and improve performance at a fine-grained level.

To facilitate better interpretability and granularity, we propose a **T**oken-**L**evel **D**etective **R**eward (**TLDR**) model to offer a more interpretable alternative. By evaluating and assigning rewards at each token, rather than across entire sequences, TLDR enables greater transparency. This fine-grained approach allows for clearer identification of where a model excels in its output generation. Such interpretability is crucial not only for aligning model behavior with human expectations but also for improving human-AI interaction – a human annotator can swiftly fix the highlighted errors given by TLDR to make them correct because token-level evaluations allow for quicker identification of errors and more targeted improvements.

Additionally, a naive binary reward model could be biased towards text modalities – the longer the text, the higher the score, despite any internal hallucinations, making them less effective in multimodal contexts where visual information is essential. Our work aims to address this by constructing a reward model that is more visually grounded, incorporating multimodal cues to better evaluate model performance. The interpretability afforded by token-level granularity helps facilitate this grounding, ensuring that visual and textual signals are both considered in reward calculations. Ab-


Describe the image in detail.
Token-Level Detective Reward (TLDR) Model
! Hallucination !
VLM

The image captures a bustling scene on a highway, where the flow of traffic is disrupted by a construction zone. The highway, a four-lane road, is filled with a variety of vehicles, including cars, trucks, and buses. The cars are predominantly black and white, while the trucks are white and red. The buses, painted in shades of blue and white, add a splash of color to the scene. The construction zone, marked by orange cones, is located on the right side of the highway. It's a busy area with several construction vehicles and workers diligently at work. The highway is surrounded by lush green fields and trees, providing a stark contrast to the concrete structures. The sky above is a clear blue, dotted with fluffy white clouds. The image is taken from an aerial perspective, providing a comprehensive view of the highway and its surroundings. This perspective allows for a detailed analysis of the traffic flow and the ongoing construction work.

The image captures a bustling scene on a highway, where the flow of traffic is disrupted by a **construction** zone. The highway, a **four**-lane road, is filled with a variety of vehicles, including cars, trucks, and **buses**. The cars are predominantly black and white, while the trucks are white and red. The **buses**, painted in shades of **blue** and white, add a splash of color to the scene. The **construction** zone, marked by **orange cones**, is located on the right side of the highway. It's a busy area with several **construction vehicles and workers** diligently at work. The highway is surrounded by lush green fields and trees, providing a stark contrast to the concrete structures. The sky above is a clear blue, dotted with fluffy white clouds. The image is taken from an aerial perspective, providing a comprehensive view of the highway and its surroundings. This perspective allows for a detailed analysis of the traffic flow and the ongoing **construction** work.

Figure 1: **T**oken-**L**evel **D**etective **R**eward (***TLDR***) Model. It can be used as hallucination detection, and to prompt models to self-correct with these detection. TLDR can also speed up human annotation speed to fix slightly mistaken image captions, to create high-quality vision language data.

lation studies in training TLDR models shown in Tables 10 and 11 verifies the claim by showing the sharp improvement of the RM's performance by further finetuning the linear projection of the VLM — projecting visual features given by the vision encoder to the textual embedding space. Unlike existing token-level RL methods that primarily serve as auxiliary post-training evaluation tools, TLDR is designed for on-policy RLHF training in vision-language models, bridging the gap between fine-grained supervision and model optimization.

Moreover, token-level reward models have the potential to enhance existing methods for model improvement. By providing detailed feedback on a token-by-token basis, these models enable more effective self-correction and refinement in generated outputs. A more granular understanding of errors can improve the performance of fine-tuning techniques such as DPO (Rafailov et al., 2024) and PPO (Schulman et al., 2017), where strong and interpretable reward signals are essential for optimizing model behavior. In Section 5.4, we show the TLDR model is automatically a likelihood training objective, that simultaneously improve the base vision language model behind the RM.

In a summary, TLDR model aims to develop a reward model that not only reflects human preferences more accurately but also enhances usability and interpretability. By improving the transparency of the reward mechanism at the token level, we provide a tool that facilitates faster feedback, better self-correction, simple likelihood finetuning, and a trustworthy hallucination evaluation metric. Our work establishes the first token-level reward model specifically tailored for vision-language models, enabling both hallucination detection and self-correction, while also naturally integrating into reinforcement learning frameworks. By unifying token-level reward modeling with vision-language tasks, TLDR significantly improves grounding capabilities and reduces hallucination rates.

2 Related Work

Reinforcement Learning from Human Feedback and Reward Model. Using Reinforcement Learning to align lan- guage models with human feedback or preferences (RLHF, Christiano et al., 2017; Ziegler et al., 2020) has led to phenomenal improved large language models such as ChatGPT (Ouyang et al., 2022) and LLaMA 3 (Meta, 2024a). Similar RLHF techniques are used to align large vision language models Liu et al. (2023) and text-to-image models (Lee et al., 2023; Sun et al., 2023a) as well. In general, RLHF involves training a **reward model** on user preference data collected from human annotators (Wang et al., 2024a). Given a reward model, a policy can be learned using Reinforcement learnings algorithms like Proximal Policy Optimization (PPO, Schulman et al., 2017). Alternatively, recent works have developed Direct Policy optimization (DPO, Rafailov et al., 2024) wherein reward models are mainly used for finding chosen and rejected pairs. DPO alignment for vision-language models is also well studied (Yu et al., 2023b; 2024; Li et al., 2024; Pi et al., 2024). Reward models themselves are also evolving including process reward models (Luo et al., 2023), step-wise reward models (Havrilla et al., 2024), etc. Recent works also attempt span-level or token-level detection but they are limited to the language domain (Yoon et al., 2024; Yang et al., 2024), and they are mostly sentence-level (Wu et al., 2023; Niu et al., 2024; Mishra et al., 2024) or need factual augmentations (Sun et al., 2023b). Unlike previous token-level reward works (Yoon et al., 2024) on for *offline RLHF*, we propose the first unified token-level reward model, TLDR, which establishes the stage for vision-language *on-policy RLHF training with token-level reward*. TLDR facilitates image-to-text hallucination detection (Rohrbach et al., 2018; Li et al., 2023; Jing et al., 2024; Lovenia et al., 2024) with efficient human correction, and improves downstream vision-language grounding performance.

Synthetic Data and Hard Negative Mining. Several NLP datasets have gathered instances of the negative class for their task. Many relied on human annotation, for example, unsupported claims in fact verification (Aly et al., 2021; Wadden et al., 2020), non-entailed hypotheses in NLI (Bowman et al., 2015), unanswerable questions (Rajpurkar et al., 2018). Some have used heuristics and external knowledge sources to automatically mine negative examples (Lee et al., 2021; Wright et al., 2022). Finally, there are hybrid approaches where candidate negative examples are first automatically generated and then manually verified (Wadden et al., 2022), or candidate negative examples are synthesized by model perturbation and verified by the same model (Fu et al., 2023).

Large Vision Language Models and Evaluation. There has been a plethora of recent developments in VLMs; they can be broadly categorized by their methods for representing visual modalities. A representative approach involves *tokenizing* visual inputs to be jointly trained with language inputs (Yu et al., 2023a; Google, 2023; Chameleon, 2024). Another line of work processes continuous visual features by directly projecting them to the language embedding space via a learnable function (Liu et al., 2024; Bavishi et al., 2023). At the core of these design choices is the hardness in representing visual features, which has been reported by several early studies (McKinzie et al., 2024) to be the key bottleneck towards better vision-language foundation models. Various benchmark datasets beyond MMMU (Yue et al., 2024) were proposed targeting these bottlenecks, such as BLINK (Fu et al., 2024b) and Vibe-Eval (Padlewski et al., 2024) for visual reasoning, IsoBench (Fu et al., 2024a) and MathVista (Lu et al., 2024) for algorithmic visual problem solving. At the essence of common mistakes made by VLMs, hallucination is a significant portion, and thus TLDR is designed as an evaluation tool to measure models' hallucination rate (see Table 2).

3 Problem Setup

A multimodal query-response instance $x = (m, p, d)$ is usually equipped with three elements, an image m, a user text prompt p, and a text response d. Training a reward model involves training a classifier $\rho(m, p, d) \in \{0, 1\}$ to predict human preference on the target response d given the image m and prompt p. In contrast, to have better granularity for the reward model, a token-level version is needed. Instead of training a point-wise scalar classifier, which only assigns a singular value to the instance x, it assigns values for every token of the target response $d = \{e_1, \cdots, e_N\}$ with $N = |d|$ tokens in total, where $|\cdot|$ denotes the number of tokens of any text sequence. It involves training the TLDR model $\mathbb{P}_\gamma$ to match fine-grained rewards. TLDR model's prediction can be written as

$$\gamma(m, p, d) = \Big(\gamma(e_1 \mid m, p, d), \cdots, \gamma(e_N \mid m, p, d)\Big) \in [0, 1]^N \tag{1}$$

where for any token e in text response d, $\gamma(e \mid m, p, d) = \begin{cases} 1, & \text{if } \mathbb{P}_\gamma(e \mid m_k, p_k, d_k) > \theta \\ 0, & \text{otherwise} \end{cases}$, with some threshold θ usually set to 0.5 if not otherwise mentioned.

Given the image, prompt, and response tuples (m_k, p_k, d_k) in an evaluation set $\mathcal{S}$, we design three accuracy metrics on the TLDR model's performance.

Token-Level Accuracy. For each instance $x_k = (m_k, p_k, d_k)$, we are given true token-level labels $\gamma^\star(m_k, p_k, d_k) = \Big(\gamma^\star(e_1), \cdots, \gamma^\star(e_N)\Big)$. Then we define token-level accuracy as

$$\mathcal{A}_{\mathrm{T}}(\gamma, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{(m_k, p_k, d_k) \in \mathcal{S}} \frac{1}{|d_k|} \sum_{e \in d_k} \mathbb{1}\left\{\gamma^\star(e) = \gamma(e \mid m_k, p_k, d_k)\right\} \tag{2}$$

Sentence-Level Accuracy. Similar to the token-level accuracy but with each response d broken into sentences $d = \{s_1, \cdots, s_{c(d)}\}$ with $s_j = \{e_{n_{j-1}+1}, \cdots, e_{n_j}\}$ where n_j is the token position of the j-th period. We also let $n_0 = 0$ as the starting position. We also $c(\cdot)$ as the function of counting number of sentences in any text response d. Then we define sentence-level accuracy as

$$\mathcal{A}_{\mathrm{S}}(\gamma, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{(m_k, p_k, d_k) \in \mathcal{S}} \frac{1}{c(d_k)} \sum_{j \in 1, \cdots, c(d_k)} \mathbb{1}\left\{\prod_{e \in s_j} \gamma^\star(e) = \prod_{e \in s_j} \gamma(e \mid m_k, p_k, d_k)\right\} \tag{3}$$

A visual illustration on grouping tokens into sentences is shown in Figure 3.

