Controlling Multimodal LLMs via Reward-guided Decoding

Oscar Mañas, Pierluca D'Oro, Koustuv Sinha,
Adriana Romero-Soriano, Michal Drozdzal, Aishwarya Agrawal

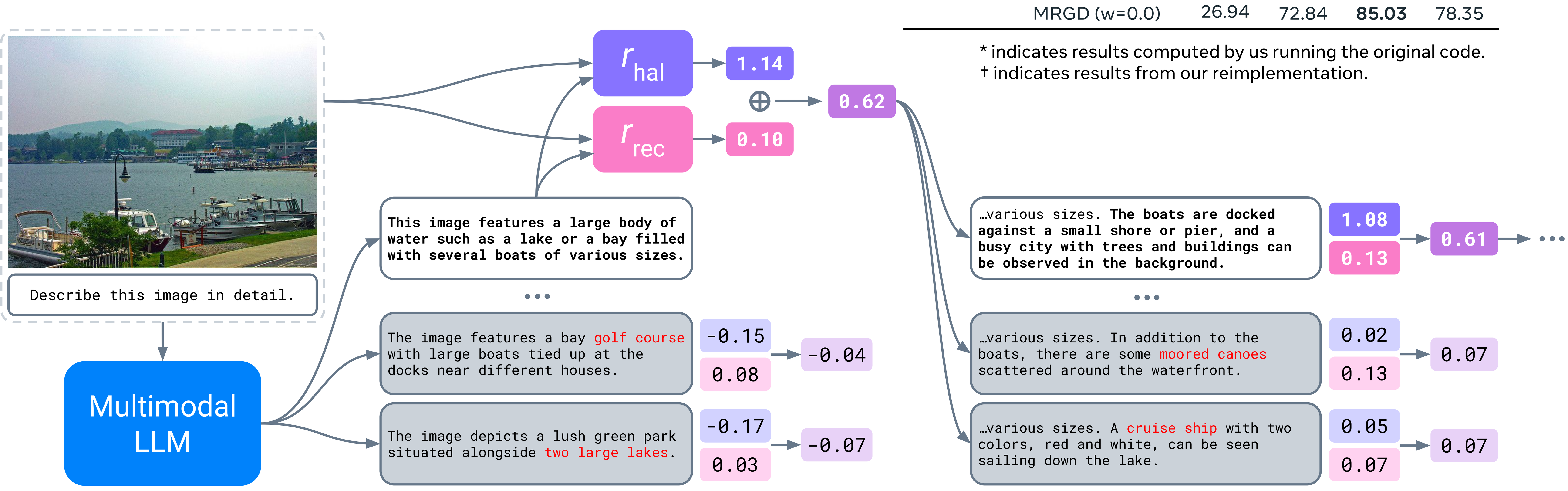
Meta, Mila, Université de Montréal, McGill University

TL;DR

We propose a method for adapting Multimodal LLMs (MLLMs) by controlling their generation through reward-guided decoding, enabling user control over visual grounding and test-time compute in image captioning tasks.

Motivation

- It is becoming increasingly desirable to **steer MLLMs to satisfy diverse user needs**.
- We focus on two user needs:
  - **Control over the trade-off between output precision and thoroughness**
  - **Control over the amount of test-time compute**
- Reward-guided decoding enables **on-the-fly fine-grained controllability**, which is not possible with existing methods (e.g. prompting, fine-tuning).



Method

1. Building multimodal reward models (MRMs)

- **MRM:** (image, instruction, response) → score
- 2 reward models to evaluate object precision (r_{hal}) and recall (r_{rec}).

Learning r_{hal} from preference data

- Fine-tune PaliGemma on multimodal preference data for visual hallucinations: $D = \{x_v, x_q, y^+, y^-\}_i$
- Objective based on the Bradley-Terry model:

$$\mathcal{L}_{RM}(x_v, x_q, y^+, y^-; \theta) = -\log \sigma(r^\theta_{\text{hal}}(x_v, x_q, y^+) - r^\theta_{\text{hal}}(x_v, x_q, y^-))$$

Building r_{rec} from off-the-shelf modules

- Pre-trained object detector (OWLv2), pre-trained word embedding (S-BERT), and POS tagger (NLTK).
- Detect target objects in the input image, extract predicted objects from the generated caption, compute assignment using word embedding similarity, and estimate object recall.

2. Multimodal reward-guided decoding (MRGD)

- **Goal:** guide an MLLM's generation modulating the response according to a combination of reward functions.
- **Algorithm:** sample ***k*** partial completions, select the one with maximum score, and repeat until generating <EOS>.
- Partial responses are evaluated at the end of semantically complete segments, i.e. every ***T*** sentences.
- Reward strength (***w***) can be chosen at test time:

$$s(x_v, x_q, y) = w \cdot r_{\text{hal}}(x_v, x_q, y) + (1 - w) \cdot r_{\text{rec}}(x_v, x_q, y)$$

Experiments

Downstream performance

MRGD (k=30, T=1) either matches or outperforms existing methods to mitigate object hallucinations, while offering greater flexibility.

