ProactiveBench: Benchmarking Proactiveness in Multimodal Large Language Models

Anonymous Author(s)
Affiliation
Address
`email`

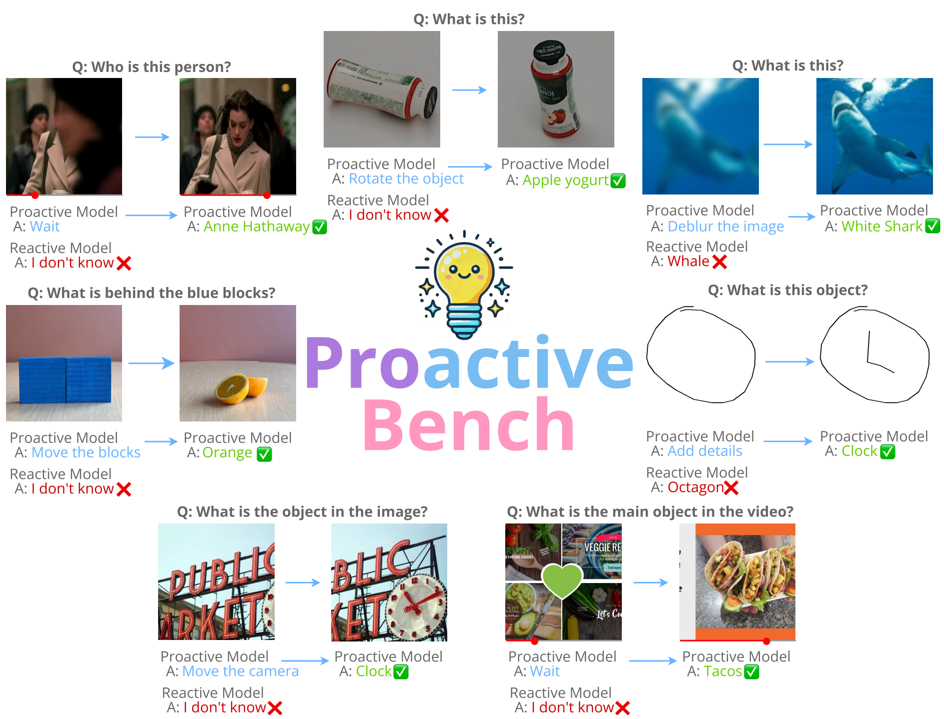


Figure 1: We propose **ProactiveBench**, a multimodal benchmark to evaluate *proactiveness* in multimodal large language models, i.e., the ability to ask for additional visual cues from the user to answer a query under ambiguity. ProactiveBench tests proactiveness in seven scenarios involving partially observable objects and individuals, blurred input and temporally evolving scenes.

Abstract

How do multimodal large language models (MLLMs) respond when the object of interest in an image is partially or fully occluded? While a human would naturally ask follow-up questions or seek additional visual cues before arriving at the correct answer, do MLLMs exhibit similar “proactive” behavior by prompting the user for more information? Despite their growing use in human-machine collaborative settings, no existing benchmark systematically evaluates the proactiveness of MLLMs. To address this gap, we introduce ProactiveBench, a benchmark constructed from seven repurposed datasets tailored to evaluate the task at hand. Given that proactiveness can manifest itself in several forms,

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

our benchmark involves recognizing occluded objects and individuals, enhancing image quality, and interpreting coarsely drawn sketches, to name a few. We evaluated 14 open-weight MLLMs on ProactiveBench and found that MLLMs generally lack proactiveness. Critical analyses reveal no clear correlation between model capacity and proactiveness. Adding "hints" in the query to encourage proactive suggestions only results in marginal performance improvement. Surprisingly, including conversation histories introduces negative biases in proposing actions. Overall, the experimental results show that instilling proactiveness in MLLMs is indeed challenging, and we hope that ProactiveBench will positively contribute to building more proactive models. Code and benchmark are available at: `https://anonymous.4open.science/r/ProactiveBench`.

1 Introduction

Making decisions under uncertainty is the hallmark of human intelligence. Studies in neuroscience suggest that meaningful perception of the world arises from dynamic interaction with our environment [18, 22, 24, 56]. Faced with incomplete or ambiguous information, we instinctively generate hypotheses, proactively search for additional clues, and revise our interpretations. This ongoing cycle of inquiry and refinement – central to how humans build coherent understanding of complex situations – has inspired machine vision, particularly in active vision [3, 15, 48].

Ambiguities may arise when a user's query is unanswerable due to false user premises [68] or bad image quality [9], like Fig. 1's example "What is behind the blue blocks?" For such an input, a model can either hallucinate an incorrect answer [37], or it can abstain from answering [21, 66]. We call such models *reactive*. Conversely, a more desirable response from the model is to ask the user to provide additional visual cues by moving the blocks to reveal the hidden object. We refer to such models as *proactive*, since they refine their predictions by asking the user to intervene, which provides additional information. With the growing adoption of multimodal large language models (MLLMs) [5, 32, 76] for complex computer vision tasks in ambiguous settings – such as embodied navigation [36, 57] and autonomous driving [55, 70] – it becomes increasingly important to assess whether MLLMs[1] actively seek additional visual cues like humans.

Despite its relevance, MLLM's proactiveness has received little to no attention in the literature. The only prior work, Liu et al. [42], examined the use of MLLMs for directional guidance, i.e., requesting camera movements in poorly framed images to assist visually impaired people in recognizing objects. Yet, we argue that proactiveness is not limited to directional guidance but can manifest in many other ways. As Fig. 1 shows, MLLMs can also, e.g., ask users to rotate an object, draw additional details to a sketch, or deblur an image. These examples highlight the need to broaden the scope of studying proactiveness in MLLMs across a wide range of tasks and modalities.

To fill this gap we introduce ProactiveBench, a novel benchmark that evaluates MLLMs' proactiveness in multiple scenarios by repurposing seven existing datasets (ROD [31], VSOD [38], MVP-N [64], ImageNet-C [23], QuickDraw [19], ChangeIt [60], and MS-COCO [39]) with different target tasks (e.g., object/sketch recognition, product identification) that require user intervention to predict the correct answer. As Fig. 1 shows, ProactiveBench datasets capture different aspects of proactiveness: (temporal) occlusion removal, camera movement, object movement, image quality enhancement, and asking for details. In total, ProactiveBench contains more than 108k images, leading to a much larger benchmark than [42]. These are grouped into 14k samples featuring 25 proactive suggestions, where each sample (see Fig. 2) contains the starting ambiguous frame, the reference frame with complete information, and all the frames in between. The user intervention results in a new frame with more visual cues based on the model's guidance (termed *proactive suggestion*).

