



Advancing Multimodal LLMs: From Seeing to Understanding and Acting

Zhe Gan

CVPR | Apple | 2025.06.12

How VLMs were Trained A Decade Ago?

Show and Tell: A Neural Image Caption Generator

Oriol Vinyals
Google
vinyals@google.com

Alexander Toshev
Google
toshev@google.com

Samy Bengio
Google
bengio@google.com

Dumitru Erhan
Google
dumitru@google.com

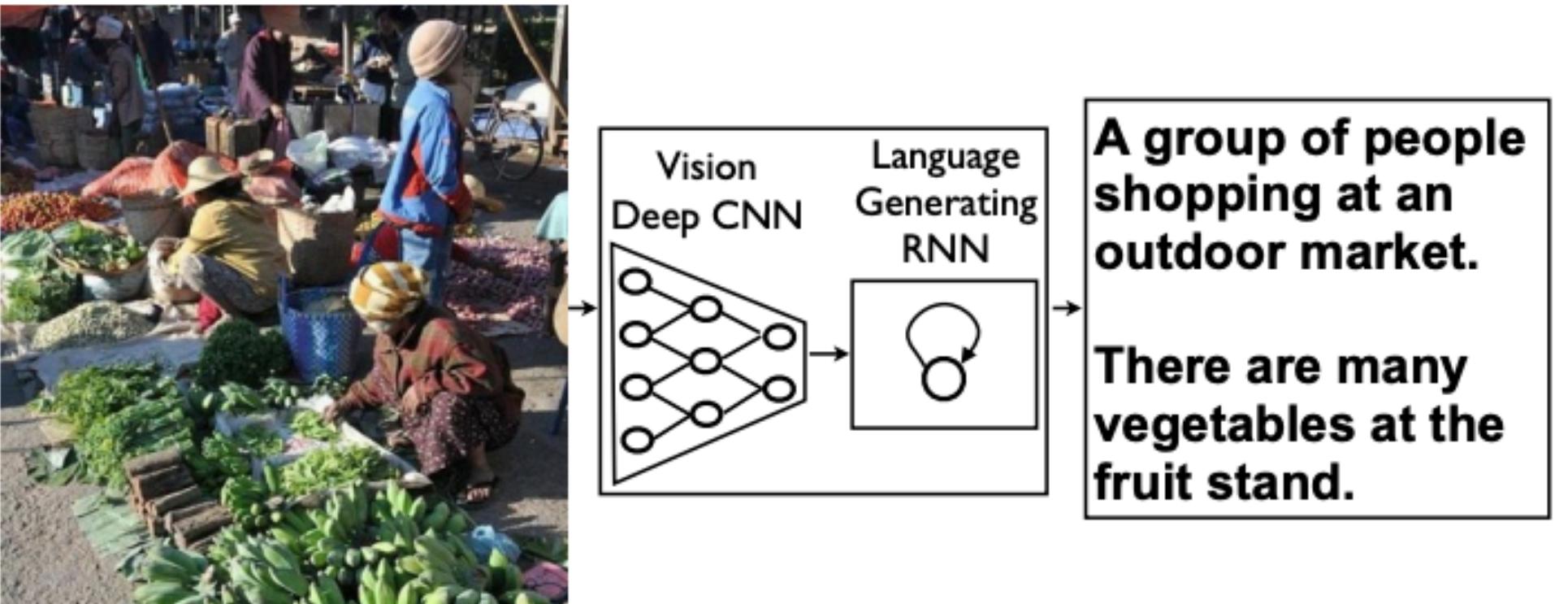


Figure 1. NIC, our model, is based end-to-end on a neural network consisting of a vision CNN followed by a language generating RNN. It generates complete sentences in natural language from an input image, as shown on the example above.

How VLMs are Trained Now?

Qwen2.5-VL Technical Report

Show and Tell: A Neural Image Caption Generator

Oriol Vinyals Alexander Toshev Samy Bengio Dumitru Erhan
Google Google Google Google
vinyals@google.com toshev@google.com bengio@google.com dumitru@google.com

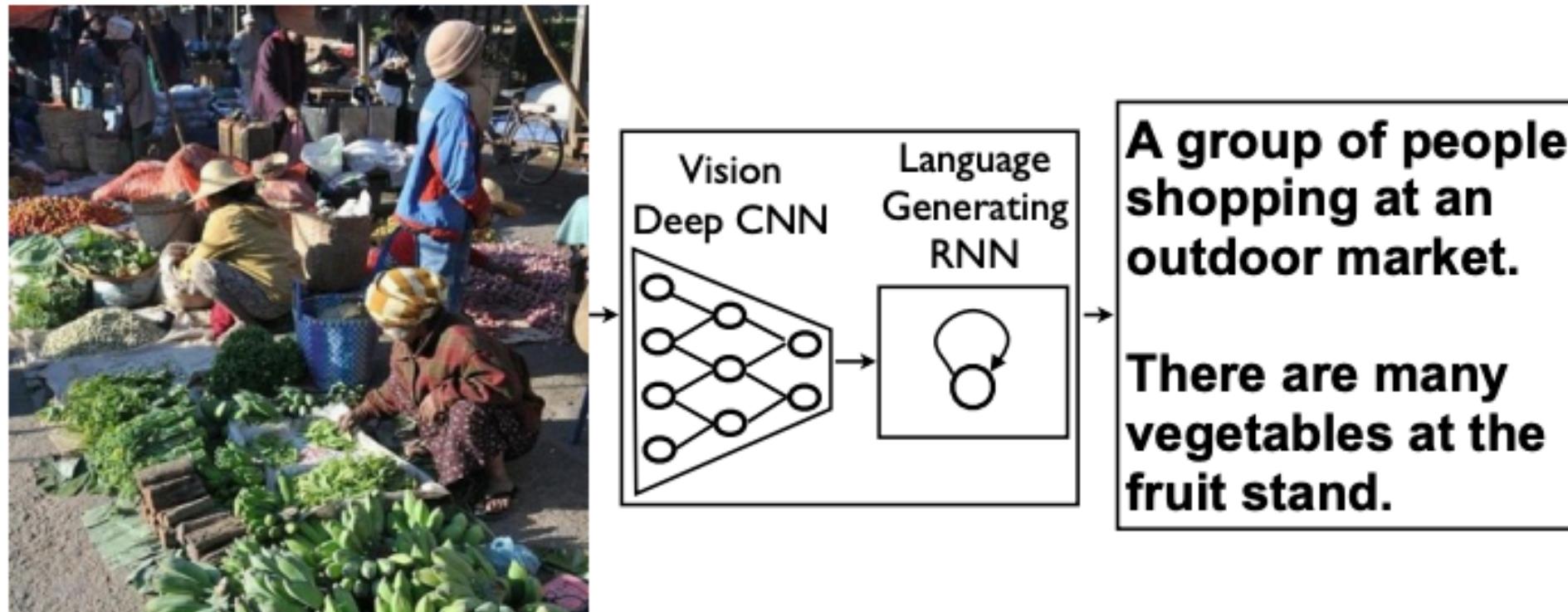
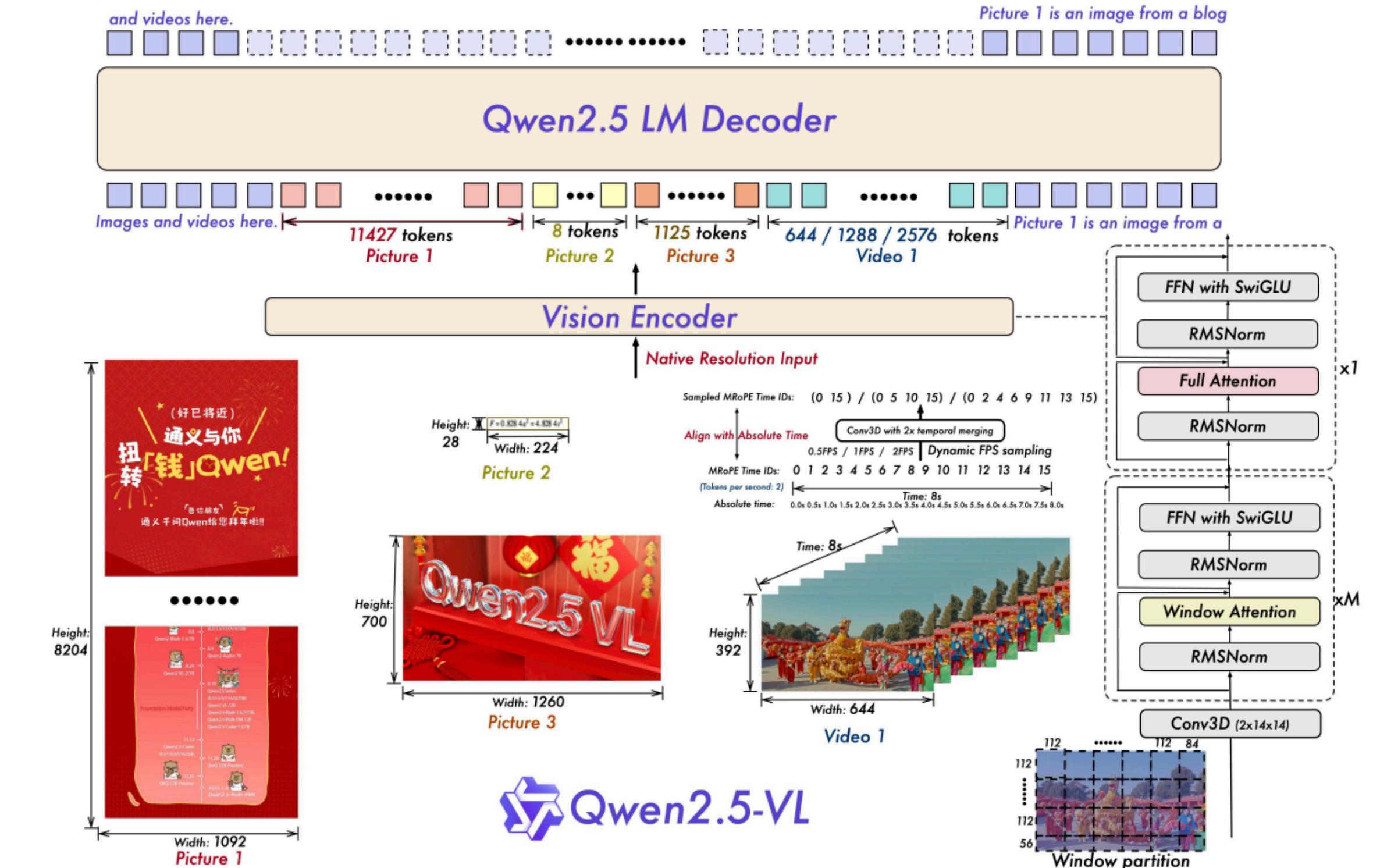


Figure 1. NIC, our model, is based end-to-end on a neural network consisting of a vision CNN followed by a language generating RNN. It generates complete sentences in natural language from an input image, as shown on the example above.

Qwen Team, Alibaba Group

<https://chat.qwenlm.ai>
 <https://huggingface.co/Qwen>
 <https://modelscope.cn/organization/qwen>
 <https://github.com/QwenLM/Qwen2.5-VL>



Qwen2.5-VL

From Show and Tell to Modern Multimodal LLMs

	Show and Tell (2015)	Qwen2.5-VL (2025)
Team	Size of 4 (Research Oriented)	Qwen team (Engineering Heavy)
Image encoder	GoogLeNet (~7M)	ViT with native any-res
Language decoder	LSTM (~13M)	LLM
Parameter size	~20M	72B (~4000 times larger)
Model training	GoogLeNet frozen, LSTM from scratch	Pre-training + Post-training
Training data	ImageNet + COCO	Large volume of data
Capabilities	Short image captions + simple VQA etc.	Knowledge-intensive, text-rich, refer & ground, UI, video, reasoning

Advancing MLLMs: Taking Apple Multimodal Research as Example

Seeing

From CLIP to CLOC

From AIM to AIMv2

Understanding

MM1, MM1.5
MM-Ego, MM-Spatial

Ferret, Ferret 2
Ferret-UI, Ferret-UI 2

SlowFast-LLaVA
SlowFast-LLaVA-1.5

Acting

Generalist Embodied Agents

And more to come...