Response-Level Accuracy. We define response level prediction as

$$\rho_\gamma(m, p, d) = \prod_{e \in d} \gamma(e \mid m, p, d) \quad \text{and} \quad \rho^\star(m, p, d) = \prod_{e \in d} \gamma^\star(e) \tag{4}$$

Then we compare with the ground truth labels $\rho^\star$ to define response-level accuracy as

$$\mathcal{A}_R(\rho_\gamma, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{(m_k, p_k, d_k) \in \mathcal{S}} \mathbb{1}\left\{\rho_\gamma(m_k, p_k, d_k) = \rho^\star(m_k, p_k, d_k)\right\} \tag{5}$$

Notably, the response-level accuracy also applies to response-level naive reward models ρ under the same definition. We will further compare the response-level accuracy of TLDR model $\mathcal{A}_R(\rho_\gamma, \mathcal{S})$ and that of the naive model $\mathcal{A}_R(\rho, \mathcal{S})$ in Section 5.

Besides accuracy metrics, which could be biased when labels are imbalanced, especially in the token-level cases where a majority of the tokens are neutral tokens – words that won't affect response quality, we report **mean Average Precision** (mAP). Together with normal mAP metrics, for which we call mAP(pos) we also design a flipped version mAP(neg), where both the predicted labels and the ground-truth labels are flipped so that we pay more attention to tokens with negative labels. Because the token-level annotations are highly imbalanced, with more than 95% of them are positive tokens (see the example in Figure 1), the mAP(neg) is a more meaningful average precision metric. We report both mAP scores in the tuple format **mAP(neg|pos)**.

4 Synthetic Data Generation

Although aligning models toward user preference has become a standard post-training procedure, open-sourced user preference data, especially multimodal ones, are increasingly difficult to source. What is even worse, existing user preference data are mostly coarsely annotated as each instance is only given one label: thumb down or thumb up. To gather large amounts of fine-grained token-level preference data, we adopt the procedure by perturbing gold labels, inspired by Fu et al. (2023).

In this work, we mainly focus on two types of tasks: dense captioning and visual question answering (VQA). For VQA data, we synthesize hard negatives from **Visual Genome (VG100K)** dataset (Krishna et al., 2016), which contains 108,077 images with over 1.7 million question-answer pairs. For any VQA instance x with image m, question p and answer d, we prompt a pretrained large language model ϕ, which takes the original question p and answer d, and is instructed to generate a perturbed answer $d' = \phi(p, d)$ so that it's the *wrong* answer to the question p given the image m, i.e., $\mathbb{P}(d' \mid m, p) = 0$. Notably, the model used for perturbation ϕ is text-only, without seeing the image m to mitigate any visual biases. In this work, we use `Llama-3.1-70B` (Meta, 2024a) as the perturbation model ϕ.

Admittedly, as dense captions are relatively longer than VQA samples, the amount of data available is more limited, and the hard negative synthesis process could be more versatile and more

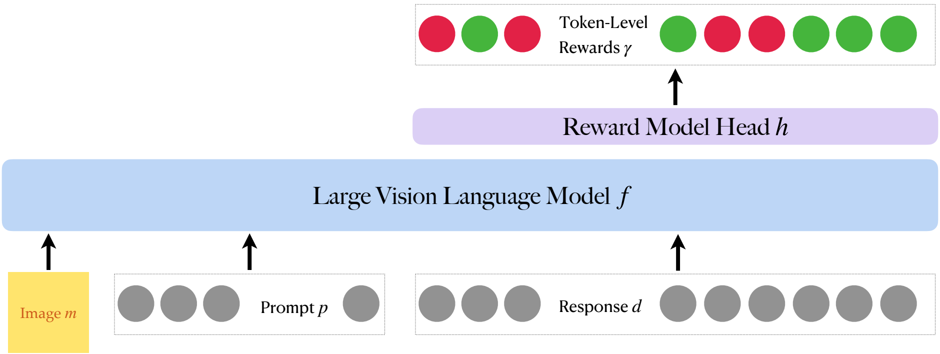


Figure 2: **TLDR Model Architecture.** For any instance with image m, prompt p, and a response d, they are passed altogether into the large VLM backbone f without the language model head ℓ. Then a shared reward model head h is applied to every token e_k of the response d to have binary predictions $\gamma(e_k)$ to determine if e_k is a good token or a bad token.

complicated. We synthesize hard negatives from **DOCCI** dataset (Onoe et al., 2024), which contains over 15,000 images and their corresponding dense captions. However, the amount of captions here is much fewer than the amount of VQA instances. To compensate the imbalance, we use `Llama-3.1-70B` to aggregate VQA instances to dense captions. For each image m in the VG100K dataset, it's equipped with on average 16 question-answer pairs $\{(p_1, d_1), \cdots, (p_k, d_k)\}$. The text-only LLM is prompted to combine them into a dense caption d for image m. Now, combining DOCCI and VG100K's synthetic caption, we have over 120,000 image-caption pairs. As prior work identifies (Lin et al., 2024), vision language models usually suffer in the following eight taxonomies. We enumerate each of them with an illustrated example pair.

I. SPATIAL RELATIONSHIP: A is **left** to B ⟷ A is **right** to B.
II. VISUAL ATTRIBUTE: A is **yellow**. ⟷ A is **blue**.
III. ATTRIBUTE BINDING: A is **blue** and B is **yellow**. ⟷ A is **yellow** and B is **blue**.
IV. OBJECT IDENTIFICATION: A **dog** chasing a ball. ⟷ A **cat** chasing a ball.
V. COUNTING: **One** duck is swimming. ⟷ **Four** ducks are swimming.
VI. SMALL OBJECT: **Cirrostratus cloud** in the sky. ⟷ **Clear** sky.
VII. TEXT OCR: A shirt writes **heavy fog**. ⟷ A shirt writes **happy frog**.
VIII. COUNTERFACTUAL: A soldier. ⟷ A soldier **has no sword in hand**.

For each taxonomy t, and for each instance with image m and caption d, we use a prompt-engineered text-only LLM ϕ_t to generate a perturbed caption $d' = \phi_t(d)$ so that d' is a minimal-edit from d. Furthermore, we prompt-engineer another LLM $\phi_c(d, d', t)$ to check they are not paraphrases and their difference lies in the desired taxonomy t. If they fail $\phi_c(d, d', t)$, we discard the perturbation. For instance, not every image has text written in it, so there is no way to generate perturbations focused on text OCR.

For either VQA or dense caption tasks, once we obtain successful perturbation $d' = \{e'_1, \cdots, e'_{|d'|}\}$, we compute the differences to the original text $d = \{e_1, \cdots, e_{|d|}\}$ to obtain the token-level label $\gamma^*(e'_k) \in \{0, 1\}$ depending on whether e'_k appear in the neighborhood of e_k in d or not. Since the original text d is human written, all of its tokens have positive label $\gamma^*(e_k) = 1, \forall e_k \in d$.

We include prompts for perturbation in Appendices A.1 to A.3 and the statistics of our synthetic data in Table 7 at Appendix A.4.

5 EXPERIMENTS

5.1 TRAINING TLDR MODELS

Model Architecture. As shown in Figure 2, we use `PaliGemma-3B-Mix-448` (Beyer et al., 2024) and `Llama-3.2-11B-Vision` (Meta, 2024b) as our backbone pretrained large Vision

Base Model	Reward Model	Token-Level Accuracy $\mathcal{A}_T$	Sentence-Level Accuracy $\mathcal{A}_S$	Response-Level Accuracy $\mathcal{A}_R$	mAP (neg|pos)
`PaliGemma-3B`	TLDR	98.6	86.5	**83.1**	(41.3|99.8)
	Naive	—	—	81.1	—
`Llama-3.2-11B-Vision`	TLDR	98.9	90.8	**88.2**	(45.7|99.8)
	Naive	—	—	86.7	—
`GPT-4o`	Prompting	95.5	66.9	52.9	(19.7|98.1)
`Random Guess`	—	—	—	*50.0*	—

Table 1: **Performance of the TLDR model.** As a reference, we include scores for the response-level naive reward model trained on the same dataset. Best response-level accuracy are highlighted in **bold** conditioning on the same base model. We find that the TLDR model has a slightly higher response-level accuracy when compared to the naive binary RM. A break down of response-level accuracy by taxonomy is shown in Table 10. Having a base model with larger number parameters also gives higher accuracies across all granularity.

Language Model f. Instead of using the pretrained language modeling head ℓ which maps the last hidden states to the vocabulary logits, we train a new *reward model head* $h : \mathbb{R}^{D_{\text{hidden}}} \to \mathbb{R}$ to map the last hidden states with dimension D_{hidden} to a scalar logit for each token from the response d. For any instance equipped with image m, prompt p and response d, we denote $|\cdot|$ as the number of tokens after tokenization for either image or text modality. The backbone language model gives the last hidden states $\mathbf{H} = f(m,p,d) \in \mathbb{R}^{(|m|+|p|+|d|)\times D_{\text{hidden}}}$ and the probability of being positive for k-th token e_k in the response d is given by

$$\mathbb{P}_\gamma(e_k \mid m,p,d) = \sigma\left(h\left(\mathbf{H}_{\star,(|m|+|p|+k)}\right)\right), \quad \text{where } \sigma \text{ is the Sigmoid function.} \tag{6}$$

In our setup, the reward model head h is a simply linear layer with $(D_{\text{hidden}} + 1)$ parameters. We provide more detailed training setups and hyperparameters in Appendix B.

Training. As the PaliGemma report (Beyer et al., 2024) suggests, PaliGemma used full attention between input images and input texts, and only has autoregressive attention at generation. Similar to LLaVA model family (Liu et al., 2024), PaliGemma model f has four major components: a 400M SigLIP (Zhai et al., 2023) vision encoder f_{enc}, a linear projection module f_{proj} to align the vision features to the proper text embedding spacing, a Gemma-2B (Google, 2024) Transformer decoder f_{dec}, and a language model head ℓ. We train our randomly initialized reward model head h, together with f_{proj} and f_{dec}. For efficient training, we use LoRA (Hu et al., 2021) technique to update weights Θ_{proj} of f_{proj} and Θ_{dec} of f_{dec}, so that $\Theta' = \Theta + \alpha_{\text{train}}AB$. We choose $\alpha_{\text{train}} = 128$ and $r := \text{rank}(A) = \text{rank}(B) = 512$ for all submodules Θ. Models are trained with respect to the cross-entropy objective on every token of the response d, compared to the token-level label generated following Section 4.