We tested 14 state-of-the-art MLLMs (e.g., LLaVA-OV 7B [32], Qwen2.5-VL 7B [5], and InternVL3 8B [76]) on ProactiveBench, reporting accuracy and number of proposed proactive suggestions before predicting the category. Our experiments suggest that evaluated models lack proactiveness, i.e., are reactive. Thus, they either tend to abstain from answering (saying, e.g., "I don't know") or predict random categories when the visual cues are insufficient, as Fig. 1 shows. Providing hints about proactive suggestions increases their sampling probability, which marginally raises accuracy.

[1] Following prior work [16, 61, 63], we define MLLMs as LLMs fine-tuned to process visual inputs.

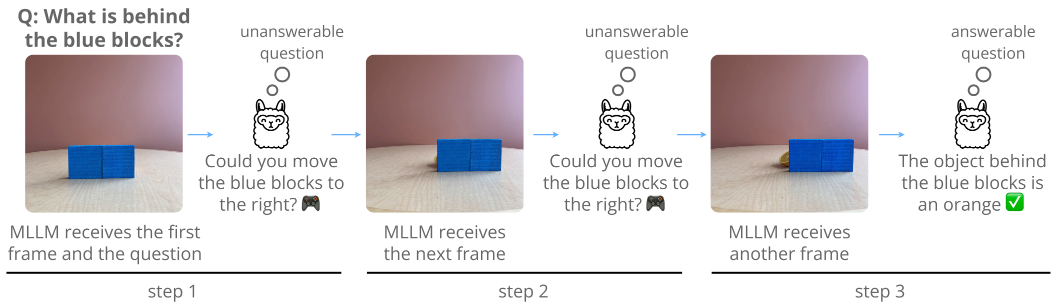


Figure 2: **ProactiveBench evlaution.** At step 1, the MLLM should propose to move the occluding object (proactive suggestion), as the question is unanswerable. ProactiveBench, then, returns a new frame following MLLM's suggestion. Since the model is still unsure, it asks to move the blocks again. Finally, step 3 holds sufficient information, allowing the MLLM to predict the answer.

Interestingly, underperforming MLLMs (e.g., LLaVA-NeXT Vicuna, InternVL3 1B) appear on the surface as more proactive than SOTA MLLMs (e.g., LLaVA-OV 7B, Qwen2.5-VL 7B, InternVL3 8B). A controlled experiment, however, indicates that the higher proactiveness results from a lower rate of abstention on unanswerable questions, not a deep understanding of the problem. Instead, conditioning on the conversation history or few-shot samples increases proactiveness, but at the cost of reduced accuracy. Finally, our results highlight that proactiveness is not an emerging property in MLLMs and must be explicitly elicited, showcasing the challenging nature of ProactiveBench.

Contributions: (i) We formalize and explore MLLMs proactiveness in a wide spectrum, promoting the development of models that can ask user assistance under ambiguity; (ii) We introduce ProactiveBench, a novel open-source benchmark that assesses MLLM's proactiveness in diverse contexts; (iii) Our evaluation of 14 MLLMs on ProactiveBench revealed limited proactiveness and a trade-off between proactiveness and prediction accuracy.

2 The ProactiveBench

This section presents ProactiveBench, formalizing how MLLM proactiveness is evaluated (Sec. 2.1), describing the datasets included in the benchmark and how they were repurposed to assess proactiveness across diverse scenarios (Sec. 2.2).

2.1 Evaluating Proactiveness in MLLMs

We study MLLMs' *proactiveness*, where a model should either answer correctly or suggest how to make a question answerable. Since suggestions may leave questions unresolved (e.g., Fig. 2's central frame), we evaluate proactiveness in a multi-turn setting, allowing the MLLM to interact with the environment over multiple steps. We use the multiple-choice question-answering framework where models select from multiple options, enabling structured evaluation over various turns.

We follow previous works [14, 43], framing the evaluation as a Markov decision process ($\mathcal{S}$, $\mathcal{A}$, π_θ, $\mathcal{R}$), over a finite states space $\mathcal{S}$, a discrete set of actions $\mathcal{A}$, a policy π_θ (the MLLM), and reward $\mathcal{R}$. At step t, the model observes state $s_t \in \mathcal{S}$, which comprises the image $\mathcal{I}_t$ and the valid actions $\mathcal{A}_t \subseteq \mathcal{A}$ (e.g., "wait for the occlusion to disappear", "I do not know", "the answer is dog"). Then, it selects an action a_t conditioned by the question q (e.g., "what is this object?") and the state $s_t = \{\mathcal{I}_t, \mathcal{A}_t\}$. Thus, the transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ is defined by the conditioned policy $\pi_\theta(a_t|q, s_t)$. By selecting a proactive suggestion (e.g., "move the occluding object to the side"), state s_t transitions to s_{t+1}, leading to a new image and a new set of valid actions. Instead, by either abstaining (e.g., "I do not know") or selecting a wrong category (e.g., dog vs cat), the evaluation stops with a wrong prediction. As the environments are discrete, the policy can select proactive suggestions a finite number of times, depending on the datasets, after which the evaluation also terminates with a wrong prediction. Finally, the evaluation also terminates if the model predicts the correct answer. For each MLLM, we report the average accuracy and the average number of proactive suggestions for each dataset. Further details about the environment implementation are in the Appendix.

2.2 Benchmark construction

We introduce seven scenarios to evaluate MLLMs' proactiveness by drawing samples from diverse datasets, which multi-choice options comprise proactive suggestions, the abstain option, and four categories, out of which only one is correct. The Appendix provides further details on each dataset.

Moving occluding objects. We repurposed the ROD [31] dataset by creating samples of 14 frames each, where the two possible suggestions are: moving the occluding object to the left or the right. The environment presents the model with the fully occluded image and the prompt, as Fig. 3 shows. The proactive suggestion asks the user to move occluding objects (e.g., the blue blocks) that obscure the object of interest (e.g., an orange), which the model aims to recognize. The model should ask to move the blocks, and, depending on the visibility of the occluded object, either predict its category or repeat.

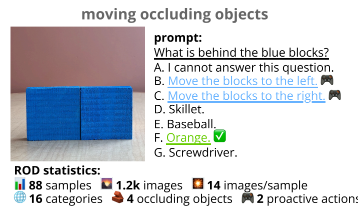


Figure 3: **ROD overview.**

Handling temporal occlusions. We repurposed VSOD [38], a dataset of public event videos with bounding-box annotations for occlusions, to evaluate proactiveness under temporal occlusions. We manually annotated public figures, number of people, and evant type for each frame, which we prompt the model to answer as the target category. Each sample contains on average ∼230 image frames. As Fig. 4 shows, the environment returns the model the most occluded frame of the sample. The proactive suggestion involves the model asking the user to rewind the video or wait for the occlusion to disappear before answering, which in this case is a public figure (e.g., Anne Hathaway).