[1] MM-Ego: Towards Building Egocentric Multimodal LLMs, ICLR 2025

[2] MM-Spatial: Exploring 3D Spatial Understanding in Multimodal LLMs, 2025

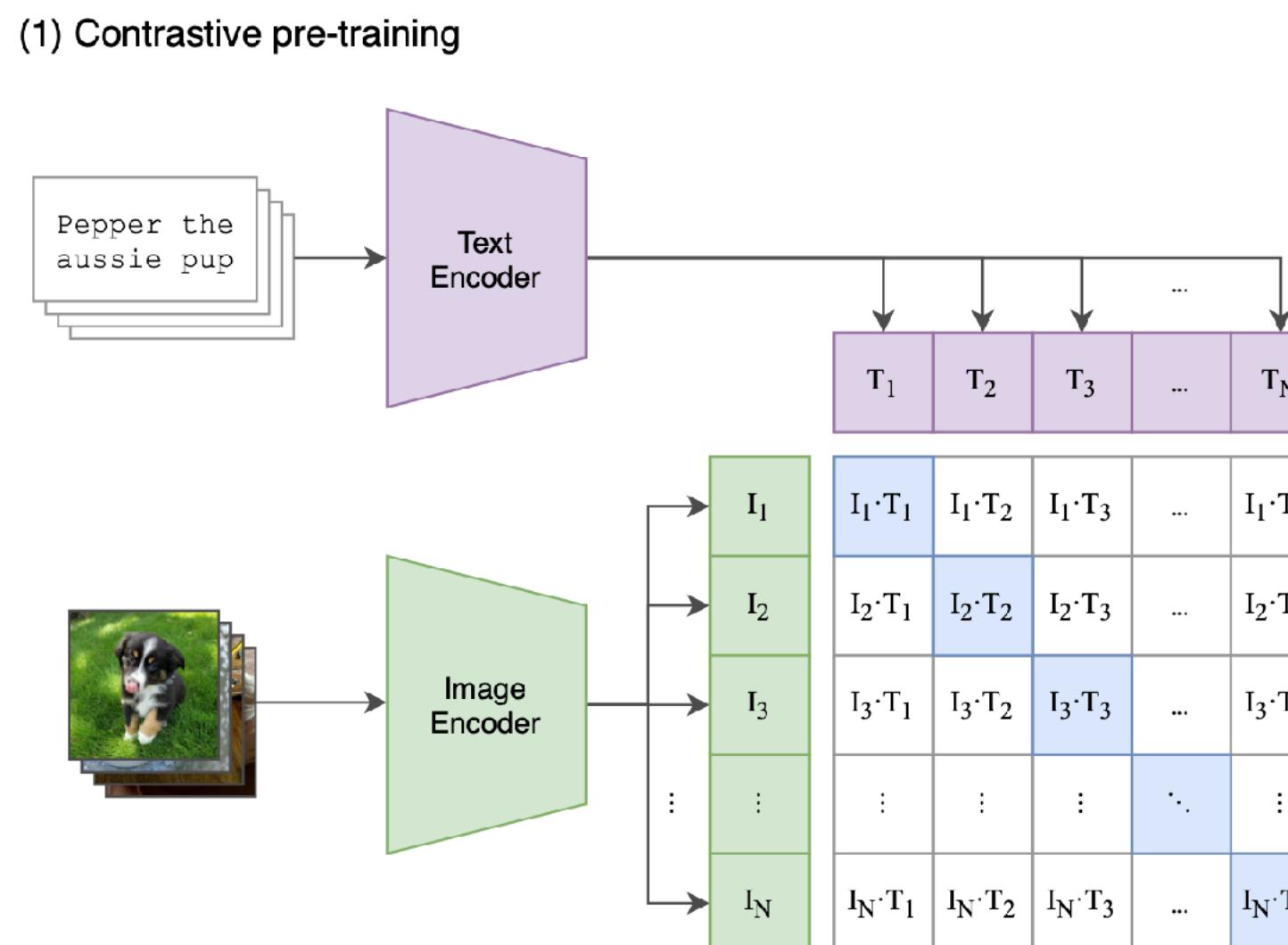
[3] MOFI: Learning Image Representations from Noisy Entity Annotated Images, ICLR 2024

[4] From Multimodal LLMs to Generalist Embodied Agents: Methods and Lessons, CVPR 2025

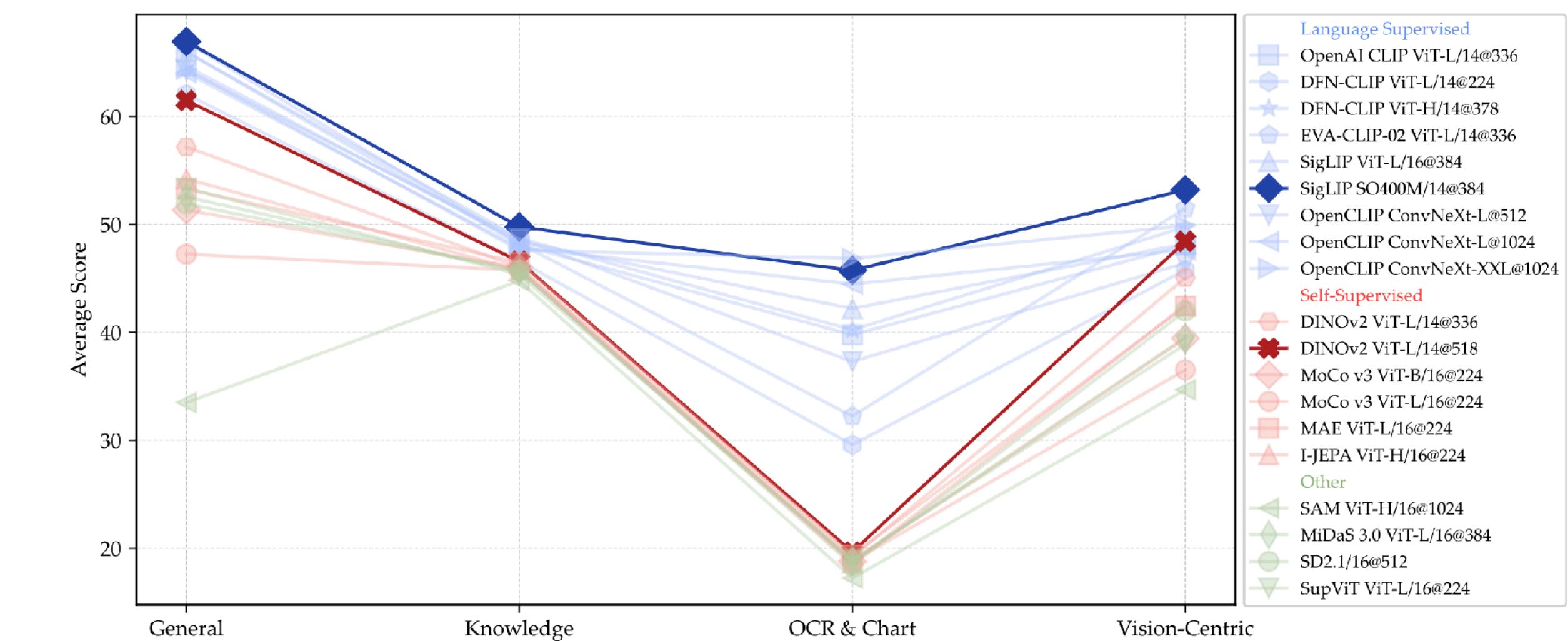
Seeing: From CLIP to CLOC

Can We Do Better than CLIP?

- CLIP has simple design thus appealing scaling properties
- Can we have better image encoder backbones for multimodal LLMs?
- A drop-in replacement for CLIP but with improved **localization** capability



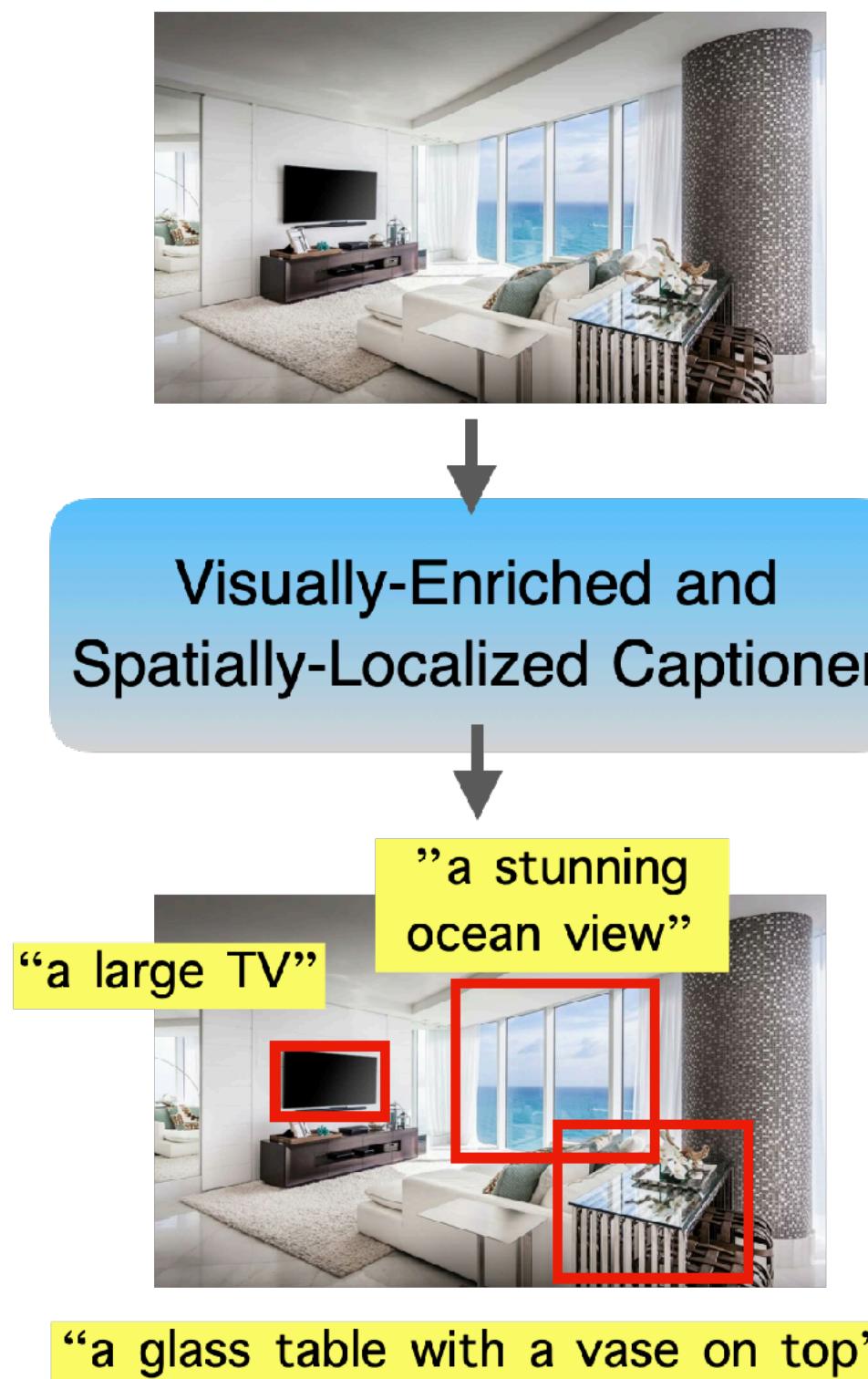
CLIP [Radford et al. 2021]



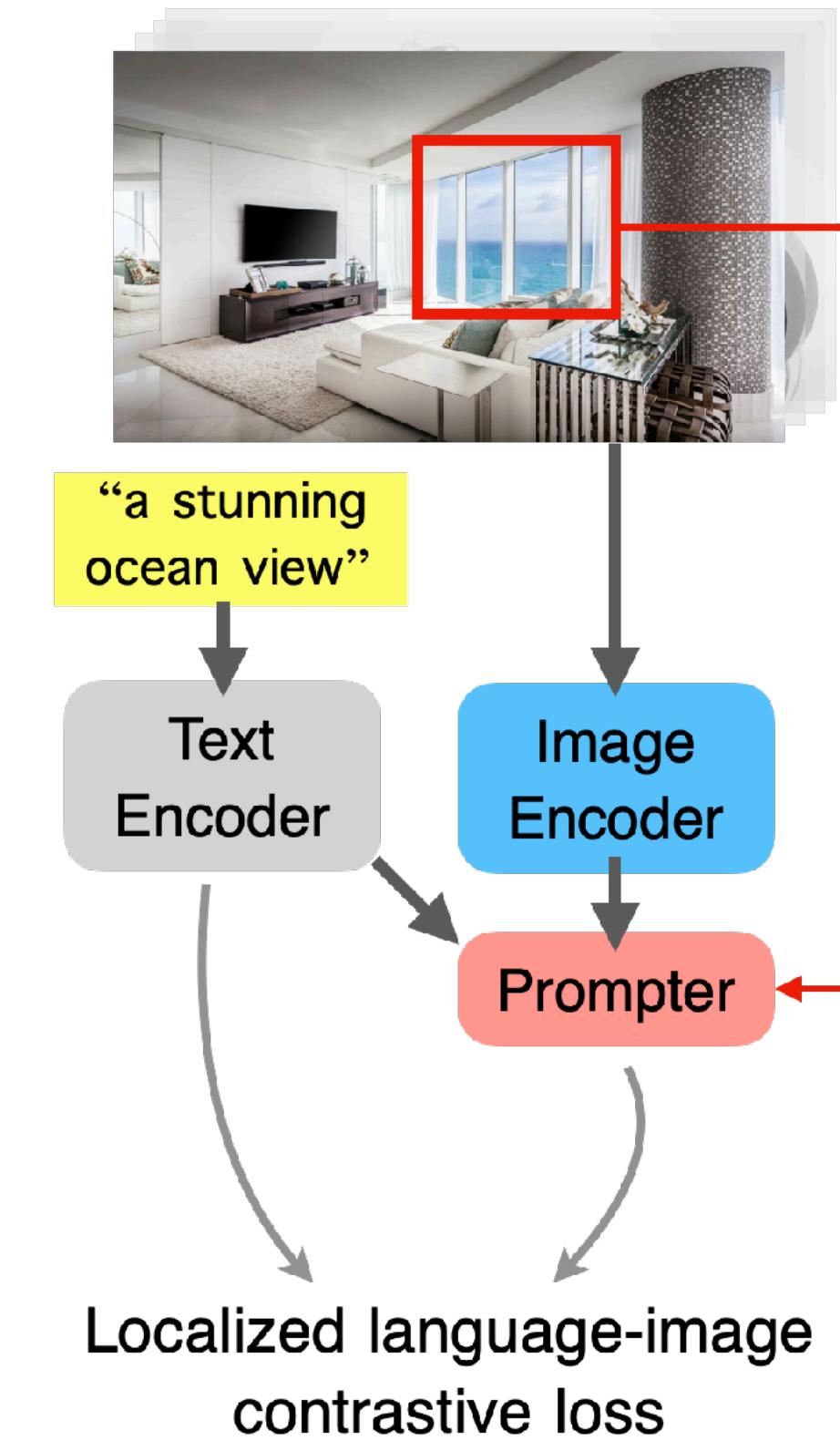
Cambrian-1 [Tong et al. 2024]