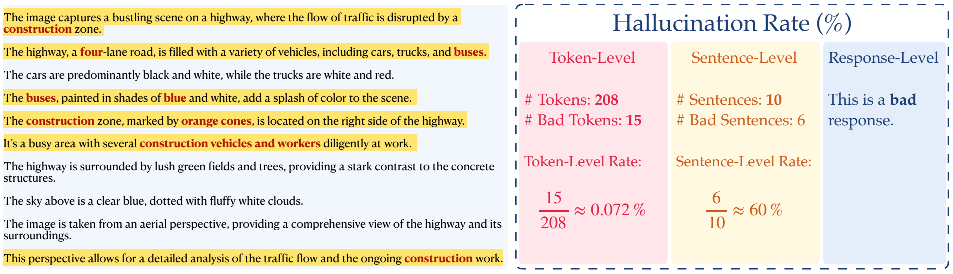
In contrast, we compare with a naive reward model trained on the same training data and with the same hyperparameters. Although sharing the same architecture and parameter count, the naive reward model differs from the TLDR model as its cross-entroy loss is only computed at the last token of each response d, instead of on all tokens.

Evaluation. We evaluate TLDR model's performance on the synthetic data generated from the test split of the DOCCI dataset (Onoe et al., 2024). We measure the performance based on the metrics discussed in Section 3. As shown in Table 1, the TLDR model has slightly higher response-level accuracy than the naive binary RM. TLDR model has a 41.3 mAP(neg) and signals further room for improvements. A break-down of response-level taxonomy in Table 10 at Appendix B shows that, TLDR model performs the worst one spatial relationship taxonomy, and this resonances prior work that image grounding to spatial relationship is one of the hardest task for both image-to-text VLMs and text-to-image generations (Lin et al., 2024).

We conduct further human evaluation on token-level predictions on 100 samples from WinoGround (Thrush et al., 2022) images with captions generated by `MiniCPM` , `Phi-Vision-3.5` and `Qwen2-VL-7B`. With a special focus on false negative (FN) type of errors and averaged among three human annotators, we find the TLDR model has a sentence-level FN rate of 8.7% , 10.5% and 9.8%, respectively.

Models	Hallucination Rate (%) ↓			MMMU ↑	MEGA-Bench ↑
	Token-Level	Sentence-Level	Response-Level		
GPT-4o	**0.016**	0.23	1.62	**69.1**	**54.1**
Llama-3.2-90B-Vision	0.017	**0.19**	**1.23**	60.3	/
GPT-4o-mini	0.030	0.38	2.12	59.4	43.0
GPT-4-Turbo-Vision	0.033	0.62	3.12	56.8	/
Qwen2-VL-7B	0.061	0.48	1.96	54.1	35.9
Qwen2-VL-2B	0.066	0.72	1.70	41.1	22.3
MiniCPM-Llama-3-V2.5	0.067	0.81	3.62	45.8	22.8
Llama-3.2-11B-Vision	0.073	0.85	1.88	50.7	18.0
Phi-Vision-3.5-Instruct	0.261	2.65	9.25	43.0	25.3
PaliGemma-3B-Mix-448	4.444	5.96	17.50	27.3	/

Table 2: **Hallucination Evaluation.** We prompt each model with image captioning instructions on 800 images from WinoGround (Thrush et al., 2022), and use TLDR to compute the hallucination rates (the lower the better), with respect to various levels of granularity. Model performances are sorted by token-level rate. We observe that GPT-4o is overall the best model with the least token-level hallucinations but Llama-3.2-90B-Vision has better sentence-level and response-level hallucinations. We also include self-reported MMMU (Yue et al., 2024) results to demonstrate their significant correlation with hallucination rates: the Pearson correlation between $-\log \mathcal{H}_T$ and MMMU score is 0.902 with a p-value of 3.45×10^{-4}. The correlation can also be visually observed in Figure 6.



The image captures a bustling scene on a highway, where the flow of traffic is disrupted by a **construction** zone.

The highway, a **four**-lane road, is filled with a variety of vehicles, including cars, trucks, and **buses**.

The cars are predominantly black and white, while the trucks are white and red.

The **buses**, painted in shades of **blue** and white, add a splash of color to the scene.

The **construction** zone, marked by **orange cones**, is located on the right side of the highway.

It's a busy area with several **construction vehicles and workers** diligently at work.

The highway is surrounded by lush green fields and trees, providing a stark contrast to the concrete structures.

The sky above is a clear blue, dotted with fluffy white clouds.

The image is taken from an aerial perspective, providing a comprehensive view of the highway and its surroundings.

This perspective allows for a detailed analysis of the traffic flow and the ongoing **construction** work.

Figure 3: **Level of Granularity in Hallucination Rate.** Using the example from Figure 1, we can easily compute token-level hallucination rates following Eq. 7. Then tokens are grouped into sentences which are separated by period marks. An entire sentence with at least one bad token is highlighted as a bad sentence. Then the sentence-level hallucination rate of one response is calculated by counting the proportion of bad sentences. Similarly, if there is at least one bad token in the response, the entire response is a bad one. Hallucination rates are averaged over an entire evaluation set to determine the overall hallucination rates of a model.

Ablation Study. As discussed earlier, we finetune a randomly initialized reward model head h, together with LoRA efficiently finetuning the multimodal projection layer f_{proj} and the Transformer decoder f_{dec}. In the section, we ablate the necessity of LoRA finetuning f_{proj} and f_{dec}.

As shown in Table 11 at Appendix Appendix B.2, both linear projection module and the decoder module are worth training and they work together to facilitate the performance of our TLDR model. One interesting observation is training f_{proj} on top of f_{dec} barely improves token-level accuracy or mAP(pos). But it helps with both sentence-level and response-level accuracy, and increases mAP(neg) significantly. This could be explained by that finetuning f_{proj} reduces the model's false negative rates on tokens because it's more visually grounded by tuning the projection from visual space to textual space.

5.2 Hallucination Evaluation with TLDR Models

Since our TLDR model provides token-level predictions, we can use it to compute a model's hallucination rates without requiring ground truth labels. Given image m, prompt p, and a model for evaluation ξ. We first obtain model's response $\hat{d} = \xi(m, p)$ with tokens $\{\hat{e}_1, \cdots, \hat{e}_{|\hat{d}|}\}$. Then our TLDR model gives its prediction $\gamma(m, p, \hat{d}) = \left(\gamma(\hat{e}_1 \mid m, p, \hat{d}), \cdots, \gamma(\hat{e}_{|\hat{d}|} \mid m, p, \hat{d})\right)$. Then the

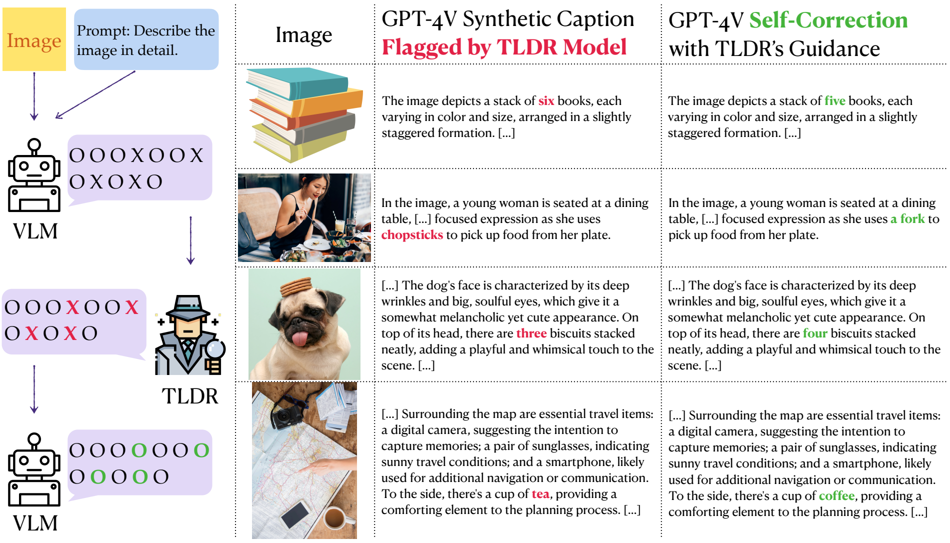


Image	GPT-4V Synthetic Caption **Flagged by TLDR Model**	GPT-4V **Self-Correction** with TLDR's Guidance
	The image depicts a stack of **six** books, each varying in color and size, arranged in a slightly staggered formation. [...]	The image depicts a stack of **five** books, each varying in color and size, arranged in a slightly staggered formation. [...]
	In the image, a young woman is seated at a dining table, [...] focused expression as she uses **chopsticks** to pick up food from her plate.	In the image, a young woman is seated at a dining table, [...] focused expression as she uses **a fork** to pick up food from her plate.
	[...] The dog's face is characterized by its deep wrinkles and big, soulful eyes, which give it a somewhat melancholic yet cute appearance. On top of its head, there are **three** biscuits stacked neatly, adding a playful and whimsical touch to the scene. [...]	[...] The dog's face is characterized by its deep wrinkles and big, soulful eyes, which give it a somewhat melancholic yet cute appearance. On top of its head, there are **four** biscuits stacked neatly, adding a playful and whimsical touch to the scene. [...]
	[...] Surrounding the map are essential travel items: a digital camera, suggesting the intention to capture memories; a pair of sunglasses, indicating sunny travel conditions; and a smartphone, likely used for additional navigation or communication. To the side, there's a cup of **tea**, providing a comforting element to the planning process. [...]	[...] Surrounding the map are essential travel items: a digital camera, suggesting the intention to capture memories; a pair of sunglasses, indicating sunny travel conditions; and a smartphone, likely used for additional navigation or communication. To the side, there's a cup of **coffee**, providing a comforting element to the planning process. [...]

Figure 4: TLDR model can guidance existing VLMs to **Self-Correct** their hallucination when generating captions for images from WinoGround (Thrush et al., 2022).

token-level hallucination rate for this instance is $\frac{1}{|\hat{d}|}\sum_{\hat{e}_k\in\hat{d}}\gamma(\hat{e}_k \mid m, p, \hat{d})$. Similar to sentence-level and response-level accuracy defined in Section 3, we can have sentence-level and response-level hallucination rates as well. Their definitions are shown in Figure 3.