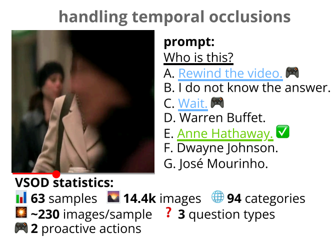


Figure 4: **VSOD overview.**

Handling uninformative views. We repurposed MVP-N [64] – a dataset of fine-grained object categories viewed from multiple angles – to evaluate proactiveness in handling uninformative views by constructing samples with one or more uninformative views followed by an informative one. As Fig. 5 shows, the environment returns the first image from a sample, which is not informative to predict the correct target category. The proactive suggestion of the model is to ask the user to rotate the object (or the camera) until it returns an informative view where the target category can be reliably predicted (e.g., Activia Yogurt Apple).

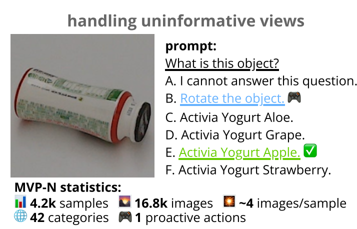


Figure 5: **MVP-N overview.**

Improving image quality. We repurposed ImageNet-C (IN-C) [23] to test proactivess under corruptions, by creating samples where the first and the last images are the most and the least corrupted, respectively. As Fig. 6 shows, the environment returns a corrupted image (e.g., defocus blur), which is not suitable to predict the correct category (e.g., White shark). The proactive suggestion of the model in this case is to conduct image quality enhancements (e.g., deblurring, reducing brightness, removing artifacts, increasing contrast) from a total of eight possible enhancements. In this example, the model should propose to deblur the image to predict the correct category.

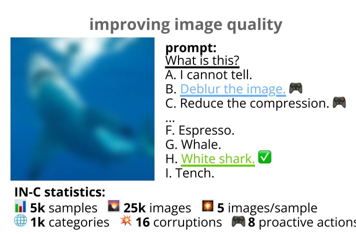


Figure 6: **IN-C overview.**

Asking for visual details. Different from the previous cases, we consider a scenario in which the proactiveness of the model is assessed by its ability to propose proactive suggestions when presented with a partial sketch at input. To this end, we repurposed the QuickDraw (QD) [19] dataset, which contains 345 target categories, by creating samples of rendered PNGs where each image includes one additional stroke compared to the previous one. As more strokes are added, the input image becomes more recognizable to the model. As Fig. 7 shows, the environment first presents an image to the model that does not have enough

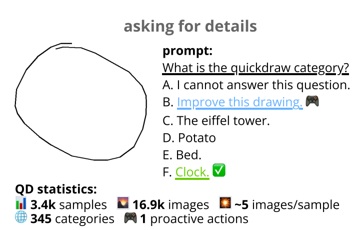


Figure 7: **QD overview.**

detail to recognize the target category (e.g., clock). In this case, the proactive suggestion by the model is to improve the drawing, i.e., adding another stroke.

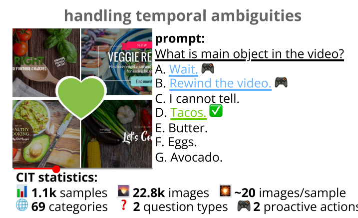


Figure 8: **CIT overview.**

Handling temporal ambiguities. We consider a more challenging scenario in which proactiveness is adjudged by the ability to seek information situated in a different instant of time in a long video. We repurposed the ChangeIt (CIT) dataset [60], consisting of videos of people interacting with objects, by creating samples comprising image frames that depict the objects' transformation (e.g., preparing tacos) from the start to the end. As Fig. 8 shows, the environment presents an input frame where the target category (e.g., tacos) is not visible. Similar to handling temporal occlusions, the proactive suggestion of the model is to ask the user to either rewind the video or wait for the informative moment to appear.

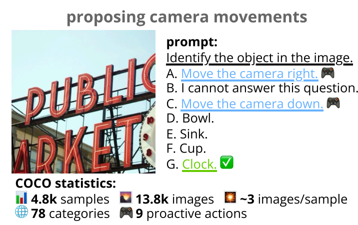


Figure 9: **Overview of MS-COCO.**

Proposing camera movements. Finally, we consider a very practical scenario that prompts the user to spatially move the camera in a 2D plane to obtain more informative visual cues. In detail, we repurposed the MS-COCO [39] images to create samples that contain different crops of the same image, where some crops are more informative than others. As Fig. 9 shows, the environment presents an uninformative crop to the model, where the target category (e.g., clock) is barely visible. The proactive suggestion of the model to the user is to move the camera in one of four cardinal and four ordinal directions, or perform a zooming operation. In this case, the user will be prompted by the model to move the camera towards the right.

3 Experiments

Section 3.1 describes our evaluation protocol, tested models, and metrics used. Then, Sec. 3.2 describes ProactiveBench results, evaluating the proactiveness of several SOTA MLLMs. Finally, Sec. 3.3 reports additional ProactiveBench analysis, evaluating ways to elicit proactive suggestions.

3.1 Experimental setup

Evaluation protocol. For each evaluation step, we feed the MLLM with the user prompt (the question), the current image, and the valid set of suggestions, as described in Sec. 2.1. Therefore, the multi-choice question prompt consists of three parts: the question, optionally a hint to elicit proactiveness, and the options (Sec. 2.2). The conversational history is always discarded from one step to another unless explicitly mentioned (see Sec. 3.3). Finally, as VSOD and ChangeIt consist of video frames, we also tell the model that the visual input is taken from a video.

Tested models. We categorize the chosen MLLMs into high- and low-performing open-weight models. The high-performing ones rank in the top 50 models with less than 10B parameters in the OpenVLM Leaderboard [13]: LLaVA-OV 7B [32], Qwen2.5-VL 7B [5], InternVL3 8B [76], and Phi-4-Multimodal [1]. We choose the low-performing models from well-established MLLMs or that have low parameter count, namely: LLaVA-1.5 7B [41], LLaVA-NeXT 7B [41] with Mistral [26] and Vicuna [8] LLMs, InstructBLIP [11], Idefics3 8B [30], LLaVA-OV 0.5B [32], Qwen2.5-VL 3B [5], SmolVLM2 2.2B [46], and InternVL3 [76] with 1B and 2B parameters.