CLOC: Contrastive Localized Language-Image Pre-training

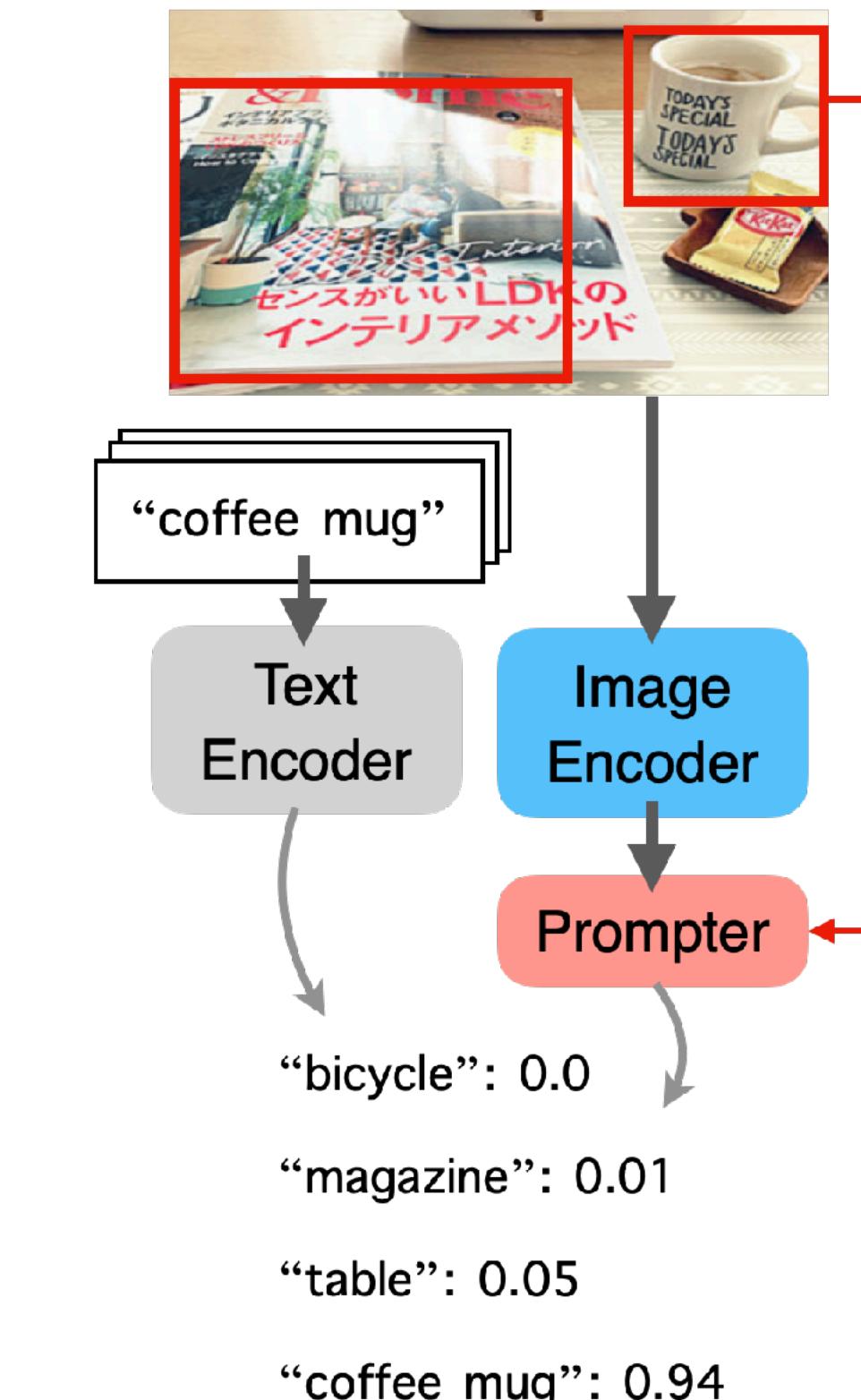
1. Pseudo-Labeling



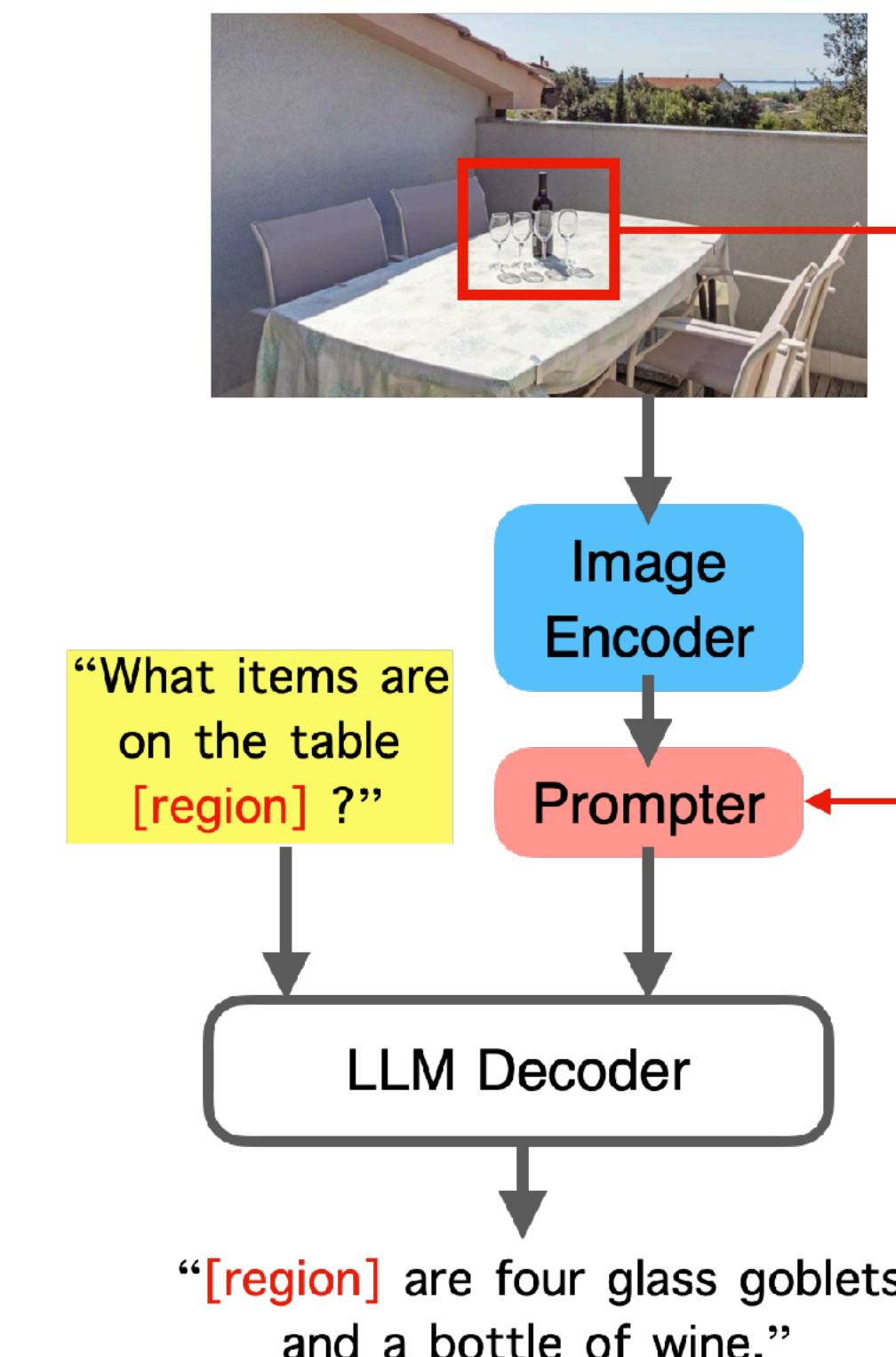
2. Encoder Pre-Training



3a. Region-Text Tasks



3b. MLLM Fine-Tuning



Data: Visually Enriched and Spatially Localized Captioning

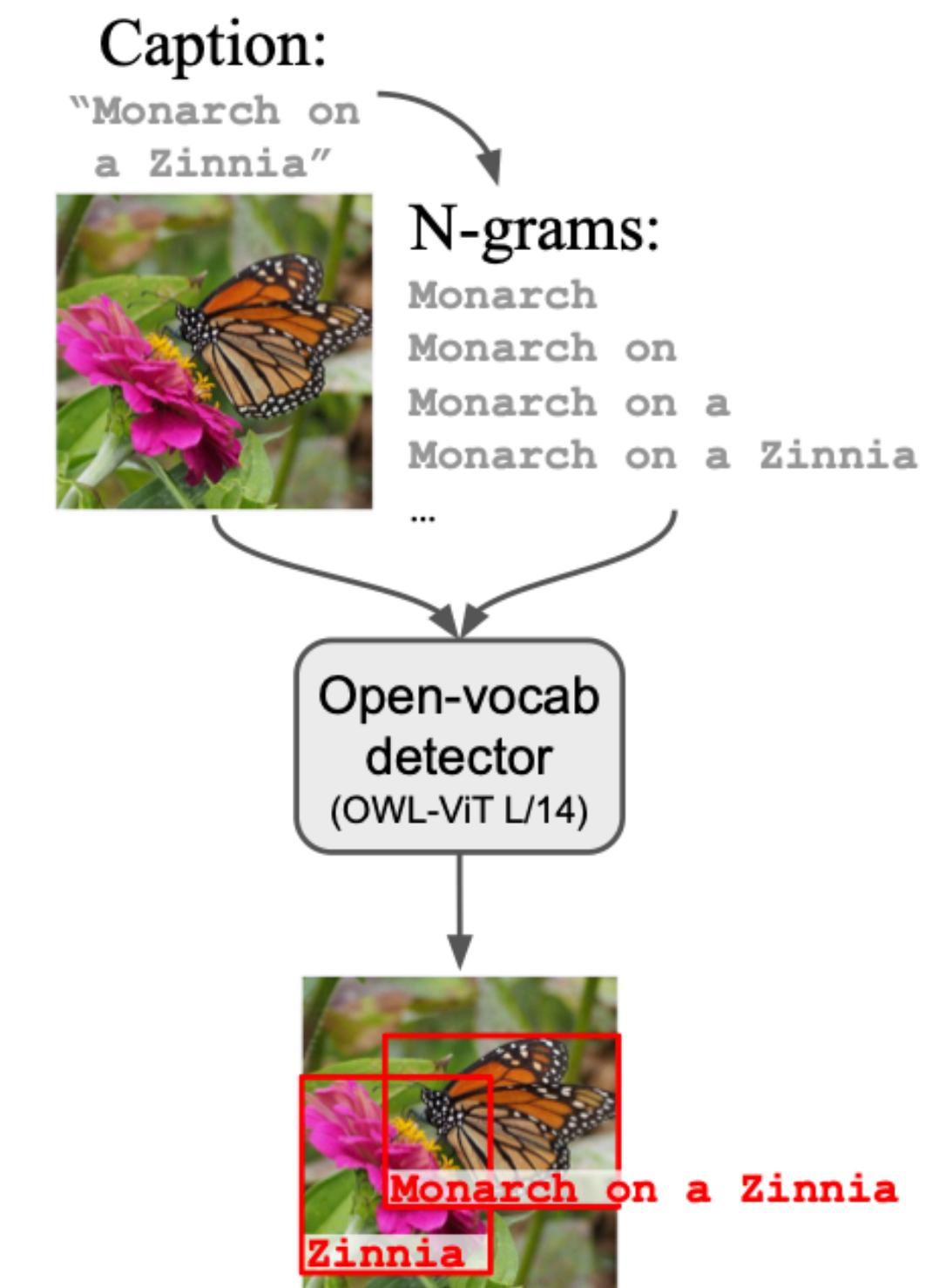


Data: Visually Enriched and Spatially Localized Captioning

- 2B images with 20B image regions for model training

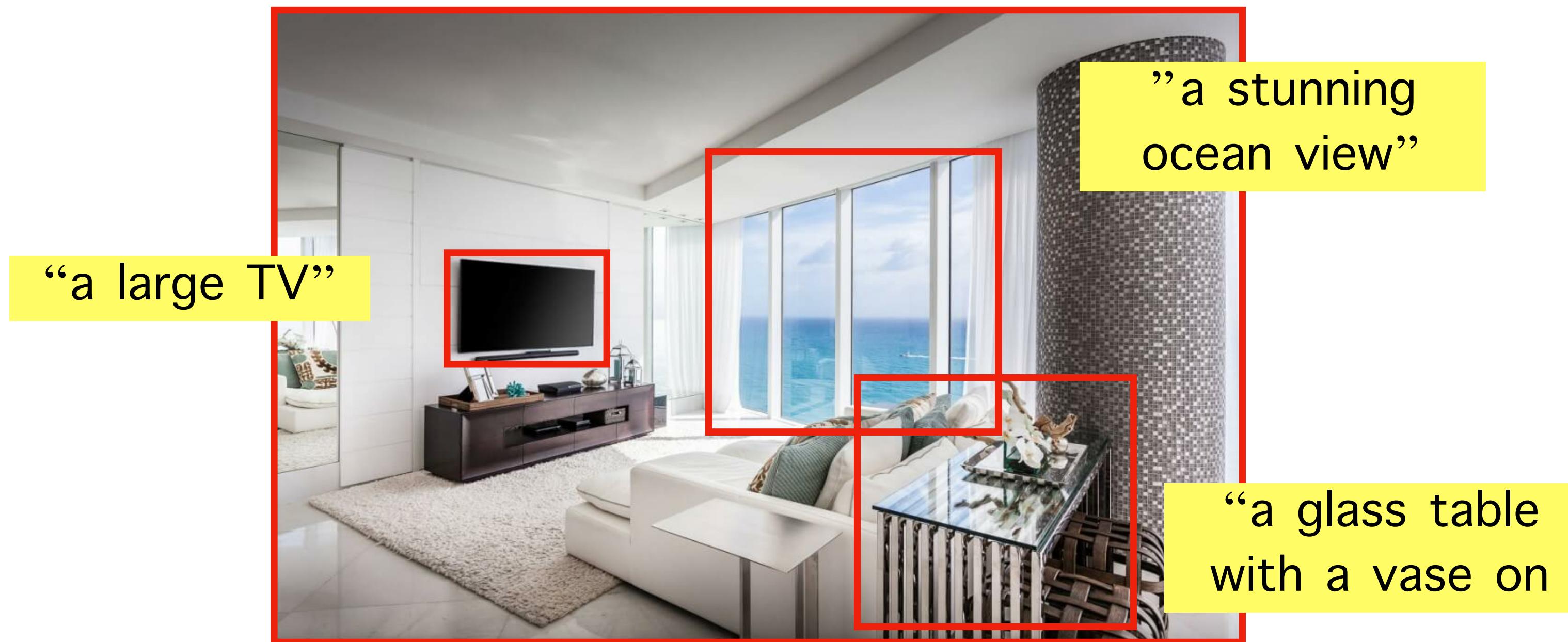
Table 1: Region-text dataset statistics. We summarize the text token length for both images and regions. Partial statistics of the proprietary datasets revealed by their papers. *The 20M subset of GRIT is released at: <https://huggingface.co/datasets/zzliang/GRIT>; we removed the invalid images.