Now we can have the token-level hallucination rates for the model ξ given any dataset $\mathcal{S}$ as follows

$$\mathcal{H}_T(\xi, \mathcal{S}) = \frac{1}{|\mathcal{S}|}\sum_{(m,p)\in\mathcal{S}}\frac{1}{|\underbrace{\xi(m,p)}_{\hat{d}}|}\sum_{\hat{e}_k\in\hat{d}}\gamma(\hat{e}_k \mid m, p, \hat{d}) \tag{7}$$

We evaluate Llama-3.2-Vision (Meta, 2024b) with 11B and 90B versions, GPT-4o, 4o-min and GPT-4 turbo vision (OpenAI, 2024), MiniCPM (Yao et al., 2024), PaliGemma (Beyer et al., 2024), Qwen2-VL (Wang et al., 2024b) with 2B and 7B versions, and Phi 3.5 Vision (Abdin et al., 2024) with our TLDR Model.

As shown in Table 2, GPT-4o is overall the best model with the least amount of hallucinations among all granularity. We also observe a strong correlation between model's hallucinate rates and its visual understanding and reasoning performance evaluated by MMMU. We conjecture for any VLM ξ,

$$\text{Performance}(\xi) \propto -\log \mathcal{H}_T(\xi) \tag{8}$$

5.3 Self-Correction with TLDR Models

Hallucination detection and evaluation (Zhou et al., 2024; Jing & Du, 2024; Yin et al., 2023) is a small leap forward. The most exciting usage of TLDR model is to guide models with **self-correction** and to be more grounded to the images by taking another guided look. Token-level annotations from TLDR model can enhance a vision-language model's ability to self-correct by providing detailed, granular feedback on specific parts of its output. These annotations allow the model to break down its response into smaller, interpretable units, aligning each token with visual and textual cues. By analyzing where errors occur at the token level—whether in object recognition, attribute descriptions, or language syntax—the model can more precisely identify the source of the mistake. Additionally, token-level feedback can guide the model to better align its language generation with the visual context, improving coherence and factual accuracy in its self-correction process. For self-correction experiments, we chose `GPT-4V` because it provides a strong baseline for assessing reward-driven refinements, while also including `Llama-3.2-90B-Vision` and `Qwen2-VL-7B` to showcase TLDR's effectiveness across diverse architectures.

In this section, we evaluate on **WinoGround** (Thrush et al., 2022) dataset to show whether given extra token-level annotation cues, the vision language model is able to self-correct its own halluci-

Model	Guidance Given By	# Samples	# Samples Flagged by RM	# Self-Corrected	Win	Tie	Loss
GPT-4V	TLDR (ours)	800	25	21	12	7	2
	Naive			15	2	11	2
Llama-3.2-90B	TLDR (ours)	800	10	8	6	1	1
	Naive			6	3	1	2
Qwen2-VL-7B	TLDR (ours)	800	25	16	9	5	2
	Naive			9	3	5	1

Table 3: **Self-Correction with the Guidance of TLDR Model.** A target model (GPT-4V, Llama-3.2-90B or Qwen2-VL-7B) is used to generate captions for 800 images from WinoGround, some of the captions are flagged by the reward model as containing bad tokens. When prompted to self-correct, extra guidance from TLDR helps the target model correct more of its own hallucinations with larger win rates.

Models	Tasks				
	BLINK ↑			IsoBench ↑	
	Count	Spatial Relation	Object Localize	Function Parity	Chess Winner
PaliGemma-3B	69.2	78.3	45.9	41.4	45.1
+ TLDR $(\tau = 0.25)$	**71.7**	80.4	**47.5**	**45.1**	45.1
+ TLDR $(\tau = 0.5)$	**71.7**	**81.1**	42.6	44.3	**47.5**
+ TLDR $(\tau = 1)$	12.5	2.1	42.6	34.4	44.8
Llama-3.2-11B-Vision	55.0	61.5	60.7	34.9	45.5
+ TLDR $(\tau = 0.25)$	**67.5**	65.0	**67.2**	**35.4**	43.6
+ TLDR $(\tau = 0.5)$	65.8	**65.7**	59.0	33.3	**47.9**
+ TLDR $(\tau = 1)$	61.7	**65.7**	56.6	35.1	39.4

Table 4: **Training TLDR model automatically gives a better vision language model for VQA.** We evaluate 3 versions of TLDR backbone model with different scales of LoRA α. They are distinguished by $\tau = \alpha_{\text{infer}}/\alpha_{\text{train}}$, the proportion of α at inference and training time. We find that when $\tau = 0.25$, it could improve the PaliGemma model's performance by at most 3.7 points and could improve Llama 3.2 model's performace by at most 12.5 points.

Models	Hallucination Rate (%) ↓		
	Token-Level	Sentence-Level	Response-Level
PaliGemma-3B	4.444	5.96	17.50
+ TLDR $(\tau = 0.10)$	0.991	3.80	10.53
+ TLDR $(\tau = 0.25)$	**0.172**	**1.13**	**3.96**
Llama-3.2-11B-Vision	0.073	0.85	1.88
+ TLDR $(\tau = 0.10)$	0.078	**0.69**	2.71
+ TLDR $(\tau = 0.25)$	**0.066**	0.74	**1.72**

Table 5: **Training TLDR model automatically gives a better vision language model with *less hallucination rate*.** Similar to Table 4 and the setup in Table 2, we evaluate TLDR backbone models' hallucination rate on the dense image captioning task. Models are able to reduce the hallucination rates, especially for less-performing models such as PaliGemma.

nations. Out of 800 captions generated by GPT-4V for images in WinGround, TLDR model flags 25 of them as including hallucinated tokens. As shown in Table 3, when prompted with TLDR guidance, GPT-4V attempts to self-correct 21 out of 25, and when evaluated by human annotators, 12 of them are improved, 7 of them are tied, and 2 of them are worsened. On the contrary, when prompted to self-correct *without* TLDR's guidance, GPT-4V attempts to self-correct 15 out of 25 with only 2 wins, 11 ties, and 2 losses. Examples of GPT-4V's self-correction results are shown in Figure 4. We include both prompt templates for self-correction, with and without TLDR's guidance in Appendix C.

5.4 TLDR Automatically Trains Token-Level Likelihood Optimization

The purpose of building reward models is to improve the backbone large vision language model. We find that, a free by-product of training the TLDR model is that the backbone model's weights are simultaneously updated together with the reward model head. As discussed in Section 5, the linear projection f_{proj} and the transformer decoder Gemma-2B f_{dec} are both updated with LoRA weights during training TLDR. Now we attach back the original pretrained language model head ℓ to the backbone of the updated PaliGemma model with TLDR (by discarding the reward model head h), we obtain an updated vision language model. Now we evaluate whether this new model has improved from the orginal model, by evaluating on both in-distribution tasks from BLINK (Fu et al., 2024b) and out-of-distribution tasks from IsoBench (Fu et al., 2024a). At inference time, we adopt a different LoRA alpha to merge the weights, for weight Θ for any updated module, $\Theta' = \Theta + \alpha_{\text{infer}} AB$, where A, B are trained LoRA weights. We find the proportion between α_{infer} and α_{train} could affect model performance significantly. We denote this proportion $\tau := \alpha_{\text{infer}}/\alpha_{\text{train}} \in [0, 1]$. With $\tau = 0$, we are evaluating the original model before training TLDR, and with $\tau = 1$, we are evaluating the model trained to provide token-level rewards. As shown in Table 4, a $\tau = 0.25$ gives the best performance.

Such automatic improvement is in fact that TLDR simultaneously trains the backbone VLM with likelihood optimization. The binary cross entropy objective on $\mathbb{P}(e \mid m, p, d)$ for any token e simply promotes the model to generate e if $\gamma^\star(e) = 1$, i.e., e is a good token; and suppresses the model to generate it if $\gamma^\star(e) = 0$, i.e., e is a bad token. As the linear projection layer f_{proj} is also finetuned, the model is promoted to be more visually grounded, with an improvement for spatial relationship and chess winner identification, both of which requires complex spatial reasoning on images.

5.5 Speeding Up Human Annotation with TLDR Models

Human annotations, especially on dense captions, are costly and model generated captions are not trustworthy. Recent work such as PixelProse (Singla et al., 2024) has started releasing model generated dense captions with an ambition to use these captions to train better models. However, on a random sampling of 3,000 images from PixelProse, TLDR model detects 22.39% of the captions have hallucinated tokens, with a token-level hallucination rate of 0.83% and a sentence-level hallucination rate of 5.23%.

Nonetheless, it's always easier and cheaper for human annotators to **correct** an existing caption than writing long captions **from scratch**. Instead of using model's self-correction as designed in Section 5.3, human correction could be more rigorous to provide better captions. Human annotators are given the similar instruction as model self-correction prompts. Instead of comparing their caption correction quality – corrected captions are later cross checked by annotators to ensure quality – annotators are asked to time their annotation speed. Each annotator is assigned two set of distinct samples, one with TLDR guidance and the other without, and is asked to fix the caption and time themselves. As shown in Table 6, all three annotators share a similar annotation speed when if given TLDR guidance or no extra guidance. Most importantly, all three annotators have a 3 times speed up, which could in the future help with creating large bulk of vision language data with lower annotation costs.

Annotator ID	Average Annotation Speed (seconds) with Guidance of	
	Binary Reward Model	TLDR Model
Annotator A	101.7	31.2
Annotator B	109.1	32.9
Annotator C	121.3	34.4
Average	110.7	32.8

Table 6: **TLDR Speeds up Human Annotation by *3 Times* to Fix Synthetic Image Caption in PixelProse (Singla et al., 2024).**

6 Discussion and Conclusion

In this paper, we introduced a **T**oken-**L**evel **D**etective **R**eward Model (**TLDR**) to provide fine-grained annotations for large vision language models. It is more interpretable than traditional naive binary reward models as TLDR can inform the user not only the response could be wrong but also *where* the response is wrong. Such feature enables many meaning usages, such as model's self-correction with TLDR guidance, and hallucination evaluation with TLDR annotations. We also presented a naive baseline where TLDR model is automatically a likelihood optimization method for its backbone vision language model, and the TLDR-tuned VLM is able to improve in several benchmarks. We believe a strong RM is a crucial basis for token-level DPO and PPO post-training. Finally, we show that, beyond guiding model's to self-correct, TLDR can also assist human annotators to fix synthetically generated image captions by improving the annotation speed by 3 times. Although our evaluation primarily focuses on hallucination mitigation, TLDR's token-level reward framework could be extended to broader safety and alignment tasks in vision-language generation.

Avenues ahead, we aim to design better human annotation interfaces under the human-computer interaction realm, to further reduce annotation overhead, to acquire large amount of high quality image captioning data with both positive human-corrected caption and negative model-generated caption. It will further facilitate us in designing token-level policy optimization methods. We believe our approach has the potential to advance the field of reward modeling and automatic evaluation. We hope that TLDR will make data annotation easier, and guide multimodal LLMs to hallucinate less. We hope our work can inspire further research to rethink the roles of reward models, and how their transparency, interpretability, and granularity can help advance the field of building better multimodal foundation models.