Metrics. We evaluate each model with two metrics: the accuracy (*acc*) and the number of proactive suggestions (*ps*). We also report the averaged results over the seven scenarios of Sec. 2.2.

Computational resources. All experiments ran using one or two Nvidia A100 GPUs with PyTorch [52], depending on the experiment, and each took around 1-2 GPU hours or less to complete.

3.2 MLLMs results in ProactiveBench

Figure 11 shows models' accuracy (*acc*) using Sec. 2 protocol, comparing it with the oracle setting, where we use a reference frame (i.e., with no occlusions or ambiguity). This comparison's goal is to

Table 1: **MLLMs results on ProactiveBench.** We report the accuracy (*acc*) in percentages (%) and average number of proactive suggestions (*ps*) for all datasets, with global averages in the last column.

	model	ROD		VSOD		MVP-N		IN-C		QD		CIT		COCO		avg.	
		acc	*ps*	*acc*	*ps*	*acc*	*ps*	*acc*	*ps*	*acc*	*ps*	*acc*	*ps*	*acc*	*ps*	*acc*	*ps*
low-perf	LLaVA-1.5-7B [40]	12.5	0.7	41.3	1.3	27.7	0.0	59.4	0.4	43.0	0.5	**70.3**	0.7	67.6	0.4	46.0	0.6
	LLaVA-NeXT-Mistral-7B [41]	0.0	0.0	9.5	0.2	13.7	0.1	53.9	0.2	12.2	0.1	46.3	1.4	49.1	0.0	26.4	0.3
	LLaVA-NeXT-Vicuna-7B [41]	19.3	0.7	25.4	0.9	26.2	0.1	69.2	0.5	22.0	0.7	68.6	0.4	67.7	0.1	42.6	0.5
	LLaVA-OV-0.5B [32]	44.3	2.3	20.6	1.9	**30.7**	0.4	53.6	0.7	45.8	1.1	59.0	0.6	61.0	0.1	45.0	1.0
	Qwen2.5-VL-3B [5]	0.0	0.0	31.7	0.0	25.4	0.0	69.5	0.9	29.1	0.1	58.9	0.2	56.6	0.0	38.7	0.2
	SmolVLM2-2.2B [46]	0.0	0.0	23.8	0.3	26.6	0.0	55.8	0.5	27.0	0.5	64.0	0.3	59.9	0.0	36.7	0.2
	Idefics3-8B [30]	31.8	1.6	31.7	2.1	27.7	0.1	70.0	0.4	27.9	0.5	58.0	0.2	62.2	0.1	44.2	0.7
	InternVL3-1B [76]	**61.4**	2.1	39.7	0.2	29.3	0.3	69.4	0.5	29.2	0.4	61.3	0.1	69.6	0.0	**51.4**	0.5
	InternVL3-2B [76]	1.1	0.0	**49.2**	0.2	30.5	0.1	76.9	0.6	37.9	0.4	69.3	0.3	**77.1**	0.1	48.9	0.2
	InstructBLIP [11]	0.0	0.0	12.7	1.5	12.8	0.1	18.9	0.1	26.0	0.1	47.6	0.1	26.7	0.0	20.7	0.3
high-perf	LLaVA-OV-7B [32]	0.0	0.0	30.2	0.3	24.2	0.0	70.3	0.4	**46.7**	0.3	56.4	0.1	60.0	0.0	41.1	0.2
	Qwen2.5-VL-7B [5]	0.0	0.0	17.5	0.0	24.7	0.0	**78.5**	0.5	34.3	0.1	60.6	0.0	59.5	0.0	39.3	0.1
	InternVL3-8B [76]	0.0	0.0	31.7	0.1	23.2	0.0	75.9	0.3	36.0	0.3	58.3	0.1	67.1	0.0	41.7	0.1
	Phi-4-Multimodal [1]	1.1	0.0	27.0	0.7	29.5	0.0	66.4	0.7	42.3	0.3	66.0	0.2	64.6	0.1	42.4	0.3

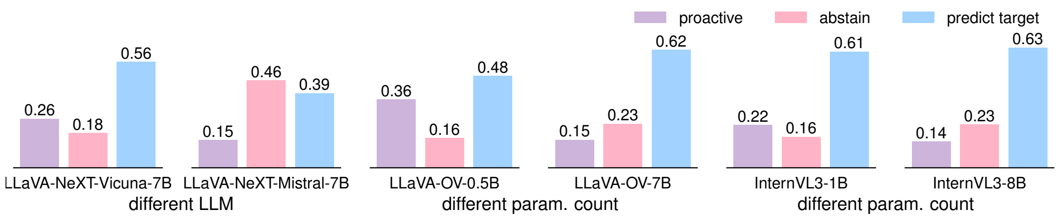


Figure 10: **Action distributions.** While high-performing models (LLaVA-OV 7B and InternVL3 8B) and LLaVA-NeXT Mistral tend to abstain or try to predict the correct answer, the other three models prefer to predict proactive suggestions, which can lead them to the correct action.

disentangle MLLMs recognition ability and their proactiveness. Results correspond to the average performance of all evaluated MLLMs. There is a large discrepancy between the two settings. While in the oracle setting MLLMs score 81.1% on average, they underperform by more than 50% when tasked with navigating to the correct answer through proactive suggestions. The discrepancy is quite stark in the ROD dataset, with models reaching 12.2%, while the oracle counterpart reaches 98.1% on average. The gap closes in IN-C and COCO, but never matches the reference. This demonstrates a severe lack of MLLMs' proactiveness. The full results table is reported in the Appendix.

Table 1 reports models' individual performance on ProactiveBench. Surprisingly, low-performing MLLMs (top half) tend to outperform more powerful models (bottom half). Specifically, InternVL3 1B, InternVL3 2B, and LLaVA-1.5 7B achieved the best, second-best, and third-best accuracy, respectively. Additionally, LLaVA-OV 0.5B and LLaVA-NeXT Vicuna also achieved higher scores than the four high-performing models evaluated. Interestingly, the LLM has an impact on the results, with LLaVA-NeXT Mistral achieving lower performance than its counterpart using Vicuna (26.4% vs 42.6%).

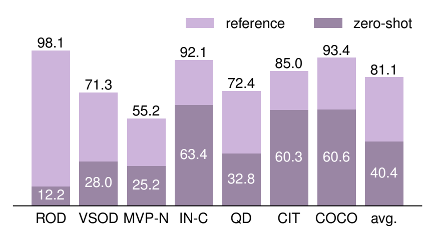


Figure 11: **Results vs. oracle performance (*acc*).** Models underperform by over 50% with ambiguous inputs.