Dataset	# of images	regions per image	image caption length	region text length
Flickr Entities (Plummer et al., 2015)	32K	8.7	–	–
RefCOCO (Yu et al., 2016)	20K	2.5	–	3.6
RefCOCO+ (Yu et al., 2016)	20K	2.5	–	3.5
RefCOCOg (Mao et al., 2016)	27K	2.1	–	8.4
Visual Genome (Krishna et al., 2017)	108K	38.0	–	–
GRIT (proprietary) (Peng et al., 2023)	91M	1.5	–	4.7
GRIT (released, clean) (Peng et al., 2023)*	17M	1.8	17.2	4.6
Florence-2 (proprietary) (Xiao et al., 2024)	126M	5.4	70.5	2.6
OWLv2 (proprietary) (Minderer et al., 2024)	2B	–	–	–
WiT labeled w/ Minderer et al. (2024)	300M	5.1	17.1	3.9
VESL WiT (Ours)	300M	11.6	44.9	2.1
VESL WiT+DFN (Ours)	2B	11.5	35.9	2.1



Promptable Embeddings: How to Obtain Region Embeddings

“A living room in a luxury apartment, featuring a stunning ocean view, a large TV, ...”



a strong image embedding that can be easily transformed into region representations aligned with fine-grained text, given **visual prompts**



“What items are on the table?”

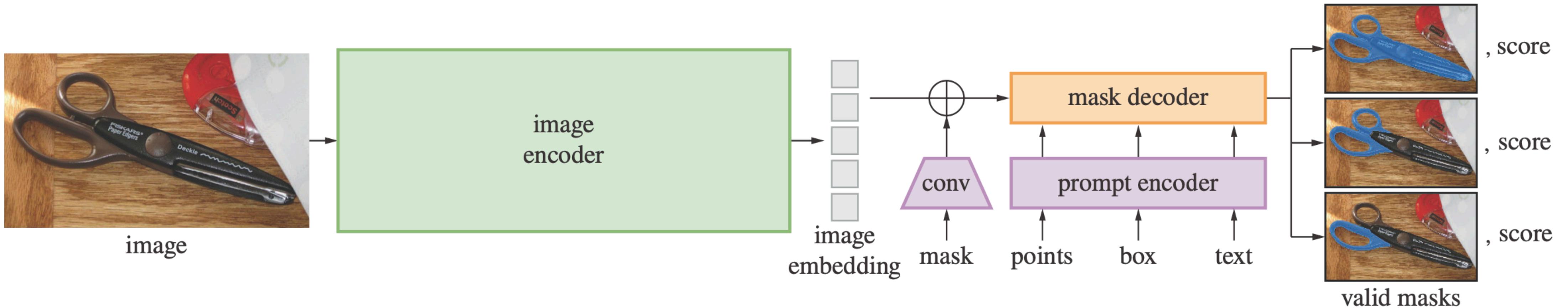
Image Encoder

LLM Decoder

“There are four glass goblets and a bottle of wine.”

Promutable Embeddings: SAM vs CLOC

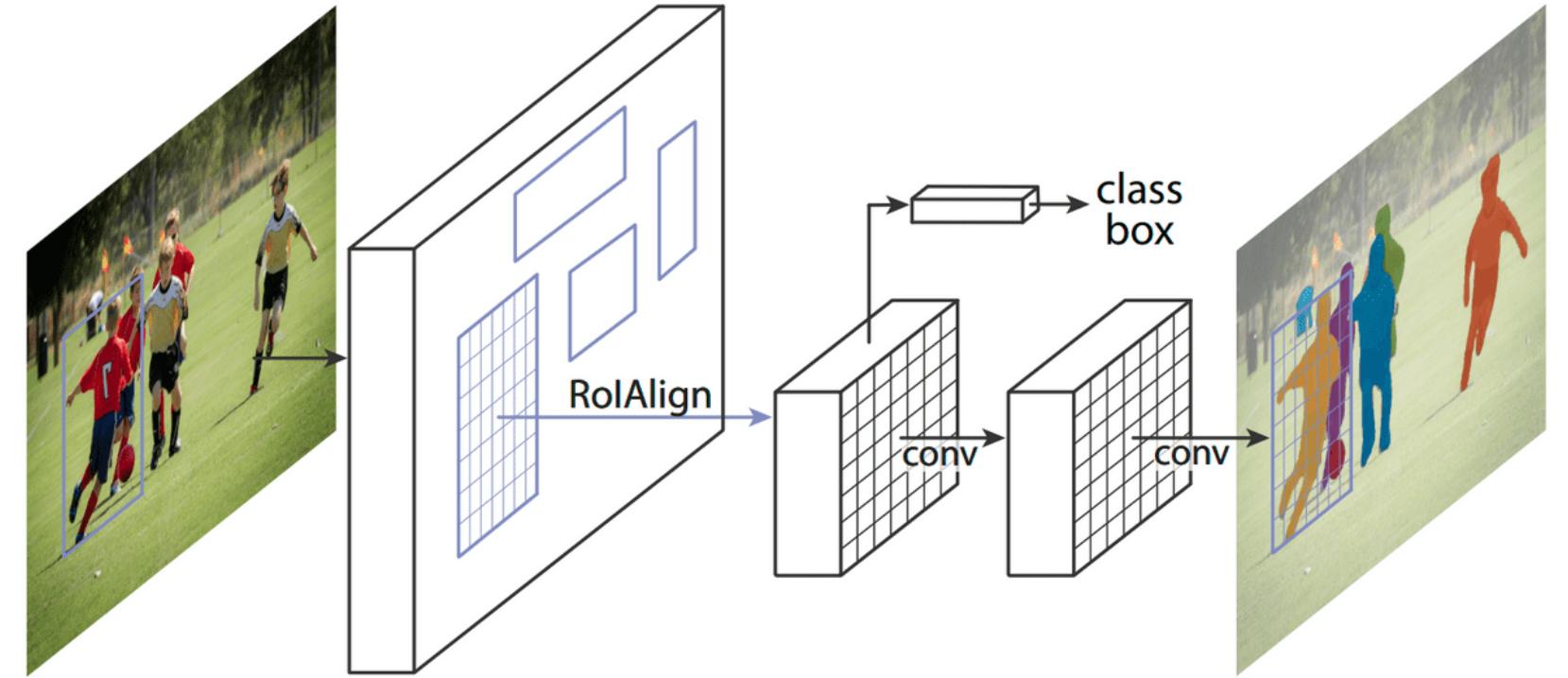
- SAM: a prompt → a mask
- CLOC: a prompt → a region embedding



SAM [Kirillov et al. 2023]

Extracting Region Features with a Promter

- How about RoI-Align?
- Image -> ViT -> spatial feature map -> RoI-Align(box) -> region features



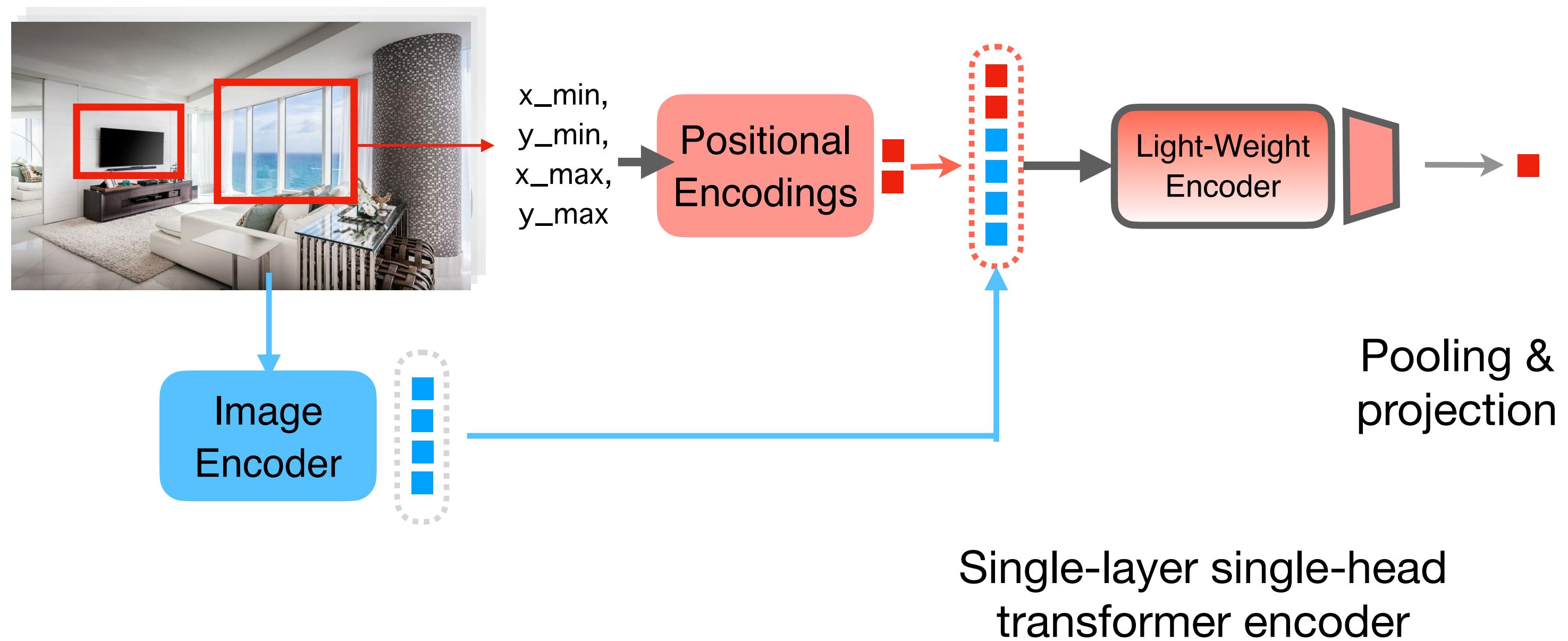
- ViT vs. CNN
- Inductive bias for downstream MLLM
- Noisy bounding boxes



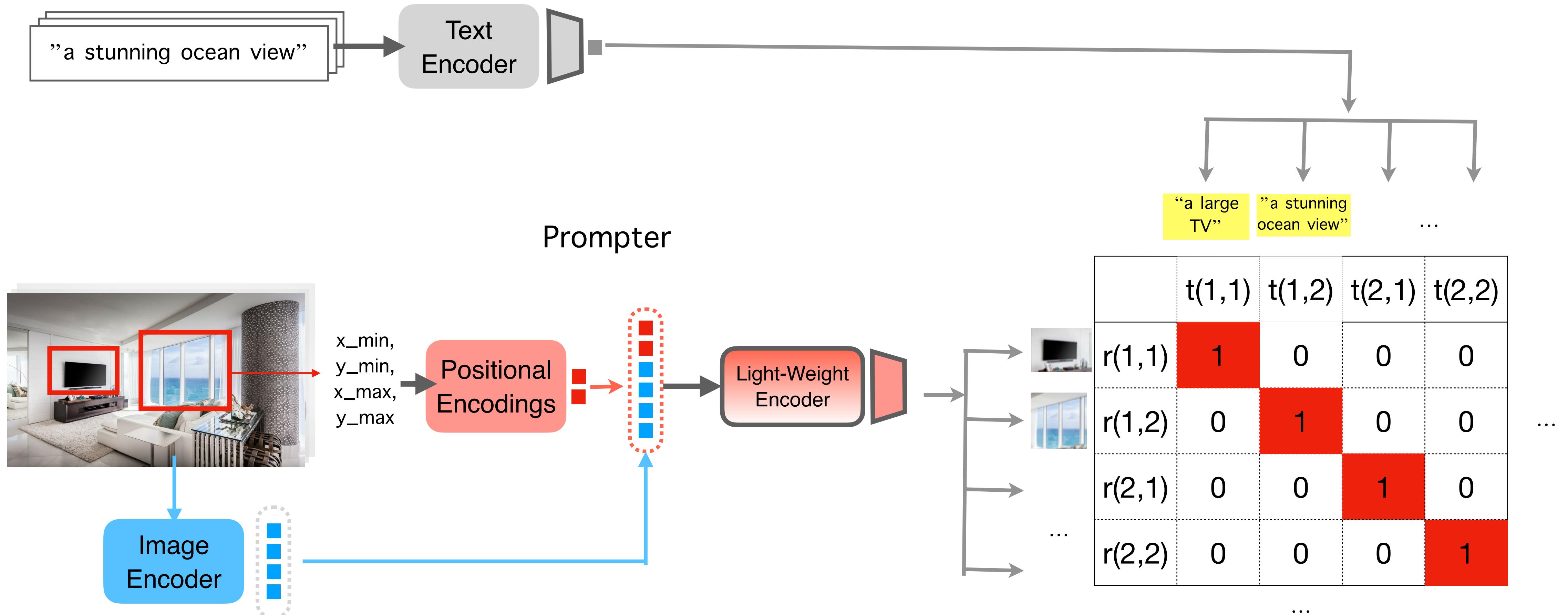
A Simple and Scalable Design for the Prompter