Acknowledgement

DF would like to thank Richard Yuanzhe Pang for the initial discussion on token-level predictions and automated model critiques for large language models. The detective and robot figures used in Figures 1 and 4 are from flaticon.com.

Getty Acknowledgement. Images in the paper that originated from the WinoGround dataset (Thrush et al., 2022) are a compilation of assets, including ©Getty Images/Natasha Breen, Maki Nakamura, Jessica Peterson, Kundanlall Sharma, lacaosa, Alberto Bogo, Vu Le, Toson Rueangsuksut, Nisian Hughes, Tanja Walter, Douglas Sacha, PBNJ Productions, Glow Images, 10'000 Hours, zoranm, Marlene Ford, Westend61.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL https://arxiv.org/abs/2404.14219.

Rami Aly, Christos Christodoulopoulos, Oana Cocarascu, Zhijiang Guo, Arpit Mittal, Michael Schlichtkrull, James Thorne, and Andreas Vlachos (eds.). *Proceedings of the Fourth Workshop on Fact Extraction and VERification (FEVER)*, Dominican Republic, November 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.fever-1.0.

Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. Introducing our multimodal models, 2023. URL https://www.adept.ai/blog/fuyu-8b.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers, Skanda Koppula, Fangyu Liu, Adam Grycner, Alexey Gritsenko, Neil Houlsby, Manoj Kumar, Keran Rong, Julian Eisenschlos, Rishabh Kabra, Matthias Bauer, Matko Bošnjak, Xi Chen, Matthias Minderer, Paul Voigtlaender, Ioana Bica, Ivana Balazevic, Joan Puigcerver, Pinelopi Papalampidi, Olivier Henaff, Xi Xiong, Radu Soricut, Jeremiah Harmsen, and Xiaohua Zhai. Paligemma: A versatile 3b vlm for transfer, 2024. URL https://arxiv.org/abs/2407.07726.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In Lluís Màrquez, Chris Callison-Burch, and Jian Su (eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://aclanthology.org/D15-1075.

Chameleon. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

Paul Francis Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *ArXiv*, abs/1706.03741, 2017. URL https://api.semanticscholar.org/CorpusID:4787508.

Deqing Fu, Ameya Godbole, and Robin Jia. SCENE: Self-labeled counterfactuals for extrapolating to negative examples. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7832–7848, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.485. URL https://aclanthology.org/2023.emnlp-main.485.

Deqing Fu, Ruohao Guo, Ghazal Khalighinejad, Ollie Liu, Bhuwan Dhingra, Dani Yogatama, Robin Jia, and Willie Neiswanger. IsoBench: Benchmarking multimodal foundation models on isomorphic representations. In *First Conference on Language Modeling (COLM)*, 2024a.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. *arXiv preprint arXiv:2404.12390*, 2024b.

Google. Gemini: A family of highly capable multimodal models, 2023.

Google. Gemma: Open models based on gemini research and technology, 2024. URL https://arxiv.org/abs/2403.08295.

Alex Havrilla, Sharath Raparthy, Christoforus Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, and Roberta Raileanu. Glore: When, where, and how to improve llm reasoning via global and local refinements, 2024. URL https://arxiv.org/abs/2402.10963.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Liqiang Jing and Xinya Du. Fgaif: Aligning large vision-language models with fine-grained ai feedback, 2024. URL https://arxiv.org/abs/2404.05046.

Liqiang Jing, Ruosen Li, Yunmo Chen, and Xinya Du. FaithScore: Fine-grained evaluations of hallucinations in large vision-language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 5042–5063, Miami, Florida, USA, November 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-emnlp.290.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32 – 73, 2016. URL https://homes.cs.washington.edu/~ranjay/visualgenome/index.html.

Jinhyuk Lee, Mujeen Sung, Jaewoo Kang, and Danqi Chen. Learning dense representations of phrases at scale. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6634–6647, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.518. URL https://aclanthology.org/2021.acl-long.518.

Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback, 2023. URL https://arxiv.org/abs/2302.12192.

Lei Li, Zhihui Xie, Mukai Li, Shunian Chen, Peiyi Wang, Liang Chen, Yazheng Yang, Benyou Wang, Lingpeng Kong, and Qi Liu. VLFeedback: A large-scale AI feedback dataset for large vision-language models alignment. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 6227–6246, Miami, Florida, USA, November 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.emnlp-main.358.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 292–305, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.20. URL https://aclanthology.org/2023.emnlp-main.20.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. *ArXiv*, abs/2404.01291, 2024. URL https://api.semanticscholar.org/CorpusID:268857167.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023. URL https://arxiv.org/abs/2304.08485.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, January 2024. URL https://llava-vl.github.io/blog/2024-01-30-llava-next/.

Holy Lovenia, Wenliang Dai, Samuel Cahyawijaya, Ziwei Ji, and Pascale Fung. Negative object presence evaluation (NOPE) to measure object hallucination in vision-language models. In Jing Gu, Tsu-Jui (Ray) Fu, Drew Hudson, Asli Celikyilmaz, and William Wang (eds.), *Proceedings of the 3rd Workshop on Advances in Language and Vision Research (ALVR)*, pp. 37–58, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.alvr-1.4. URL https://aclanthology.org/2024.alvr-1.4.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts, 2024. URL https://arxiv.org/abs/2310.02255.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct, 2023. URL https://arxiv.org/abs/2308.09583.

Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, et al. Mm1: Methods, analysis & insights from multimodal llm pre-training. *arXiv preprint arXiv:2403.09611*, 2024.

Meta. The llama 3 herd of models, 2024a. URL https://arxiv.org/abs/2407.21783.

Llama Team @ Meta. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models, 2024b. URL https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/.

Abhika Mishra, Akari Asai, Vidhisha Balachandran, Yizhong Wang, Graham Neubig, Yulia Tsvetkov, and Hannaneh Hajishirzi. Fine-grained hallucination detection and editing for language models, 2024. URL https://arxiv.org/abs/2401.06855.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Randy Zhong, Juntong Song, and Tong Zhang. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models, 2024. URL https://arxiv.org/abs/2401.00396.

Yasumasa Onoe, Sunayana Rane, Zachary Berger, Yonatan Bitton, Jaemin Cho, Roopal Garg, Alexander Ku, Zarana Parekh, Jordi Pont-Tuset, Garrett Tanzer, Su Wang, and Jason Baldridge. Docci: Descriptions of connected and contrasting images, 2024. URL https://arxiv.org/abs/2404.19753.

OpenAI. Gpt-4o, 2024. URL https://openai.com/index/hello-gpt-4o/.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. URL https://arxiv.org/abs/2203.02155.

Piotr Padlewski, Max Bain, Matthew Henderson, Zhongkai Zhu, Nishant Relan, Hai Pham, Donovan Ong, Kaloyan Aleksiev, Aitor Ormazabal, Samuel Phua, Ethan Yeo, Eugenie Lamprecht, Qi Liu, Yuqi Wang, Eric Chen, Deyu Fu, Lei Li, Che Zheng, Cyprien de Masson d'Autume, Dani Yogatama, Mikel Artetxe, and Yi Tay. Vibe-eval: A hard evaluation suite for measuring progress of multimodal language models, 2024. URL https://arxiv.org/abs/2405.02287.

Renjie Pi, Tianyang Han, Wei Xiong, Jipeng Zhang, Runtao Liu, Rui Pan, and Tong Zhang. Strengthening multimodal large language model with bootstrapped preference optimization. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part XXXIII*, pp. 382–398, Berlin, Heidelberg, 2024. Springer-Verlag. ISBN 978-3-031-73413-7. doi: 10.1007/978-3-031-73414-4_22. URL https://doi.org/10.1007/978-3-031-73414-4_22.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024. URL https://arxiv.org/abs/2305.18290.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://aclanthology.org/P18-2124.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Vasu Singla, Kaiyu Yue, Sukriti Paul, Reza Shirkavand, Mayuka Jayawardhana, Alireza Ganjdanesh, Heng Huang, Abhinav Bhatele, Gowthami Somepalli, and Tom Goldstein. From pixels to prose: A large dataset of dense image captions, 2024. URL https://arxiv.org/abs/2406.10328.

Jiao Sun, Deqing Fu, Yushi Hu, Su Wang, Royi Rassin, Da-Cheng Juan, Dana Alon, Charles Herrmann, Sjoerd van Steenkiste, Ranjay Krishna, and Cyrus Rashtchian. Dreamsync: Aligning text-to-image generation with image understanding feedback, 2023a. URL https://arxiv.org/abs/2311.17946.

Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. Aligning large multimodal models with factually augmented rlhf. 2023b.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5228–5238, 2022.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. Fact or fiction: Verifying scientific claims. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7534–7550, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.609. URL https://aclanthology.org/2020.emnlp-main.609.

David Wadden, Kyle Lo, Bailey Kuehl, Arman Cohan, Iz Beltagy, Lucy Lu Wang, and Hannaneh Hajishirzi. Scifact-open: Towards open-domain scientific claim verification, 2022. URL https://arxiv.org/abs/2210.13777.

Binghai Wang, Rui Zheng, Lu Chen, Yan Liu, Shihan Dou, Caishuang Huang, Wei Shen, Senjie Jin, Enyu Zhou, Chenyu Shi, Songyang Gao, Nuo Xu, Yuhao Zhou, Xiaoran Fan, Zhiheng Xi, Jun Zhao, Xiao Wang, Tao Ji, Hang Yan, Lixing Shen, Zhan Chen, Tao Gui, Qi Zhang, Xipeng Qiu, Xuanjing Huang, Zuxuan Wu, and Yu-Gang Jiang. Secrets of rlhf in large language models part ii: Reward modeling, 2024a. URL https://arxiv.org/abs/2401.06080.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution, 2024b. URL https://arxiv.org/abs/2409.12191.

Dustin Wright, David Wadden, Kyle Lo, Bailey Kuehl, Arman Cohan, Isabelle Augenstein, and Lucy Lu Wang. Generating scientific claims for zero-shot scientific fact checking. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2448–2460, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.175. URL https://aclanthology.org/2022.acl-long.175.

Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A. Smith, Mari Ostendorf, and Hannaneh Hajishirzi. Fine-grained human feedback gives better rewards for language model training, 2023. URL https://arxiv.org/abs/2306.01693.