We investigate these unexpected behaviors by visualizing the action distributions of predicting proactive, abstain, and target categories in Fig. 10. Specifically, we compare six MLLMs having different LLMs (i.e., LLaVA-NeXT Mistral and Vicuna) and different parameter counts (i.e., LLaVA-OV 0.5B and 7B, InternVL3 1B and 8B). While the high-performing models (LLaVA-OV 7B and InternVL3 8B) and LLaVA-NeXT Mistral tend to abstain from sampling proactive suggestions, the other three show the exact opposite behavior, i.e., they are more likely to be proactive (more than twice as likely for LLaVA-OV 0.5B). A similar behavior was reported in [67], with LLaVA-NeXT Mistral abstaining more than LLaVA-NeXT Vicuna.

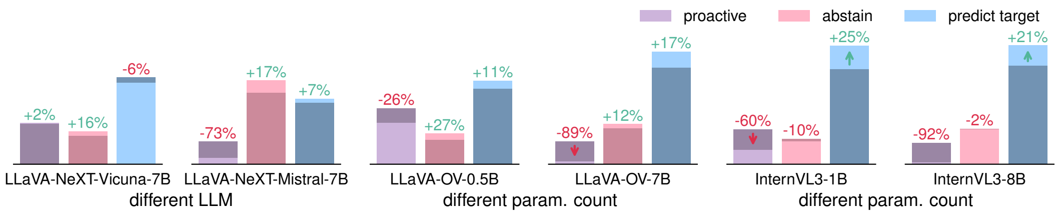
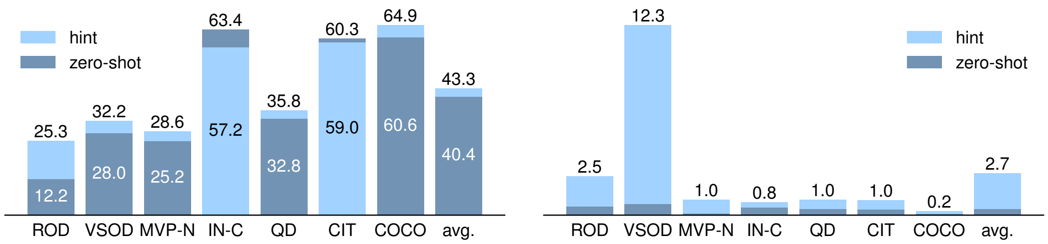


Figure 12: **Action distributions with random proactive options.** Lighter bars describe variations when using random proactive suggestions.



(a) Avg. dataset accuracy per dataset.

(b) Avg. proactive suggestions per dataset.

Figure 13: Performance when **conditioning action sampling with hints.** Results are averaged across all MLLMs. Zero-shot refers to models not prompted with hints.

3.3 Analyzing and eliciting MLLMs proactiveness

Are low-performing MLLMs more proactive than high-performing ones? Following Tab. 1 findings, we investigate why low-performing models seem more proactive than high-performing ones. To answer this question, we replaced valid proactive suggestions with invalid ones chosen from other datasets (e.g., "rewind the video" for QuickDraw). Choosing random options over abstaining indicates random selection and not proactiveness. Figure 12 shows the action distribution in this experiment with the same six models as Fig. 10. Replacing valid proactive suggestions with invalid ones substantially reduces proactiveness for LLaVA-NeXT Mistral, LLaVA-OV 7B, and InternVL3 8B (i.e., -73%, –89%, and -92% relative decrease, respectively). Instead, the other models seem less bothered by the random practice options, with LLaVA-NeXT Vicuna even increases the probability of predicting one (from 26% to 27%). These insights suggest that low-performing models are **not** proactive, but rather they are less prone to abstain [58], preferring unknown answers.

Does hinting boost proactiveness? Explicitly hinting at proactive suggestions may help navigate to the correct answer by eliciting MLLMs' proactiveness. To evaluate this hypothesis, we add environment-specific hints to the prompt (e.g., "Hint: moving the occluding object might reveal what is behind it" for ROD), measuring how it affects the accuracy and number of proactive suggestions. Figure 13b shows that hinting increases the proactive suggestions by 2.3 on average, with a significant boost in VSOD, likely caused by numerous frames. Nonetheless, the accuracy does not improve equally, only increasing by 2.9% on average and even reducing in ImageNet-C and ChangeIt. We also noticed that 14.9% of the time, MLLMs entered "infinite loops" in which they constantly proposed proactive suggestions, failing to predict the correct category. Thus, although hinting increases proactiveness, models may over-exploit proactive suggestions, failing to classify the object even if they stumble across the reference image. Figure 14 further visualizes this by showing how action distributions change using the same six models as Fig. 10. While original distributions (in darker colors) suggest that models infrequently chose proactive options, adding hints completely changes this behavior (especially for LLaVA-NeXT Mistral, LLaVA-NeXT Vicuna, and InternVL3 1B), preferring hinted actions over predicting the correct category. We report the full results in the Appendix.

Does knowledge of the past elicit proactiveness? Section 2.1 formalizes ProactiveBench evaluation, allowing MLLMs to only observe the current state. A key question is whether incorporating previous states and actions into the policy, i.e., $\pi_\theta(a_t|q, s_0, a_0, ..., s_t)$, elicits proactiveness. Thus, in this experiment, we keep the MLLM conversation history, limiting this evaluation to models supporting

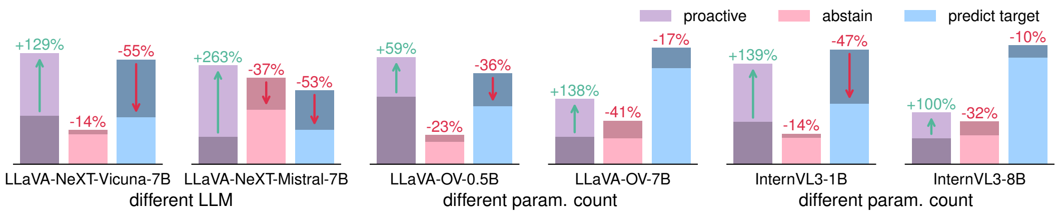
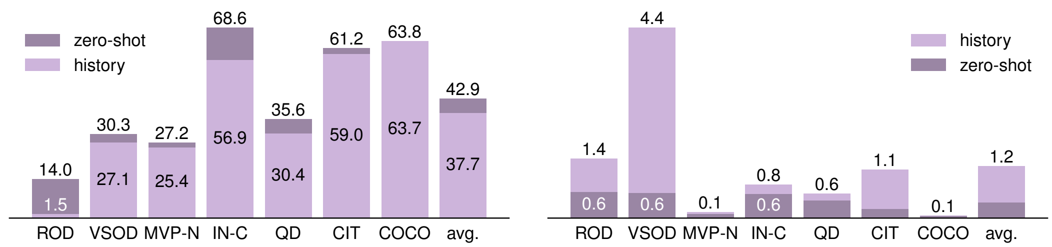


Figure 14: **Action distributions with hints.** Bars describe action distributions with (light) or without (dark) hints in the prompt. Hinting tilts the action distributions in favor of the proactive suggestion.