$\text{RegionFeature}(x, \text{box}) =$

$\text{Prompter}(\text{ImageEncoder}(x), \text{box})$



CLOC: A Localized CLIP Training Loss



$$\text{RegionFeature}(x, \text{box}) = \text{Promoter}(\text{ImageEncoder}(x), \text{box})$$

$$\mathcal{L}_{\text{CLOC}}$$

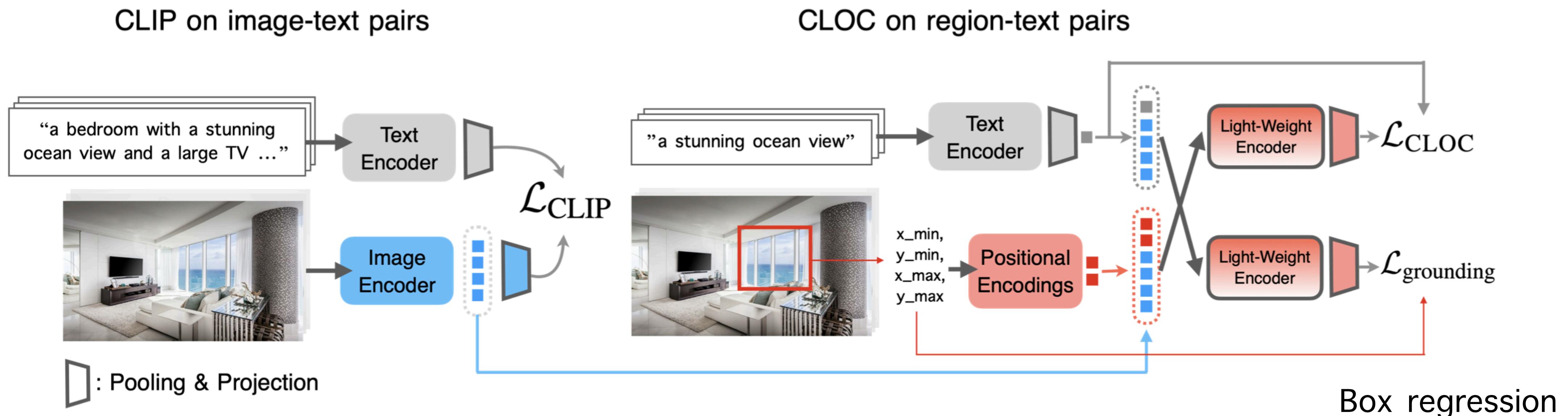
Referring and Grounding in Ferret

- Mimic the concept of referring and grounding for model training
 - Referring: visual prompt → text output
 - Grounding: text input → grounded bbox output



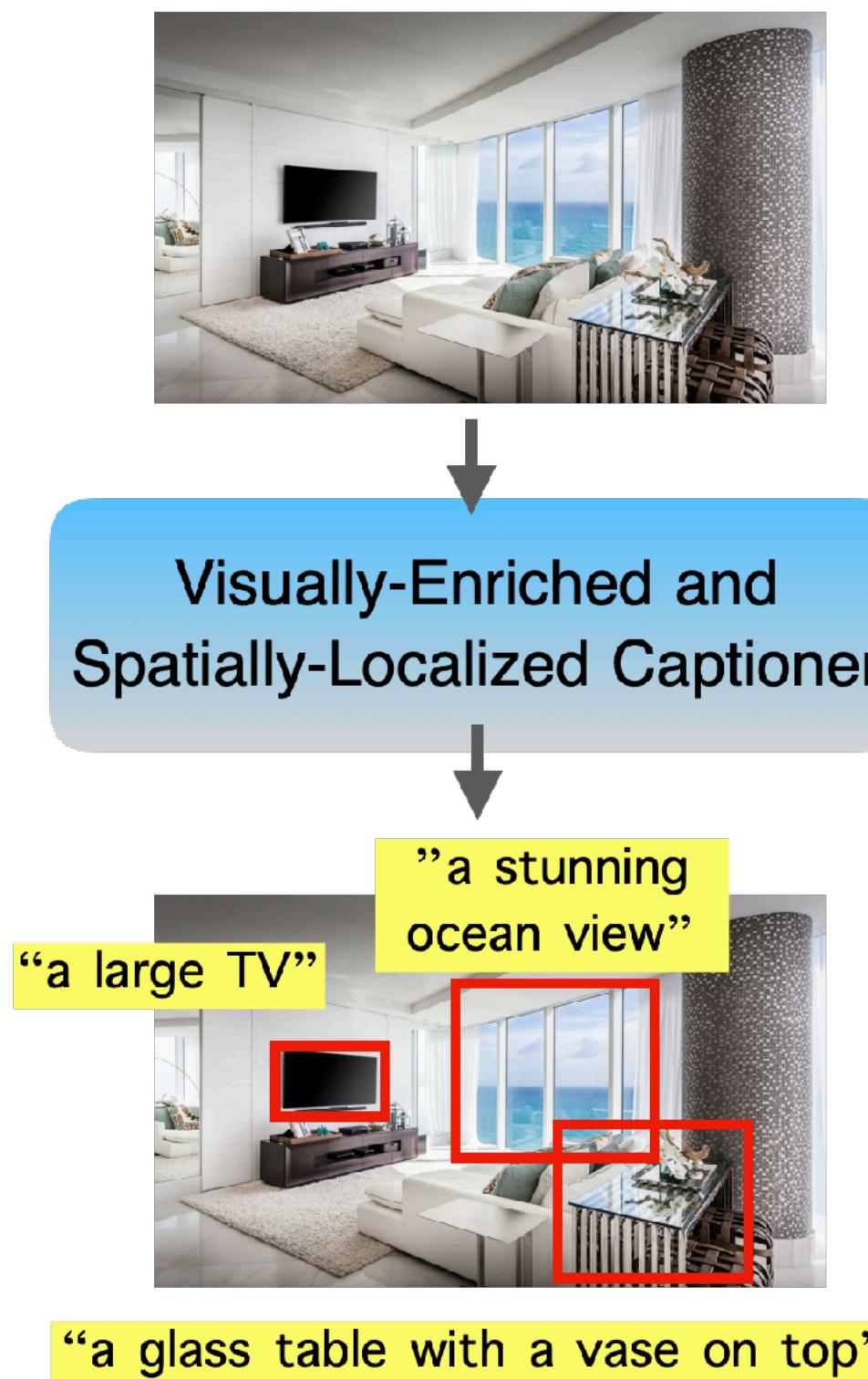
Overall CLOC training

- **Referring**: bbox → region caption (i.e., the CLOC loss)
- **Grounding**: region caption → bbox (i.e., an additional box regression loss)

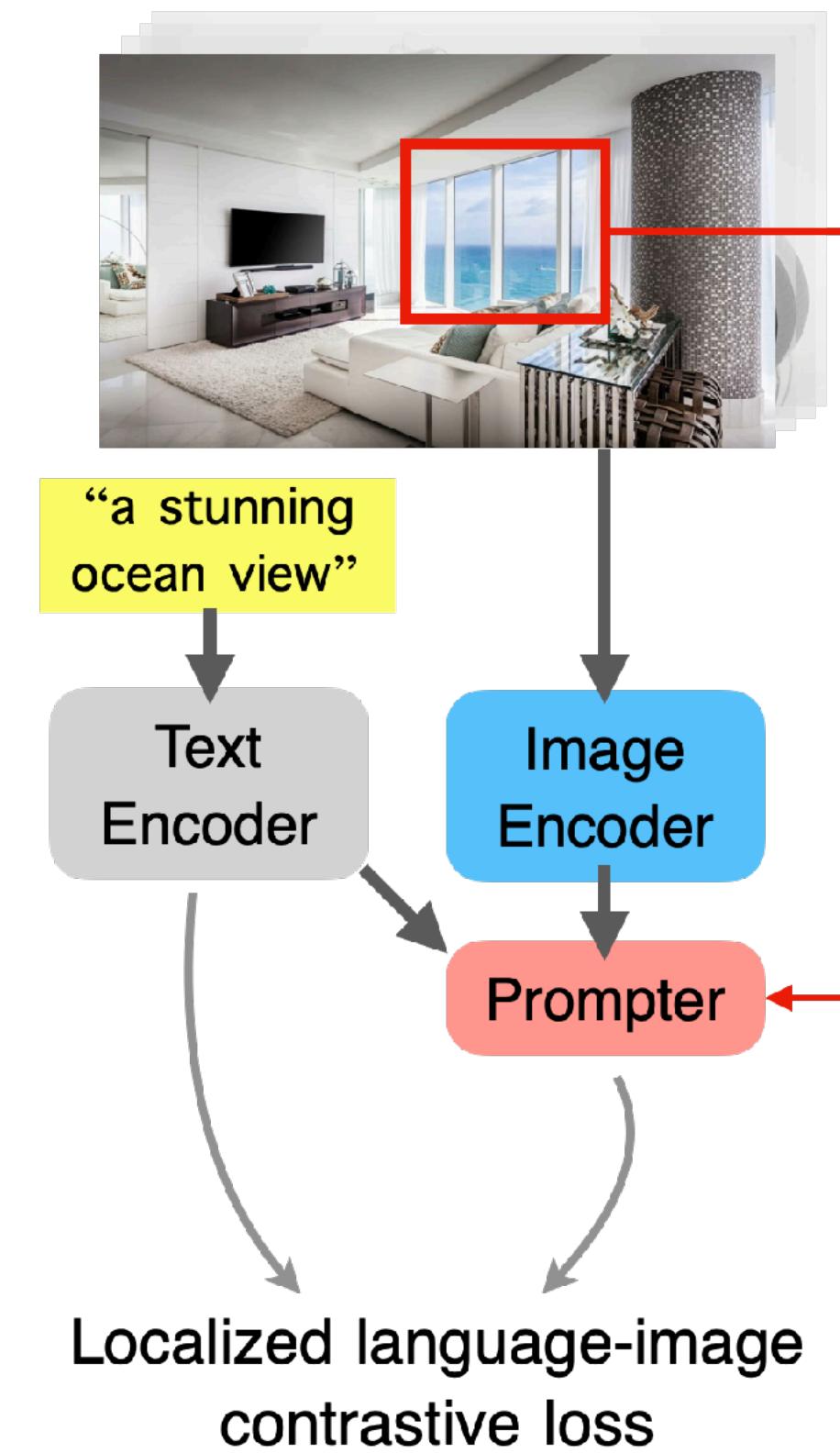


How to Use it for Multimodal LLM?