Model	Decoding	$CHAIR_i$ (↓)	$CHAIR_s$ (↓)	Recall (↑)	Length
Baselines					
LLaVA-1.5$_{7B}$	Greedy	15.05	48.94	81.30	90.12
	BS@10	15.80	52.94	81.48	96.31
Fine-tuning approaches					
POVID	?	5.4	31.8	-	-
CSR	BS@5	7.3	28.0	-	-
Guided decoding approaches					
LLaVA-1.5$_{7B}$	VCD*	15.76	54.18	81.66	102.91
	CGD†	10.44	41.76	80.43	92.26
	MRGD (w=1.0)	**6.83**	**26.38**	74.52	93.28
	MRGD (w=0.5)	7.83	29.68	77.54	94.26
	MRGD (w=0.25)	9.76	37.20	79.96	95.93
	MRGD (w=0.0)	26.94	72.84	**85.03**	78.35

* indicates results computed by us running the original code.
† indicates results from our reimplementation.

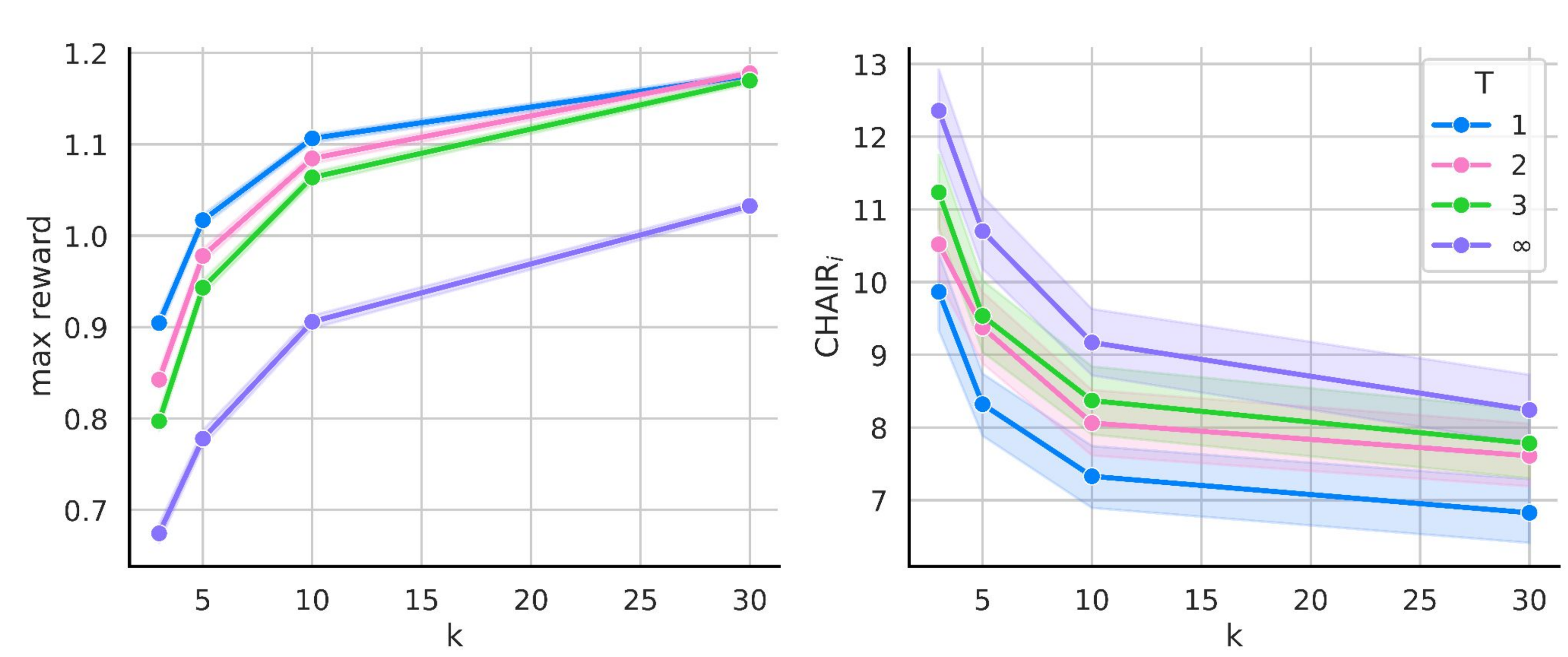
Reward model evaluation

Accuracy: percentage of times the reward model assigns a higher score to the chosen response than to the rejected one.

Average validation accuracy: **77.54%**

Trade-off between visual grounding and compute

Leveraging the reward model to guide the generation more often (lower ***T***) improves compute-efficiency.



Trade-off between object precision and recall

Using more compute by increasing ***k*** improves both object precision (inverse of $CHAIR_i$) and recall, while varying ***w*** modulates the trade-off for a given level of compute.