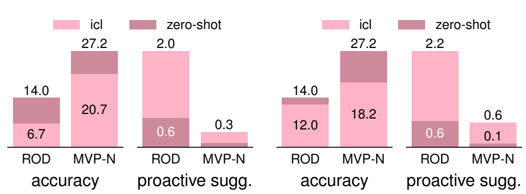
(a) Avg. accuracy per dataset.

(b) Avg. proactive suggestions per dataset.

Figure 15: Performance when **conditioning on conversation histories.** Results are averaged across all MLLMs. Zero-shot refers to models not integrating information about previous states.

multi-image inference. Figure 15 show the results of this experiment. The accuracy drops by 5.2% while the number of proactive suggestions increases from 0.4 to 1.2 on average, compared to the zero-shot case. ROD average accuracy, in particular, is lowered by almost ten times (1.5% vs. 14.0%). Although models are not explicitly "told" to be proactive, like in Fig. 13, past proactive suggestions bias the models towards preferring them. In fact, 9.8% of the time models enter "infinite loops", repeatedly selecting the proactive suggestions until reaching the maximum number of allowed steps. This value is lower compared to 14.9% of hinting, as the first action is always unconditioned; thus, infinite loops occur only if the first action is proactive. Finally, low-performing models prefer proactive suggestions while the other ones are more robust (results are shown in the Appendix).

Do few-shot samples improve proactiveness? We now investigate whether conditioning the policy on a few correct examples elicits proactiveness, improving accuracy. Let $c = (q^c, s_0^c, a_0^c, ..., s_t^c, a_c^c)$ be a conversation example leading to the correct answer a_c^c. We condition the action sampling on m of such examples, $\pi_\theta(a_t|c_0, ..., c_m, q, s_t)$ on ROD and MVP-N, the only datasets supporting automatic few-shot sample generation (image informativeness is annotated), conducted with 1 and 3 shots. Figure 16 shows how proactiveness changes with few-shot in-context learning (ICL). Compared to the previous setting (indicated as zero-shot in the figure), the avg. proactive suggestions increase by 1.4 and 0.2 on ROD and MVP-N, and 1.6 and 0.5 with one and three samples, respectively. Furthermore, the accuracy drops in ROD and MVP-N, resulting in 6.7% and 20.7% with one sample and 12.0% and 18.2% with three. When conditioning ROD experiments with one sample, we notice that models either tend to predict the same category of the ICL example or enter infinite loops. Instead, scaling ICL to three samples helps some high-performing models (i.e., LLaVA-OV 7B and Phi-4-Multimodal) predicting the correct answer. Generally, small MLLMs enter infinite loops, while larger ones tend to abstain. Similarly, in MVP-N, model errors arise either from random guesses, abstentions, or, occasionally, valid proactive sequences ending with incorrect predictions. Full results are shown in the Appendix.



(a) 1 sample

(b) 3 samples

Figure 16: Performance when **conditioning on few shots.** Results are averaged across all MLLMs.

4 Related work

MLLMs. Earlier MLLMs emerged from pioneering efforts to extend frozen LLMs multimodally, such as Frozen [62] and Flamingo [2]. These seminal works convert pre-trained LLMs by injecting visual tokens in the language model's attention layers and fine-tuning them. Subsequent models, like PaLI [7], BLIP [33, 34], LLaVA [40, 41], and InstructBLIP [11], simplified the architecture by forwarding projected visual tokens as input to the LLM, reduced parameter count, and improved performance. Furthermore, LLaVA [40] proposes fine-tuning LLMs using instruction tuning data, improving data efficiency and reasoning capabilities. We focus on benchmarking the proactive capabilities of such models on a broad spectrum of tasks, a currently unexplored research direction.

Benchmarking for MLLMs. While early efforts evaluate MLLMs on visual question answering [4, 20, 47], a second wave of benchmarks relied on numerous tasks requiring reasoning abilities and world knowledge [27, 37, 44, 45, 72]. As recent MLLMs support multiple images and videos as inputs, more complex, multi-input benchmarks have been introduced to evaluate their reasoning capabilities [12, 16, 25, 28, 29, 35, 49, 61, 63]. A parallel effort has emerged in the embodied AI literature, where numerous studies evaluate agents that integrate LLMs [36, 51, 54, 57, 65]. However, none of these works benchmark MLLMs' proactiveness to ambiguous or even unanswerable queries. Related to our work, Liu et al. [42] explores whether MLLM's directional guidance can help visually impaired individuals in capturing images. However, [42] limits the evaluation to a single type of proactive suggestion and to single-turn conversations, not measuring the effectiveness of the MLLM's proposed suggestion. Instead, we investigate models' proactiveness in seven distinct scenarios over multiple turns, enabling a more comprehensive analysis of failure cases and false proactive behaviors.

Active vision improves perception [3] by allowing an active observer to control sensing strategies (e.g., viewpoint) dynamically. Active vision has been extensively studied in view planning (i.e., determining optimal sensor viewpoints) [73], object recognition [6], scene and 3D shape reconstruction [59], and robotic manipulation [10]. To overcome passive systems' drawbacks, Xu et al. [69] introduces an open-world *synthetic* game environment, where agents actively explore their surroundings, performing multi-round abductive reasoning. Although we inherit the underlying spirit of active vision, our work is unique from previous research as: (i) ProactiveBench contains real-world images from diverse and complex scenarios, as opposed to synthetic toy environments, and (ii) unlike self-regulating active vision models, in our case, the observer receives feedback from the MLLM, through proactive suggestions, which results in additional information supplied by the mobile observer. Thus, it fosters a collaboration of the model and the user, which is ideal for human-machine cooperative tasks.

5 Conclusion

This paper presents ProactiveBench, a novel benchmark that evaluates MLLMs' proactiveness by pairing multi-choice questions with visual inputs that require human intervention (e.g., move the occluding object) to make it answerable. We built ProactiveBench by repurposing seven existing datasets designed for different tasks, creating sequences that allow evaluating proactiveness in seven distinct scenarios in a multi-turn fashion. Our findings suggest that existing MLLMs are not proactive and prefer to abstain or predict random categories. Additionally, our analysis shows that hinting at the proactive action improves proactivity, with marginal accuracy gains. Furthermore, conditioning models on conversation histories and few-shot examples negatively biases the action distribution, with lower accuracy scores. These findings highlight ProactiveBench challenges, which we publicly release for future research.