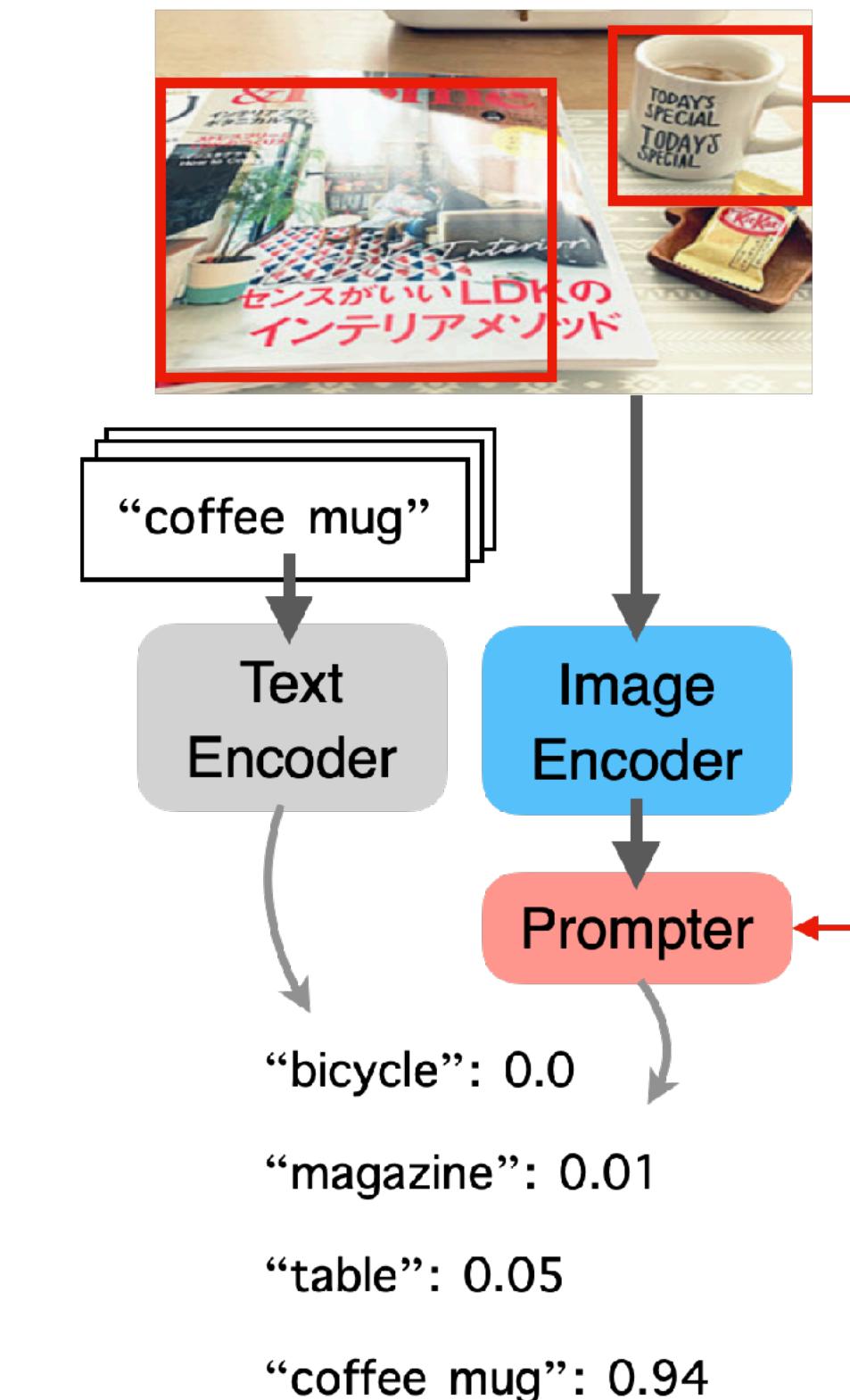
1. Pseudo-Labeling



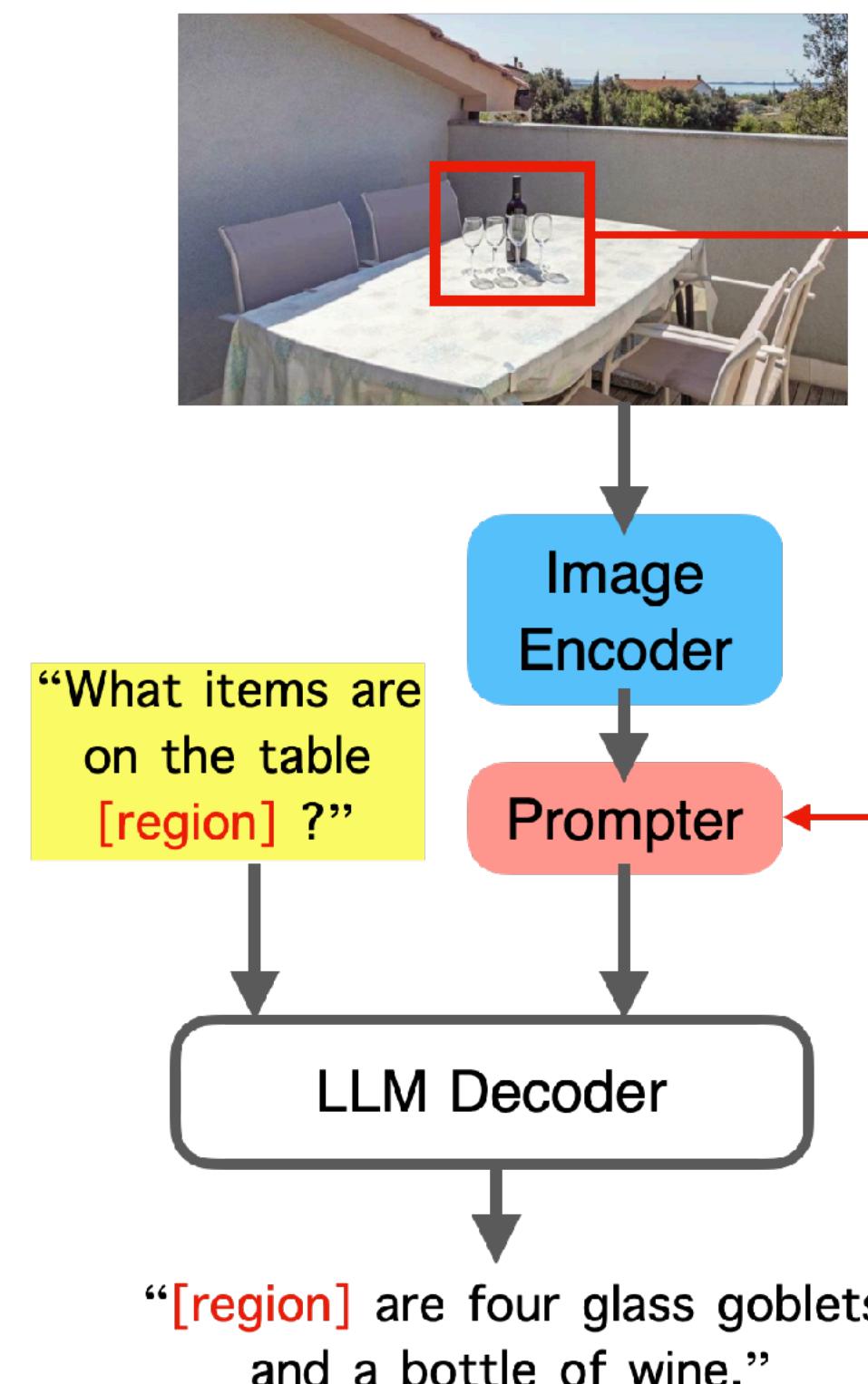
2. Encoder Pre-Training



3a. Region-Text Tasks



3b. MLLM Fine-Tuning



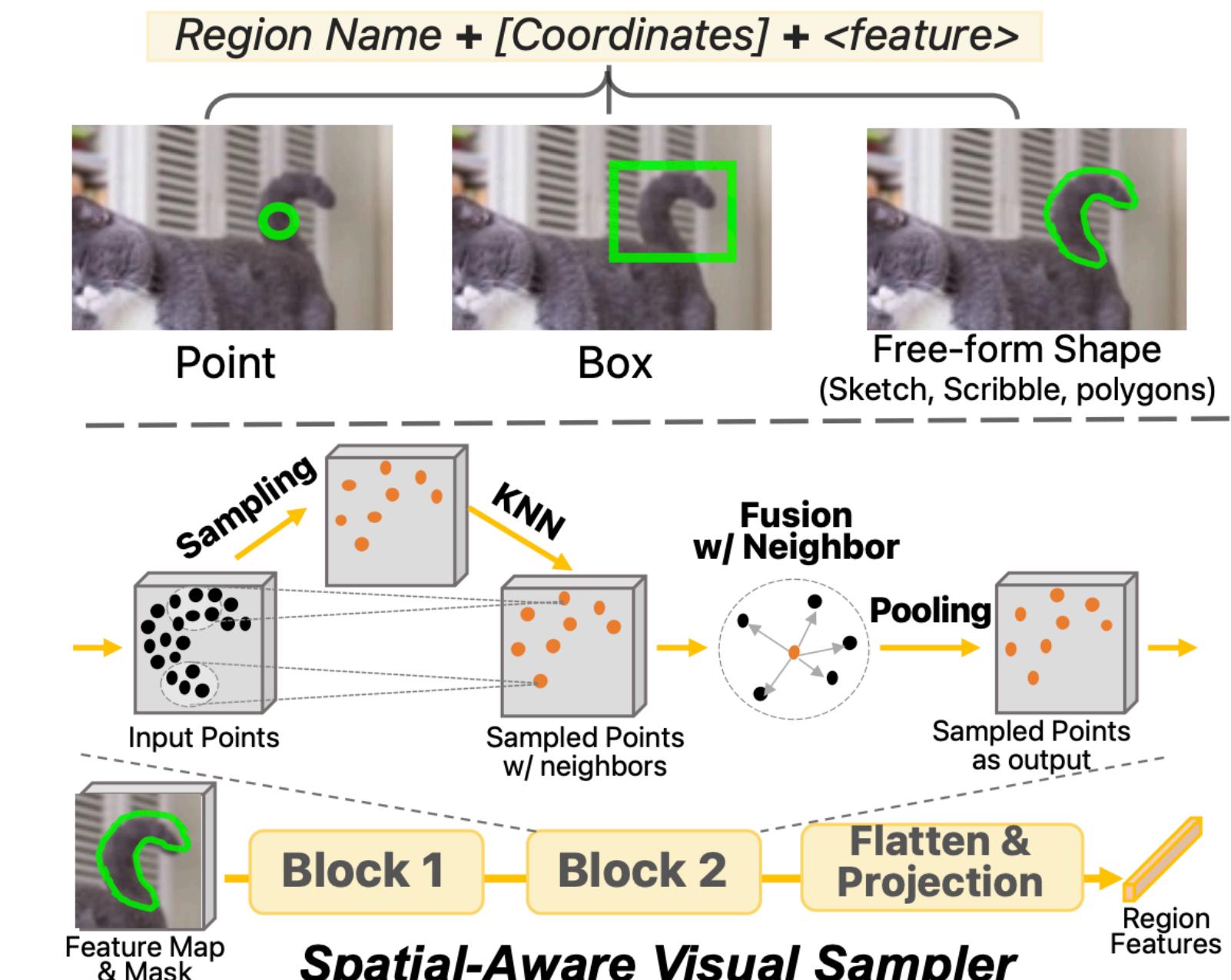
Translates into Improved Referring and Grounding in MLLM

Method	ViT	Region Alignment	# of images w/ region labels	Referring Description	Referring Reasoning	Grounding in Conversation	Avg. (Δ to CLIP)
CLIP	B/16	None	None	47.5	50.3	45.3	47.7
CLOC	B/16	RoI-Align	300M	48.0	48.4	40.0	45.5
CLOC	B/16	Prompter	300M	50.2	55.5	41.5	49.1
CLOC	B/16	Prompter	2B	53.6	53.7	42.2	49.8 (+2.1)
CLOC *	B/16	Prompter	2B	54.8	54.9	44.7	51.5 (+3.7)

Model	Encoder	LVIS box point free-form	RefCOCO val testA testB	RefCOCO+ val testA testB	RefCOCOg val test	Flickr val test	Avg. (Δ to CLIP)
FERRET	CLIP B/16	72.5 56.9 57.2	80.7 84.2 77.1	71.9 76.1 63.7	75.9 76.2	76.2 78.3	72.8
FERRET	CLOC B/16	74.3 56.7 60.2	84.2 87.0 80.0	74.7 80.0 67.0	78.8 79.5	80.0 81.5	75.7 (+2.9)
FERRET *	CLOC B/16	78.9 58.2 61.4	84.4 86.8 78.9	74.0 78.7 65.5	78.0 78.7	80.1 81.4	75.8 (+3.0)

Shikra	OpenAI-CLIP L/14	57.8 67.7 n/a	87.0 90.6 80.2	81.6 87.4 72.1	82.3 82.2	75.8 76.5	-
FERRET	OpenAI-CLIP L/14	79.4 67.9 69.8	87.5 91.4 82.5	80.8 87.4 73.1	83.9 84.8	80.4 82.2	80.8
FERRET	CLIP L/14	78.7 66.9 70.2	88.0 90.4 83.5	80.1 85.8 73.3	82.8 83.4	79.0 80.1	80.2
FERRET	CLOC L/14	81.6 67.9 69.9	89.0 91.0 84.7	81.4 86.8 74.7	84.0 85.2	82.3 83.3	81.7 (+1.5)
FERRET *	CLOC L/14	79.8 67.9 69.1	88.2 91.1 84.5	80.6 86.7 73.9	84.8 85.1	82.4 83.5	81.4 (+1.2)