Kailai Yang, Zhiwei Liu, Qianqian Xie, Jimin Huang, Erxue Min, and Sophia Ananiadou. Selective preference optimization via token-level reward function estimation, 2024.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, Qianyu Chen, Huarong Zhou, Zhensheng Zou, Haoye Zhang, Shengding Hu, Zhi Zheng, Jie Zhou, Jie Cai, Xu Han, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint 2408.01800*, 2024.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Tong Xu, Hao Wang, Dianbo Sui, Yunhang Shen, Ke Li, Xing Sun, and Enhong Chen. Woodpecker: Hallucination correction for multimodal large language models. *arXiv preprint arXiv:2310.16045*, 2023.

Eunseop Yoon, Hee Suk Yoon, SooHwan Eom, Gunsoo Han, Daniel Nam, Daejin Jo, Kyoung-Woon On, Mark Hasegawa-Johnson, Sungwoong Kim, and Chang Yoo. TLCR: Token-level continuous reward for fine-grained reinforcement learning from human feedback. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics ACL 2024*, pp. 14969–14981, Bangkok, Thailand and virtual meeting, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.889. URL https://aclanthology.org/2024.findings-acl.889.

Lili Yu, Bowen Shi, Ramakanth Pasunuru, Benjamin Muller, Olga Golovneva, Tianlu Wang, Arun Babu, Binh Tang, Brian Karrer, Shelly Sheynin, et al. Scaling autoregressive multi-modal models: Pretraining and instruction tuning. *arXiv preprint arXiv:2309.02591*, 2023a.

Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, et al. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. *arXiv preprint arXiv:2312.00849*, 2023b.

Tianyu Yu, Haoye Zhang, Yuan Yao, Yunkai Dang, Da Chen, Xiaoman Lu, Ganqu Cui, Taiwen He, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Rlaif-v: Aligning mllms through open-source ai feedback for super gpt-4v trustworthiness. *arXiv preprint arXiv:2405.17220*, 2024.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi, 2024. URL https://arxiv.org/abs/2311.16502.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training, 2023. URL https://arxiv.org/abs/2303.15343.

Yiyang Zhou, Chenhang Cui, Jaehong Yoon, Linjun Zhang, Zhun Deng, Chelsea Finn, Mohit Bansal, and Huaxiu Yao. Analyzing and mitigating object hallucination in large vision-language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=oZDJKTlOUe.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences, 2020. URL https://arxiv.org/abs/1909.08593.

Appendix

A Synthetic Data Generation

A.1 Prompt for Synthesizing Image Caption from VQA

Synthesize Caption from Visual Question Answering

```
Your task is to convert a list of question-answer pairs to a
descriptive paragraph.
Keep in mind these rules:
Do not start with greetings or salutations.  Simply return
the new caption.
Do not write anything else at the end of your response.
Your crafted descriptive caption should be very faithful to
the given question-answer pairs.
Do not add any additional information that is not in the
question-answer pairs.
The parapgraph should not start with the photo, the image,
the picture etc.
Instead of saying, for example, the photo shows a cute koala
bear sleeping on the tree, you should just say a cute koala
bear is sleeping on the tree.
For example, you are given the following question-answer
pairs:
Question:  What is the color of the mug?  Answer:  red.
Question:  Where is the mug at?  Answer:  on the table.
Question:  Is there anything written on the mug?  Answer:
Yes.
Question:  What is written on the mug?  Answer:  Hello World.
Question:  What is the color of the texts?  Answer:  yellow.
Your response should be a descriptive paragraph that
aggregate the information from the question-answer pairs:
Paragraph:
A red mug sitting on the table has yellow texts written on
it:  Hello World.
```

```
Now you are given the following question-answer pairs and are
asked to generate a paragraph:
Question: ⟨Question⟩ Answer: ⟨Answer⟩
Question: ⟨Question⟩ Answer: ⟨Answer⟩
⋮
Paragraph:
```

A.2 Prompts for Synthesizing VQA Negatives

Prompt Template for Generating VQA Negatives

```
You are provided with a visual question-answer pair. Your
task is to generate a wrong answer that is subtly different
from the original answer.
Keep in mind the following rules:
Keep formatting the same, such as sentence structure,
paragraph blocks, or newlines.
The changes should be subtle but concrete, replacing words
and phrases with opposite meanings or alternative options.
Do not start with greetings or salutations. Simply return
the new answer.
Do not write anything else at the end of your response.
Do not fix any typos or grammar errors. If there are any,
please ignore them.
The new answer should be nearly identical, other than 1 or 2
very small changes.
The changes should be very visually different. The new
answer should be realistic.
More importantly, keep the wrong answer within the same
taxonomy as the original answer.
Here are some examples.
Question: What color is the apple? Answer: red. Wrong
answer: green.
Question: What is the spatial relationship between the man
and the chair? Answer: The man is sitting on a brown chair.
Wrong answer: The man is sitting next to a brown chair.
Question: How many apples are there? Answer: There are 5
apples. Wrong answer: There are 6 apples.
Now write your new answer:
Question: ⟨Question⟩ Answer: ⟨Answer⟩ Wrong answer:
```

Check if VQA Negative is Valid

```
You are provided with a paragraph, and a question-answer
pair. Your task is to determine if the answer is a valid
answer to the question given the paragraph.
The answer should be a simple yes or no. Do not write
anything else at the end of your response. Here are some
examples.
Paragraph: The man is sitting on a brown chair. Question:
What is the spatial relationship between the man and the
chair? Answer: The man is sitting on a brown chair. Valid
answer: yes.
```

```
Paragraph: There are six apples on the table and they are
all red. One of the apples is rotten and is at the left side
of the table. Question: How many apples are there? Answer:
There are 5 apples. Valid answer: no.
Paragraph: On a sunny day, a man sailing a boat on the ocean
sees a fish jumping out of the water. Question: What is the
man doing? Answer: The man is fishing. Valid answer: no.
Now check this paragraph and the question answer pair:
Paragraph: ⟨Paragraph⟩ Question: ⟨Question⟩ Answer: ⟨Answer⟩
Valid answer:
```

Note: we discard the synthetic negatives that are marked as *valid* because the perturbation is unsuccessful.

A.3 Prompts for Synthesizing Image Caption Negatives

We first present the general prompt for perturbation and then break these down into taxonomy-specific rules and in-context examples on all 8 taxonomies described in Section 4.

General Prompt

```
You are provided with a caption to an image. Your task is
to generate a new caption that is subtly different from the
original caption.
Keep in mind the following rules:
Keep formatting the same, such as sentence structure,
paragraph blocks, or newlines.
The changes should be subtle but concrete, replacing words
and phrases with opposite meanings or alternative options.
Do not start with greetings or salutations. Simply return
the new caption.
Do not write anything else at the end of your response.
Do not fix any typos or grammar errors. If there are any,
please ignore them.
The new caption should be nearly identical, other than 1 or 2
very small changes.
The changes should be very visually different.
The new caption should be realistic.
Most importantly, make the change with the following
taxonomy: ⟨Taxonomy⟩
Here are the additional rules for the taxonomy
⟨Taxonomy-Specific Rules and In-Context Examples⟩
Here is the original caption: ⟨Caption⟩
Now write your new caption:
```

Spatial Relationship Rules and In-Context Examples

```
Change the spatial relationship of two objects in the given
caption.
The new spatial relationship should be different from the
original spatial relationship.
The new spatial relationship should be realistic, and is
visually different from the original spatial relationship.
You should fix the consistency of the caption. If you change
the spatial relationship, you should also change the noun,
verb, etc.  so that there is no grammar error.
Do not change anything not related to spatial relationship,
even there are spatial relationship present.
Do not change other attributes of objects, such as color,
texture, material, etc.
If there are no spatial relationship present in the caption,
you should simply copy the original caption.

For example,
Original caption:  A man is sitting on the left to a coffee
table.
New caption:  A man is sitting on the right to a coffee
table.

Original caption:  A duck is swimming in a pool and a fish is
swimming underneath.
New caption:  A duck is swimming in a pool and a fish is
swimming on top of it.

Original caption:  A man is sitting in front of a car.
New caption:  A man is sitting in a car.

Original caption:  An apple is placed on an open book.
New caption:  An apple is placed under an open book.

Original caption:  There is a black cat.
New caption:  There is a black cat.
```

Visual Attribute Rules and In-Context Examples

```
Change the visual attributes of objects in the given caption.
The new visual attributes should be different from the
original visual attributes.
The new visual attributes should be realistic, and is
visually different from the original visual attributes.
You should fix the consistency of the caption.  If you change
the visual attributes, you should also change the noun, verb,
etc.  so that there is no grammar error.
Do not change anything not related to visual attributes, even
there are visual attributes present.

For example,
Original caption:  A man is sitting on a marble bench.
New caption:  A man is sitting on a wooden bench.

Original caption:  A red apple is placed on a table.
New caption:  A green apple is placed on a table.

Original caption:  A corgi dog is sitting on a chair.
New caption:  A corgi dog is sitting on a couch.
```

Attribute Binding Rules and In-Context Examples

```
Change the attribute bindings of many objects in the given
caption.
```

```
Definition of attribute binding is that the attribute of
an object is bound to the object.  Changing the attribute
binding means swap the attributes of many different objects.
The new attribute bindings should be different from the
original attribute bindings.
The new attribute bindings should be realistic, and is
visually different from the original attribute bindings.
You should fix the consistency of the caption.  If you change
the attribute bindings, you should also change the noun,
verb, etc.  so that there is no grammar error.
Do not change anything not related to attribute bindings,
even there are attribute bindings present.
Do not add or deletes any objects and attributes other than
changing the attribute bindings.

For example,
Original caption:  A man is sitting on a bench and a woman is
sitting on a chair.
New caption:  A man is sitting on a chair and a woman is
sitting on a bench.

Original caption:  A red apple and a stack of blue books are
on a table.
New caption:  A blue apple and a stack of red books are on a
table.

Original caption:  An apple made of aluminum and a chair made
of wood are on display at the art museum.
New caption:  An apple made of wood and a chair made of
aluminum are on display at the art museum.

Original caption:  Two yellow cats are chasing one flurry
blue ball of yarn.
New caption:  Two blue cats are chasing one flurry red ball
of yarn.
```

Object Identification Rules and In-Context Examples

```
Change the object identifications in the given caption.
Definition for object identification is that the entity of an
object, e.g., a book, a man, a dog, a table, etc.
The new object should be different from the original object.
The new object should be realistic, and is visually different
from the original object.
If possible, you can make the new object subtly different
from the original object.  For example, change a corgi dog to
a dachshund dog.
You should fix the consistency of the caption.  If you change
the object, you should also change the noun, verb, etc.  so
that there is no grammar error.
Do not change anything not related to object identification,
even there are object identifications present.
Do not add or deletes any objects and attributes other than
changing the object identification.
```

```
Here are some valid examples
Original caption:  A man is sitting on a bench.
New caption:  A man is sitting on a chair.

Original caption:  A red apple is placed on a table.
New caption:  A red apple is placed on a bench.

Original caption:  A corgi dog is sitting on a chair.
New caption:  A dachshund dog is sitting on a couch.
```