Limitations. While our work is the first to evaluate the proactiveness of MLLMs, we acknowledge a few limitations. First, ProactiveBench is built upon existing datasets, and each evaluated scenario is limited to a single dataset. However, collecting new data is costly, and identifying additional datasets that capture diverse, realistic scenarios remains challenging, particularly because ProactiveBench requires a large number of images per sample and detailed annotations of proactive suggestions. Additionally, our evaluation relies on multiple-choice questions that include proactive options that might encourage proactiveness [17, 50, 53, 71, 74]. Open-ended generated answers would provide a better estimate of MLLMs' proactiveness, but this is highly impractical as multi-turn open-ended conversations typically require human judgment, which is particularly costly for numerous long sequences. For the sake of completeness, we assess proactiveness in open-ended generation using the LLM-as-a-judge paradigm [75] and report the results in the Appendix.

References

[1] Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, et al. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*, 2025.

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022.

[3] John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. Active vision. *IJCV*, 1988.

[4] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, 2015.

[5] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[6] Björn Browatzki, Vadim Tikhanoff, Giorgio Metta, Heinrich H Bülthoff, and Christian Wallraven. Active object recognition on a humanoid robot. In *ICRA*, 2012.

[7] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022.

[8] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. URL `https://lmsys.org/blog/2023-03-30-vicuna/`.

[9] Tai-Yin Chiu, Yinan Zhao, and Danna Gurari. Assessing image quality issues for real-world problems. In *CVPR*, 2020.

[10] Ian Chuang, Andrew Lee, Dechen Gao, M Naddaf-Sh, Iman Soltani, et al. Active vision might be all you need: Exploring active vision in bimanual robotic manipulation. *arXiv preprint arXiv:2409.17435*, 2024.

[11] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *NeurIPS*, 2023.

[12] Song Dingjie, Shunian Chen, Guiming Hardy Chen, Fei Yu, Xiang Wan, and Benyou Wang. Milebench: Benchmarking mllms in long context. In *COLM*, 2024.

[13] Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *ACMMM*, 2024.

[14] Jinhao Duan, Renming Zhang, James Diffenderfer, Bhavya Kailkhura, Lichao Sun, Elias Stengel-Eskin, Mohit Bansal, Tianlong Chen, and Kaidi Xu. Gtbench: Uncovering the strategic reasoning limitations of llms via game-theoretic evaluations. In *NeurIPS*, 2024.

[15] Cornelia Fermuller and Yiannis Aloimonos. Representations for active vision. In *IJCAI*, 1995.

[16] 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. In *ECCV*, 2024.

[17] Manu Gaur, Makarand Tapaswi, et al. Detect, describe, discriminate: Moving beyond vqa for mllm evaluation. *arXiv preprint arXiv:2409.15125*, 2024.

[18] Melvyn A Goodale and A David Milner. Separate visual pathways for perception and action. *Trends in neurosciences*, 1992.

[19] Google. Quick draw!, 2016. URL `https://quickdraw.withgoogle.com/data`.

[20] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, 2017.

[21] Yangyang Guo, Fangkai Jiao, Zhiqi Shen, Liqiang Nie, and Mohan Kankanhalli. Unk-vqa: A dataset and a probe into the abstention ability of multi-modal large models. *T-PAMI*, 2024.

[22] Amanda J Haskins, Jeff Mentch, Thomas L Botch, and Caroline E Robertson. Active vision in immersive, 360 real-world environments. *Scientific Reports*, 2020.

[23] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

[24] Anna Heuer, Sven Ohl, and Martin Rolfs. Memory for action: A functional view of selection in visual working memory. *Visual Cognition*, 2020.

[25] Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhu Chen. Mantis: Interleaved multi-image instruction tuning. In *TMLR*, 2024.

[26] Fengqing Jiang. Identifying and mitigating vulnerabilities in llm-integrated applications. Master's thesis, University of Washington, 2024.

[27] Mehran Kazemi, Hamidreza Alvari, Ankit Anand, Jialin Wu, Xi Chen, and Radu Soricut. Geomverse: A systematic evaluation of large models for geometric reasoning. *arXiv preprint arXiv:2312.12241*, 2023.

[28] Mehran Kazemi, Nishanth Dikkala, Ankit Anand, Petar Devic, Ishita Dasgupta, Fangyu Liu, Bahare Fatemi, Pranjal Awasthi, Sreenivas Gollapudi, Dee Guo, et al. Remi: A dataset for reasoning with multiple images. In *NeurIPS*, 2024.

[29] Jihyung Kil, Zheda Mai, Justin Lee, Zihe Wang, Kerrie Cheng, Lemeng Wang, Ye Liu, Arpita Chowdhury, and Wei-Lun Chao. Compbench: A comparative reasoning benchmark for multimodal llms. In *NeurIPS*, 2024.

[30] Hugo Laurençon, Andrés Marafioti, Victor Sanh, and Léo Tronchon. Building and better understanding vision-language models: insights and future directions. In *NeurIPSW*, 2024.

[31] Ariel N Lee, Sarah Adel Bargal, Janavi Kasera, Stan Sclaroff, Kate Saenko, and Nataniel Ruiz. Hardwiring vit patch selectivity into cnns using patch mixing. *arXiv preprint arXiv:2306.17848*, 2023.

[32] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. In *TMLR*, 2025.

[33] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.

[34] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023.

[35] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *CVPR*, 2024.

[36] Manling Li, Shiyu Zhao, Qineng Wang, Kangrui Wang, Yu Zhou, Sanjana Srivastava, Cem Gokmen, Tony Lee, Erran Li Li, Ruohan Zhang, et al. Embodied agent interface: Benchmarking llms for embodied decision making. *NeurIPS*, 2024.

[37] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *EMNLP*, 2023.

[38] Junhua Liao, Haihan Duan, Xin Li, Haoran Xu, Yanbing Yang, Wei Cai, Yanru Chen, and Liangyin Chen. Occlusion detection for automatic video editing. In *ACMMM*, 2020.

[39] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[40] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.

[41] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *CVPR*, 2024.

[42] Li Liu, Diji Yang, Sijia Zhong, Kalyana Suma Sree Tholeti, Lei Ding, Yi Zhang, and Leilani Gilpin. Right this way: Can vlms guide us to see more to answer questions? In *NeurIPS*, 2024.