Hybrid Region Representation



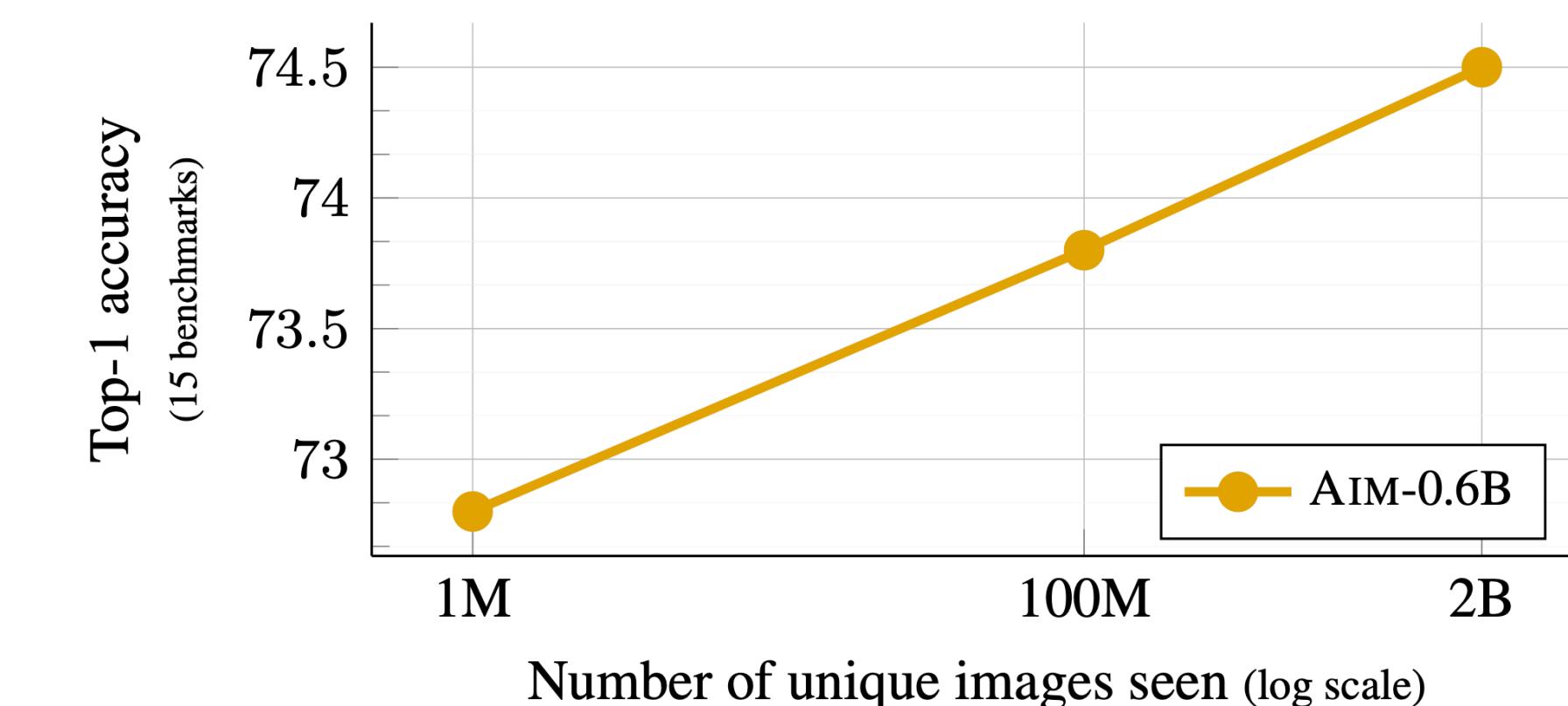
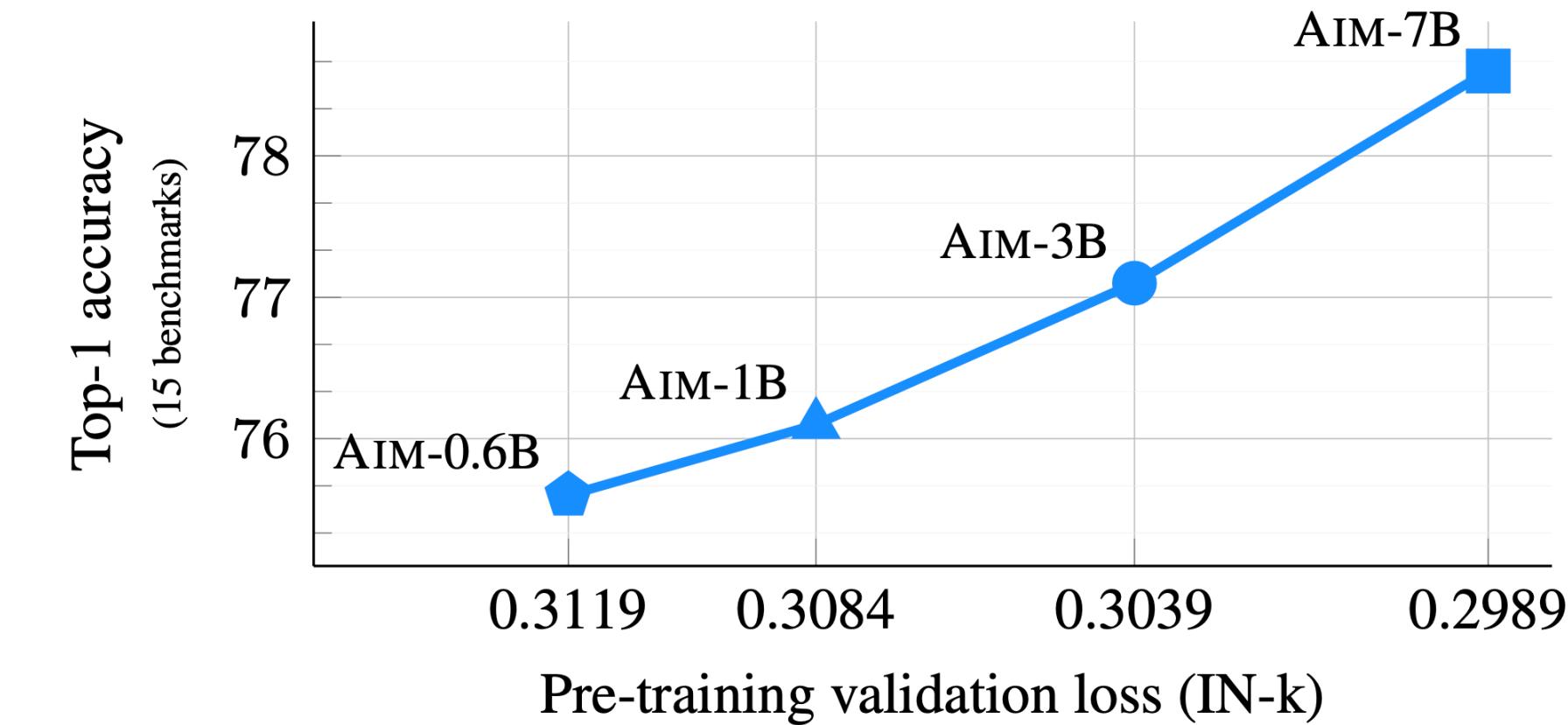
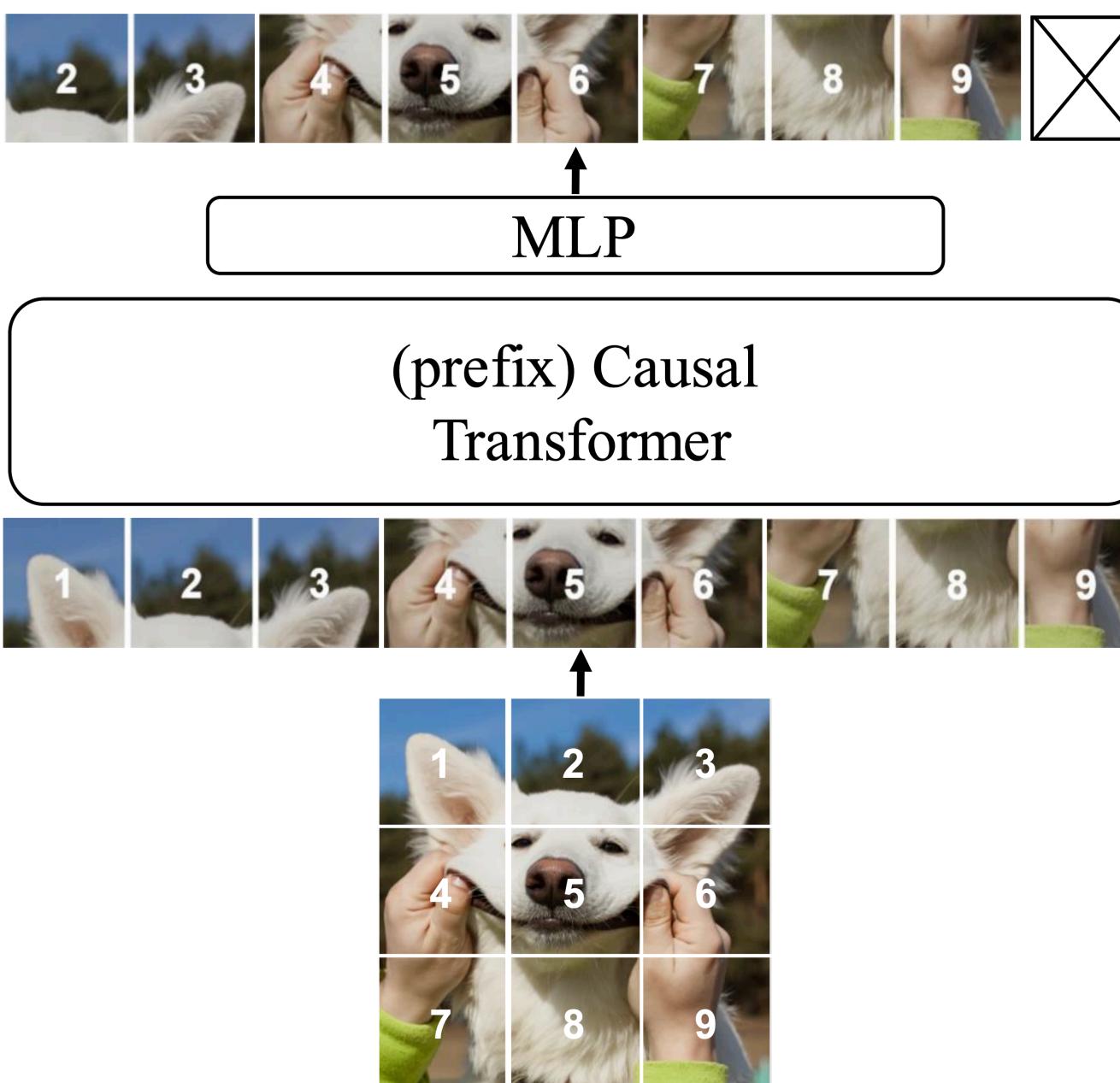
Spatial-Aware Visual Sampler

* replace Ferret visual sampler with a simple prompter

Seeing: From AIM to AIMv2

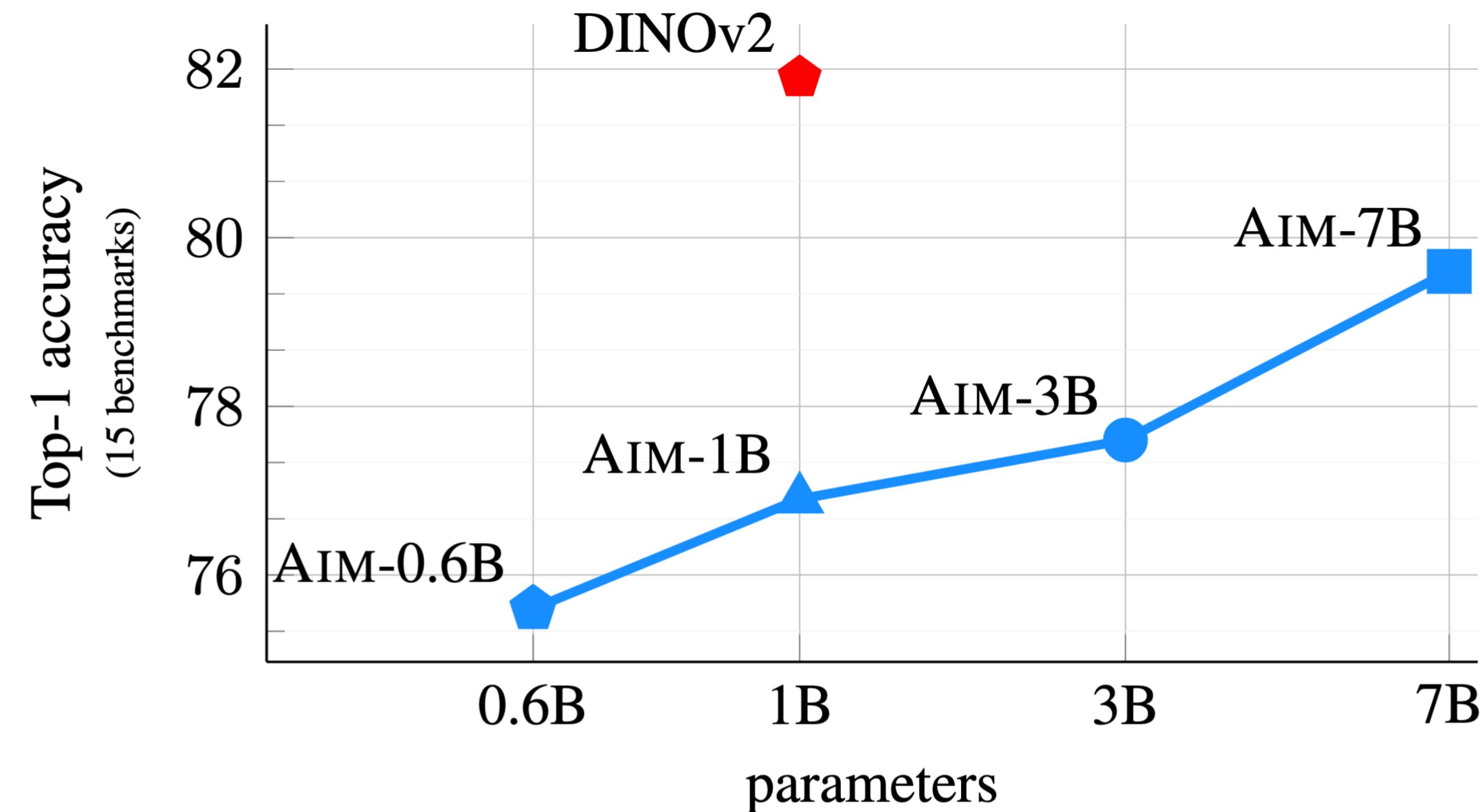
Autoregressive Image Models (AIM)

- Pre-train an image encoder only using autoregressive image pixel losses



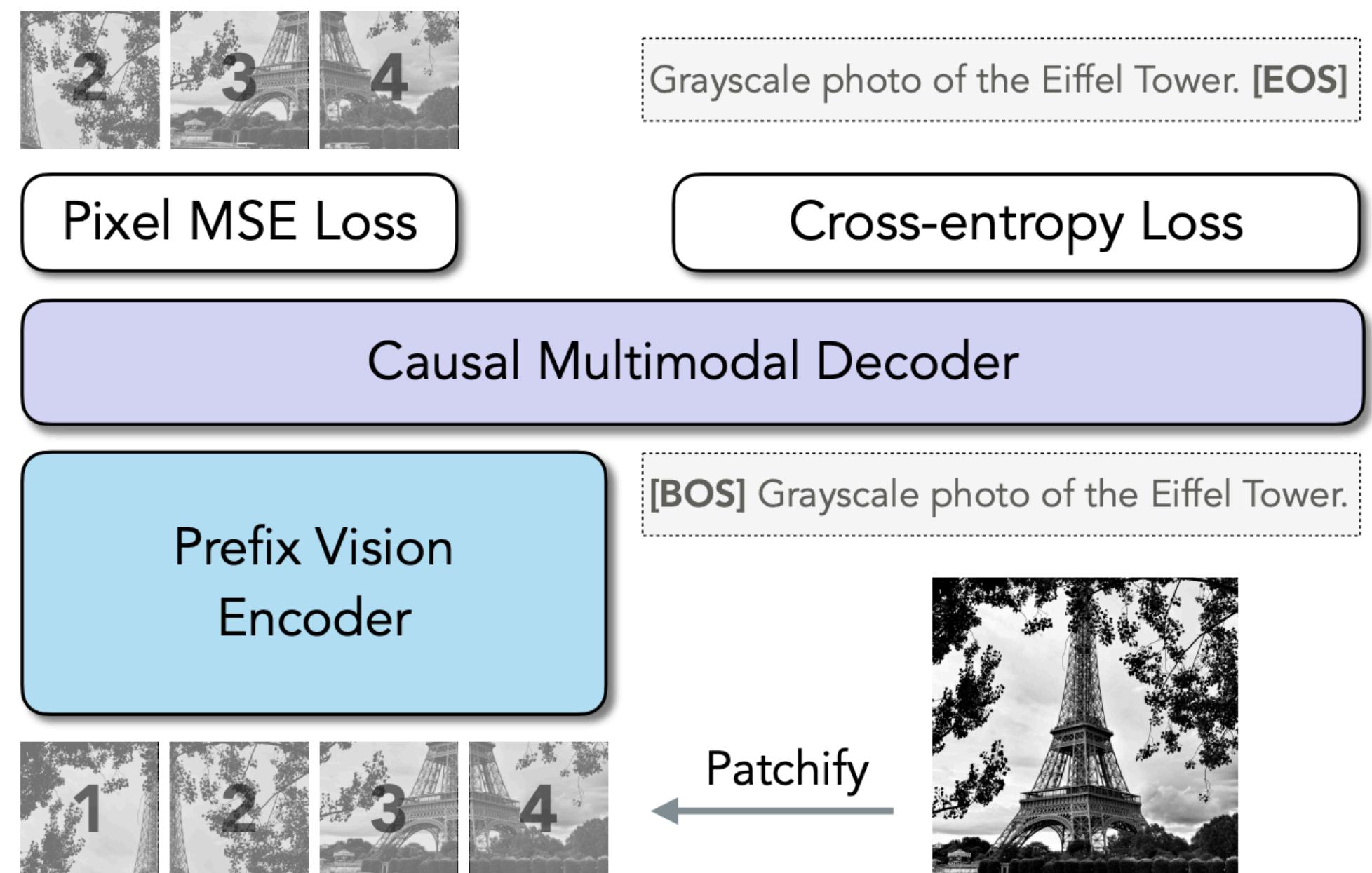
Autoregressive Image Models (AIM)

- Contrastive/Joint embedding (e.g., DINOv2) methods are still more parameter efficient!