COUNTING RULES AND IN-CONTEXT EXAMPLES

```
Change the counting of one object in the given caption to a
different number.
The new number should be different from the original number.
The new number should be realistic, and is not so different
from the original number.
You should fix the consistency of the caption.  If you change
the number, you should also change the noun, verb, etc.  so
that there is no grammar error.
Only change the counting of one object in the caption.  Do
not change the number of other objects.
Do not change anything not related to counting, even there
are numbers present.
Do not change things related to written texts in quotation
marks.  For example, the original caption has A man wears a
shirt with text ’cute cat’ written on it.  DO NOT change the
caption to A man wears a shirt with text ’cute cats’ written
on it.
Do not change things ralated to time in the caption.  For
example, the original caption has The clock reads 13:00.  DO
NOT change the caption to The clock reads 14:00.
Do not change things related to proportions.  For example,
the orginal caption has The book covers 2/3 of the table.  Do
NOT change the caption to The book covers 3/4 of the table.
You can change the caption to Two books cover 2/3 of the
table instead.
If there are no counting in the caption, you should simply
copy the original caption.

Here are some valid examples
Original caption:  There are five cats on the table and they
are black.
New caption:  There are seven cats on the table and they are
black.

Original caption:  There are two dogs standing on the chairs,
one is white and one is black.
New caption:  There are three dogs standing on the chairs,
one is white and the other two are brown.

Original caption:  A side view of a Rouen duck that is brown
and tan and in some water.  It is facing to the right.
New caption:  A side view of two Rouen ducks that are brown
and tan and in some water.  They are facing to the right.
```

```
Original caption:  A pair of stop sign poles on a street.
New caption:  Three stop sign poles on a street.

Original caption:  The sky is blue.
New caption:  The sky is blue.
```

Small Objects Rules and In-Context Examples

```
Change the small and background objects in the given caption.
The new small and background objects should be subtly
different.  You can change their counts, size, shape, color,
etc.
You should ONLY change very small, neglible, background
objects, that are explictly described as so in the caption.
For example, you can pay attention to words like tiny, small,
mini, micro, nano, pico, femto, nano, micro, milli, etc.
If there are no small objects in the caption, you should
simply copy the original caption.

For example,
Original caption:  A man is sitting on a bench, in a library
with a white background board.  On the bookshelf, there is a
tiny crystal superman figure standing on a stack of books.
New caption:  A man is sitting on a bench, in a library with
a black background board.  On the bookshelf, there is a tiny
plastic batman figure standing in front of a stack of books.

Original caption:  A man is sitting on a bench, in a library
with a white background board.
New caption:  A man is sitting on a bench, in a library with
a black background board.
```

Text OCR Rules and In-Context Examples

```
Change the text OCR in the image caption to a different text.
The new text should be different from the original text.
The new text should be realistic to the context, and is
visually different from the original text.
You should fix the consistency of the caption.  If you change
the text, you should also change the noun, verb, etc.  so
that there is no grammar error.
Do not change anything not related to text, even there are
texts present.
If there are no texts OCR present in the caption, you should
find a place to put some reasonable text OCR. If you can’t
find a place to put some reasonable text OCR, you should
simply copy the original caption.

For example,
Original caption:  A man is wearing a shirt with texts
’SUPERMAN’.
New caption:  A man is wearing a shirt with texts ’BATMAN’.
```

```
Original caption:  The digital clock reads 12:00 AM.
New caption:  The digital clock reads 12:08 PM.

Original caption:  The road sign says ’STOP’.
New caption:  The road sign says ’YIELD’.

Original caption:  The board says ’Best College in the US’.
New caption:  The board says ’Best College in the World’.

Original caption:  A man wearing yellow shirt is sitting on
the bench.
New caption:  A man wearing yellow shirt with words ’Hello
World’ is sitting on the bench.
```

COUNTERFACTUAL RULES AND IN-CONTEXT EXAMPLES

```
Change the caption with counterfactuals.
The new caption should be different from the original
caption.
The new caption should be realistic, and is visually
different from the original caption.  You should fix the
consistency of the caption.  If you change the caption, you
should also change the noun, verb, etc.  so that there is no
grammar error.
Do not change anything not related to counterfactuals, even
there are counterfactuals present.
Do not add or deletes any objects and attributes other than
changing the counterfactuals.
If it’s hard to put in counterfactual, you should simply copy
the original caption.

For example,
Original caption:  A man is sitting on a bench.
New caption:  A man is not sitting on a bench.

Original caption:  A red apple is placed on a table.
New caption:  A red apple is not placed on a table.

Original caption:  A soldier.
New caption:  A soldier has no sword in hand.
```

A.4 STATISTICS OF DATA

Task	Data Source	Taxonomy	# Positive	# Negative	Train Set Proportion (%)
VQA	VG100K	—	1,179,007	1,179,007	80%
Image Caption	Synthetic Caption from VG100K	Spatial Relation	94,684	45,225	
		Visual Attribute		86,366	
		Attribute Binding		59,219	
		Object Identification		75,328	
		Counting		75,156	
		Small Object		80,455	
		Text OCR		84,164	
		Counterfactual		57,153	
Image Caption	DOCCI	Spatial Relation	14,639	8,867	65%
		Visual Attribute		13,811	
		Attribute Binding		13,561	
		Object Identification		10,618	
		Counting		10,491	
		Small Object		11,680	
		Text OCR		13,366	
		Counterfactual		12,844	

Table 7: **Statistics of Data.** Overall, we have over 1M VQA data with both positive and negative answers, and over 100K caption datapoints with 650K negative captions. We oberserve that we have the least amount of spatial relationship data, because spatial relationship negatives are the hardest to synthesize and not every caption has spatial relationship descriptions.

B Training and Model Performance

B.1 Model Training Setup and Hyperparameters

In the section, we present all the (hyper-)paramters we used to training TLDR model.

Hyperparameters for training TLDR Model	
Base Model	`PaliGemma-3B-Mix-448`
Image Resolution	448 × 448
Number of Image Tokens	1024
Hidden Dimension Size	2048
LoRA Rank	512
LoRA α	128
LoRA dropout	0.1
GPU	8 × NVIDIA H100
Batch Size	8
Gradient Accumulation Steps	8
Warmup Steps	200
Learning Rate	0.001
Learning Rate Scheduler	Cosine

Table 8: **Hyperparameters for training TLDR Model with PaliGemma Backbone.**

Hyperparameters for training TLDR Model	
Base Model	`Llama-3.2-11B-Vision`
Image Resolution	1120 × 1120
Number of Image Tokens	1024
Hidden Dimension Size	4096
LoRA Rank	512
LoRA α	128
LoRA dropout	0.1
GPU	8 × NVIDIA H100
Batch Size	8
Gradient Accumulation Steps	8
Warmup Steps	200
Learning Rate	0.001
Learning Rate Scheduler	Cosine

Table 9: **Hyperparameters for training TLDR Model with Llama Vision Backbone.**

B.2 Performance Evaluation

In this section, we present TLDR model's performance by taxonomy, and its comparison to the naive binary RM and TLDR's ablations discussed in Section 5. We also provide the ablation study on TLDR's design choices in Table 11.

Taxonomy	Naive RM	TLDR (ours)	ABLATION	
			Only f_{proj}	Only f_{dec}
Spatial Relationship	**74.1**	60.2	50.0 (-10.2)	51.8 (-8.4)
Visual Attribute	**89.8**	**89.8**	54.7 (-35.1)	**89.8** (±0.0)
Attribute Binding	88.1	**90.6**	55.0 (-35.6)	85.0 (-5.6)
Object Identification	73.2	**90.6**	51.4 (-39.2)	86.9 (-3.7)
Counting	71.0	**73.9**	50.7 (-23.2)	68.1 (-5.8)
Small Object	**79.0**	75.0	51.6 (-23.4)	71.8 (-3.2)
Text OCR	82.6	**86.5**	52.2 (-34.3)	83.7 (-2.8)
Counterfactual	87.3	**90.0**	53.3 (-36.7)	89.3 (-0.7)
OVERALL	81.1	**83.1**	52.5 (-30.6)	79.4 (-3.7)

Table 10: **Response-Level Accuracy by Taxonomy.** We find that overall TLDR models have higher response-level accuracy except for spatial relationship and small objects. We suspect the cause is from a brutal conversion of prediction from token-level to response level: by taking the product $\rho_\gamma(m, p, d) = \prod_{e \in d} \gamma(e \mid m, p, d)$, any single token e has the power to veto the entire response d. In delicate and subtle instances such as spatial relationship and small objects, such veto power by any single token is too brutal, and a more elegant conversion from token-level probabilities $\mathbb{P}_\gamma(e \mid m, p, d)$ could be interesting for future work.

Multimodal Projection f_{proj}	Gemma Decoder f_{dec}	Token-Level Accuracy $\mathcal{A}_T$	Sentence-Level Accuracy $\mathcal{A}_S$	Response-Level Accuracy $\mathcal{A}_R$	mAP (neg|pos)
✓	✓	**98.6**	**86.5**	**83.1**	(**41.3**|**99.8**)
✓		97.4	80.0	52.5	(18.2|99.2)
	✓	98.3	84.8	79.4	(38.2|99.7)

Table 11: **Ablation Study on `PaliGemma-3B` Base Model.** It shows that unfreezing gradients on both linear projection layer f_{proj} and the transformer decoder layers f_{dec} are meaningful. Without tuning f_{dec}, the model barely works, and without f_{proj}, as shown in Table 10, the model is less visually grounded, especially on spatial relationship and counting.

B.3 Correlation Between Model Performance and Token-Level Hallucination Rate

This section, we draw plots to back our conjecture in Eq. 7 that,for any VLM ξ,

$$\text{Performance}(\xi) \propto -\log \mathcal{H}_T(\xi)$$

As shown in Figure 6, the linear correlation is very significant both visually and backed by the small p-value.