[43] Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. Agentbench: Evaluating llms as agents. In *ICLR*, 2023.

[44] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *ECCV*, 2024.

[45] Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 2024.

[46] Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, et al. Smolvlm: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*, 2025.

[47] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *CVPR*, 2019.

[48] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc., 1982.

[49] Fanqing Meng, Jin Wang, Chuanhao Li, Quanfeng Lu, Hao Tian, Jiaqi Liao, Xizhou Zhu, Jifeng Dai, Yu Qiao, Ping Luo, et al. Mmiu: Multimodal multi-image understanding for evaluating large vision-language models. In *ICLR*, 2024.

[50] Aidar Myrzakhan, Sondos Mahmoud Bsharat, and Zhiqiang Shen. Open-llm-leaderboard: From multi-choice to open-style questions for llms evaluation, benchmark, and arena. *arXiv preprint arXiv:2406.07545*, 2024.

[51] Aishwarya Padmakumar, Jesse Thomason, Ayush Shrivastava, Patrick Lange, Anjali Narayan-Chen, Spandana Gella, Robinson Piramuthu, Gokhan Tur, and Dilek Hakkani-Tur. Teach: Task-driven embodied agents that chat. In *AAAI*, 2022.

[52] A Paszke. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.

[53] Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. In *ACL*, 2024.

[54] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *ICCV*, 2019.

[55] Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L Waslander, Yu Liu, and Hongsheng Li. Lmdrive: Closed-loop end-to-end driving with large language models. In *CVPR*, 2024.

[56] Larry Shapiro. The embodied cognition research programme. *Philosophy compass*, 2007.

[57] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *CVPR*, 2020.

[58] Mustafa Shukor, Alexandre Rame, Corentin Dancette, and Matthieu Cord. Beyond task performance: Evaluating and reducing the flaws of large multimodal models with in-context learning. In *ICLR*, 2024.

[59] Edward Smith, David Meger, Luis Pineda, Roberto Calandra, Jitendra Malik, Adriana Romero Soriano, and Michal Drozdzal. Active 3d shape reconstruction from vision and touch. In *NeurIPS*, 2021.

[60] Tomáš Souček, Jean-Baptiste Alayrac, Antoine Miech, Ivan Laptev, and Josef Sivic. Look for the change: Learning object states and state-modifying actions from untrimmed web videos. In *CVPR*, 2022.

[61] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *CVPR*, 2024.

[62] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. In *NeurIPS*, 2021.

[63] Fei Wang, Xingyu Fu, James Y Huang, Zekun Li, Qin Liu, Xiaogeng Liu, Mingyu Derek Ma, Nan Xu, Wenxuan Zhou, Kai Zhang, et al. Muirbench: A comprehensive benchmark for robust multi-image understanding. In *NeurIPS*, 2024.

[64] Ren Wang, Jiayue Wang, Tae Sung Kim, Jinsung Kim, and Hyuk-Jae Lee. Mvp-n: A dataset and benchmark for real-world multi-view object classification. In *NeurIPS*, 2022.

[65] Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? In *EMNLP*, 2022.

[66] Spencer Whitehead, Suzanne Petryk, Vedaad Shakib, Joseph Gonzalez, Trevor Darrell, Anna Rohrbach, and Marcus Rohrbach. Reliable visual question answering: Abstain rather than answer incorrectly. In *ECCV*, 2022.

[67] Robert Wolfe, Isaac Slaughter, Bin Han, Bingbing Wen, Yiwei Yang, Lucas Rosenblatt, Bernease Herman, Eva Brown, Zening Qu, Nic Weber, et al. Laboratory-scale ai: Open-weight models are competitive with chatgpt even in low-resource settings. In *ACM-FAccT*, 2024.

[68] Tsung-Han Wu, Giscard Biamby, David Chan, Lisa Dunlap, Ritwik Gupta, Xudong Wang, Joseph E Gonzalez, and Trevor Darrell. See say and segment: Teaching lmms to overcome false premises. In *CVPR*, 2024.

[69] Manjie Xu, Guangyuan Jiang, Wei Liang, Chi Zhang, and Yixin Zhu. Active reasoning in an open-world environment. In *NeurIPS*, 2023.

[70] Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *IEEE RA-L*, 2024.

[71] Mengge Xue, Zhenyu Hu, Liqun Liu, Kuo Liao, Shuang Li, Honglin Han, Meng Zhao, and Chengguo Yin. Strengthened symbol binding makes large language models reliable multiple-choice selectors. In *ACL*, 2024.

[72] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *CVPR*, 2024.

[73] Rui Zeng, Yuhui Wen, Wang Zhao, and Yong-Jin Liu. View planning in robot active vision: A survey of systems, algorithms, and applications. *CVM*, 2020.

[74] Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. In *ICLR*, 2024.

[75] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. In *NeurIPS*, 2023.

[76] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: This work proposes a benchmark that evaluates proactive reasoning in multimodal large language models. The benchmark is described in Sec. 2, highlighting the repurposed datasets, and model evaluation is reported in Sec. 3, detailing the experiments conducted; thus covering all the abstract claims.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper limitations are described in the "Conclusion" section (Sec. 5).

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The benchmark construction and the evaluation protocol are described in Sec. 2, while Sec. 3 describes the evaluated models, metrics, and prompt used. Moreover, our code and benchmark are attached to this submission, allowing for full reproducibility.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The provided code contains detailed instructions in the README on how to replicate the exact Python environment, download evaluated models, the proposed benchmark, and run all main paper experiments.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The experimental procedure to evaluate models in the proposed benchmark is described in Sec. 3. Furthermore, we also provide the code with the submission.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Although our benchmark allows for computing the statistical significance of each experiment, we follow previous works [16, 49, 63] and did not report these results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The required computational resources are detailed in Sec. 3.1. Further details are in the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We carefully reviewed the NeurIPS Code of Ethics and believe our work conforms.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper's Appendix describes the potential societal impacts of our work.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper does not collect new data but repurposes existing datasets for benchmarking multimodal large language models. Therefore, our work inherits the safeguard measures implemented in such datasets.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We reviewed all the repurposed datasets' and evaluated models' licences and all permits to use and redistribute their data. Moreover, creators of existing datasets and models are properly credited throughout the paper (e.g., Sec. 1). Licences, instead, are reported in the Appendix.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The provided code is well-documented both in the README and with inline comments in source files.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: We did not crowdsource experiments nor conduct research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: We did not crowdsource experiments nor conduct research with human subjects.

    Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: Our goal is to only benchmark multimodal large language models' proactiveness, not proposing a method; thus, it is inapplicable to us.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.