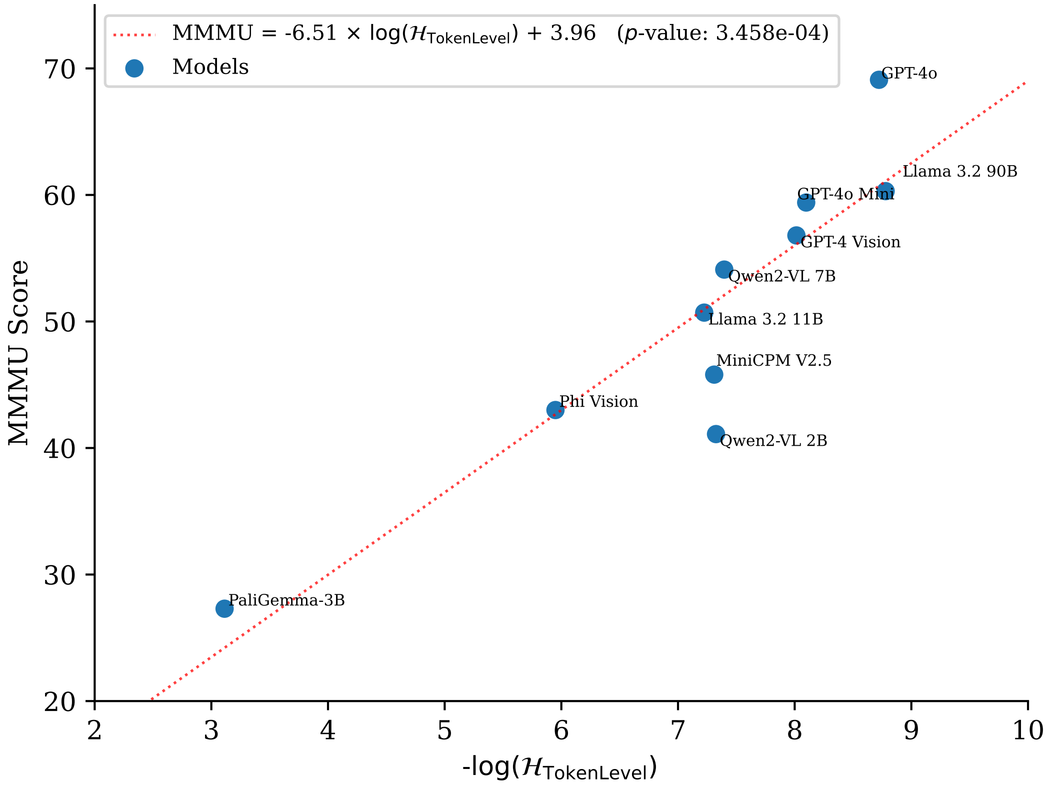


Figure 5: There is a strong linear correlation between model performance (evaluated by MMMU) and the negative log hallucination rate $-\log(\mathcal{H}_T)$, which is an approxy to model's negative log-likelihood of producing a correct token. The p-value of this linear correlation is 3.458e-4.

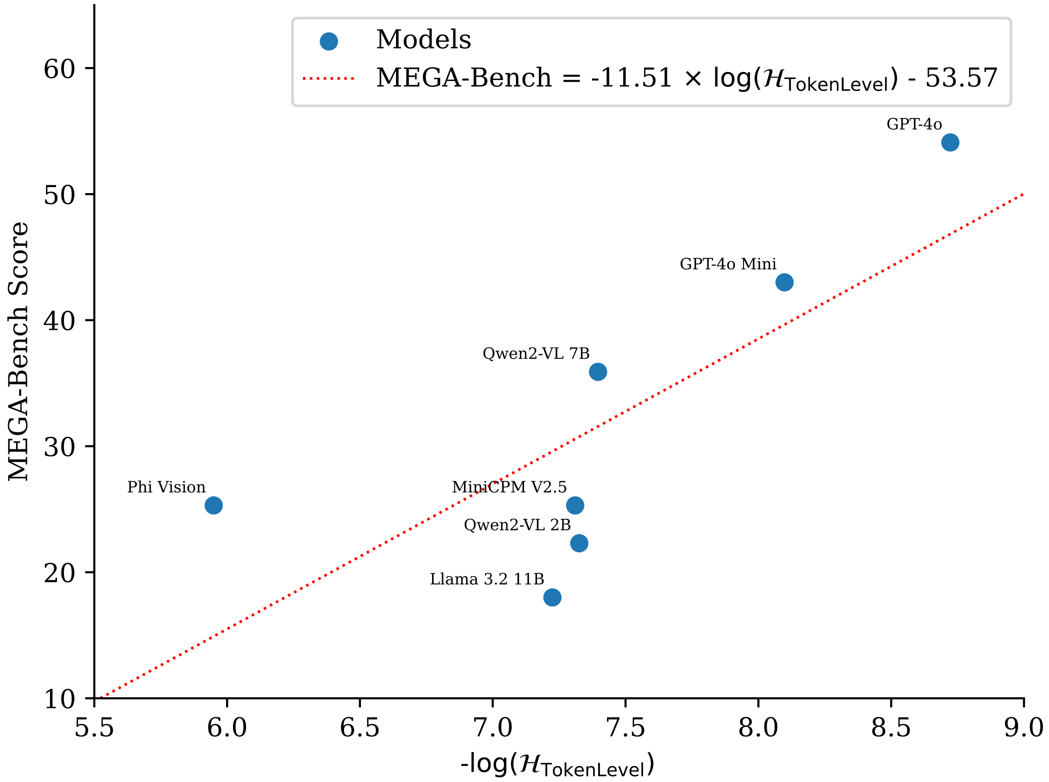


Figure 6: There is a strong linear correlation between model performance (evaluated by MEGA-Bench) and the negative log hallucination rate $-\log(\mathcal{H}_T)$, which is a proxy to model's negative log-likelihood of producing a correct token. The p-value of this linear correlation is 0.047.

C Self Correction

In the section, we present two exemplar prompts used for self-correction – one with TLDR's guidance and the other with only naive binary RM's guidance saying the original generation is wrong. The displayed image is from WinoGround.

Note: For better visualization, we highlight the TLDR Model's annotations in **red** when presenting the TLDR prompts.

Self Correction with TLDR's Guidance

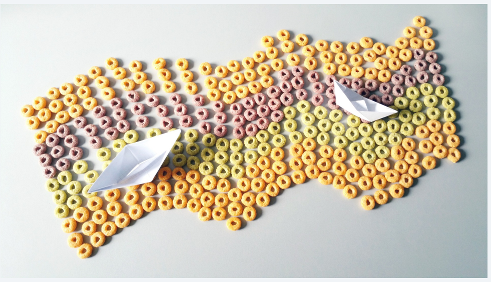


Image:

You task to make the provided 'Response' to the 'Text Query' better aligned with the given image.
The 'Response' has several sentences with issues that you need to take a closer look at. Focus on the image more when looking at the highlighted words or phrases in these sentences.
Here are some rules to keep in mind:

1. Your edited response should be as close as possible to the original response (minimal edits) but without the errors.
2. You should try to fix the errors by only changing the highlighted words. If you think these words are hallucinated, after looking closely at the image again, you can delete them.
3. If you think you can't do minimal edits, you can rewrite the whole sentences with errors. However, do not rewrite sentences without any annotated errors.
4. For sentences not enumerated, you can simply copy them if your changes won't affect them. If you changes to other marked sentences will also change the meaning of the unmarked sentences, you can change them as well. Please make the whole paragraph coherent.
5. The highlighted words that require extra attention are not necessarily always incorrect. If you think they align well with the image, you can keep them as is. After you look closer to the image, use your own judgement to decide if they need to be changed, deleted, or kept. You may change or delete the highlighted tokens if you think they are not aligned with the image. Otherwise, you can keep them as is.

Text Query: Describe the image in details.

Response: The image shows a creative arrangement of colorful **Lego bricks** forming a shape that resembles a **cat**. The **bricks** are in various colors such as yellow, pink, and green. There are two white paper boats placed on top of the **Lego cat**, one on the left and one on the right side. The background is a plain, light color, providing a neutral backdrop that highlights the colorful **Lego cat**.

Please take a closer look at these words or phrases: Lego, Lego bricks, Lego cat, bricks, cat.

Now we break them into their corresponding sentences to provide you with more context.

Please fix the sentence "The image shows a creative arrangement of colorful Lego bricks forming a shape that resembles a cat" with more attention to the following words: Lego bricks, cat.

Please fix the sentence "The bricks are in various colors such as yellow, pink, and green" with more attention to the following words: bricks.

Please fix the sentence "There are two white paper boats placed on top of the Lego cat, one on the left and one on the right side" with more attention to the following words: Lego.

Please fix the sentence "The background is a plain, light color, providing a neutral backdrop that highlights the colorful Lego cat" with more attention to the following words: Lego cat.

Please correct all the errors. Do not start or end the edited response with anything extra.

Corrected Response:

Self Correction with Naive RM's Guidance (no TLDR)

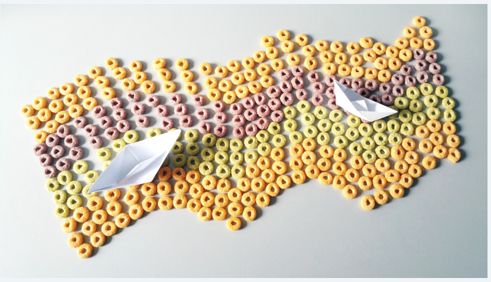


Image:

You task to make the provided 'Response' to the 'Text Query' better aligned with the given image.
Here are some rules to keep in mind:

1. Your edited response should be as close as possible to the original response (minimal edits) but without the errors.
2. If you think you can't do minimal edits, you can rewrite the whole sentences with errors. However, do not rewrite sentences without any annotated errors.

Text Query: Describe the image in details.

```
Response:  The image shows a creative arrangement of colorful
Lego bricks forming a shape that resembles a cat.  The bricks
are in various colors such as yellow, pink, and green.  There
are two white paper boats placed on top of the Lego cat,
one on the left and one on the right side.  The background
is a plain, light color, providing a neutral backdrop that
highlights the colorful Lego cat.

Please correct all the errors.  Do not start or end the
edited response with anything extra.

Corrected Response:
```

D LIMITATIONS

In this work, we mainly studied natural images, and whether training on natural images can extend robustly to diagrams, OCR, and more complex text-rich images is still unknown. However, we believe that as long as the training and testing distributions are similar, i.e., extending TLDR training set to text-rich images, it should also succeed. We will leave this thoughtful extension to future works. Although self-correcting smaller and less powerful models with TLDR guidance shows success, we also acknowledge the potential limitations of self-correcting more powerful models such as GPT-4o, which hallucinate less frequently, and running TLDR on each of its outputs could be costly and unrewarding. Nonetheless, it would also be nice and interesting to study if a much smaller model (e.g. Qwen2-VL-7B) with the TLDR guidance, can self-correct itself to reach GPT-4o level performance.