Multimodal Autoregressive Pre-training

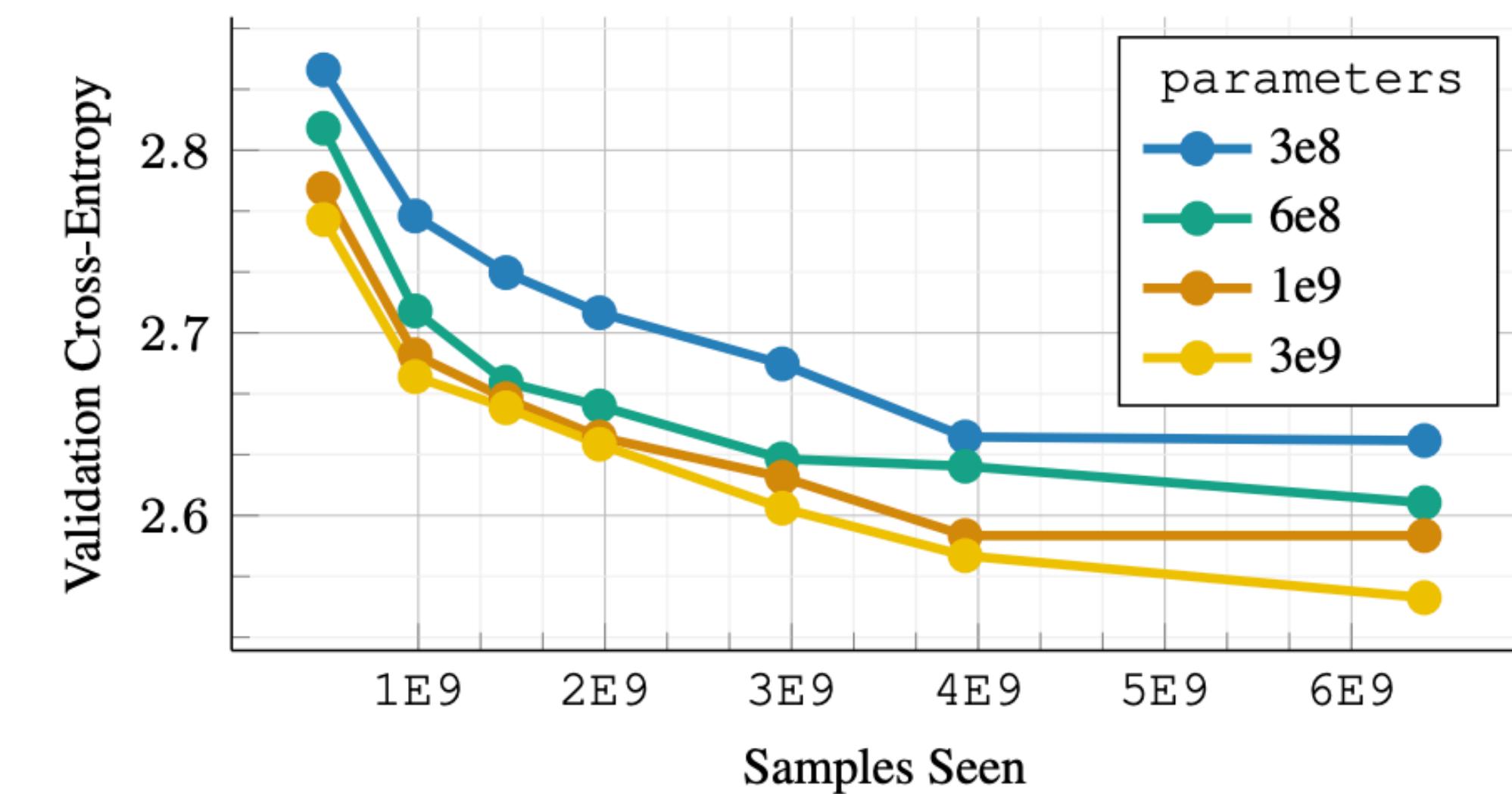
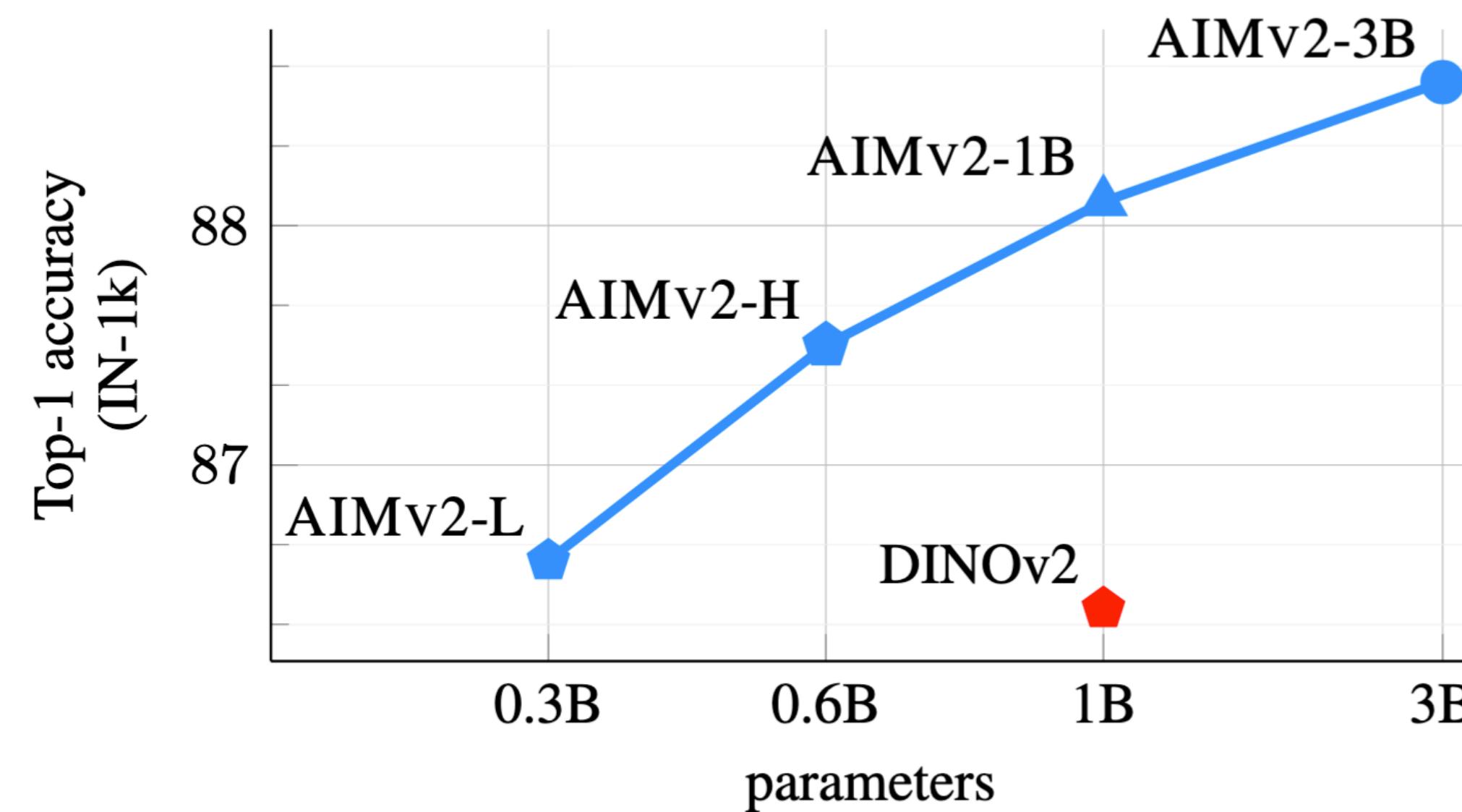
- AIMv2 is a paradigm shift from the predominant CLIP pre-training



- Purely autoregressive objective, easy to scale and parallelize (e.g., no intra-batch sync required)
- Dense supervision with a loss term per token rather than a single global loss
- Better alignment with the multimodal LLM use cases.

Scaling Properties of AIMv2

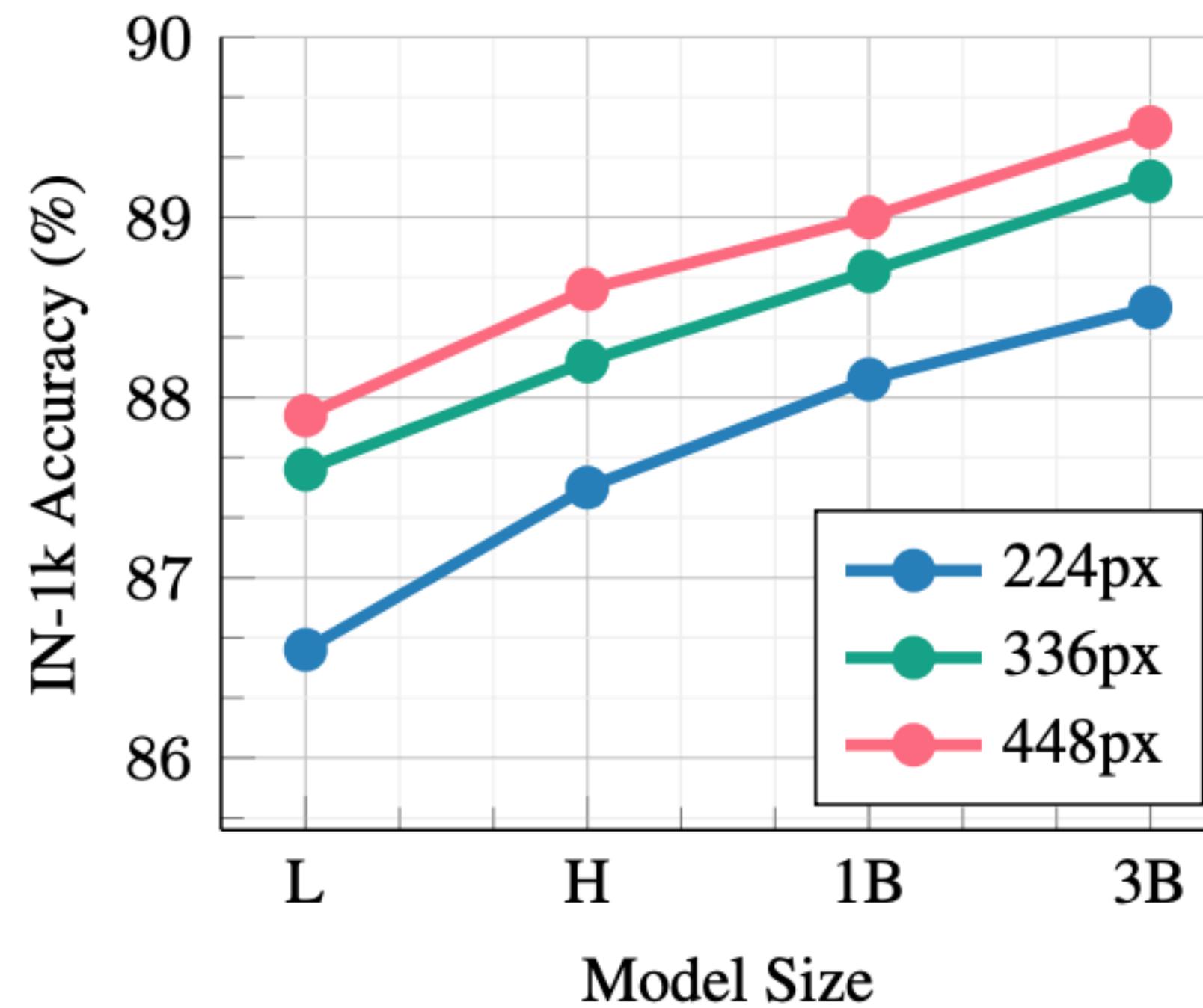
- Scaling in terms of model size and data size



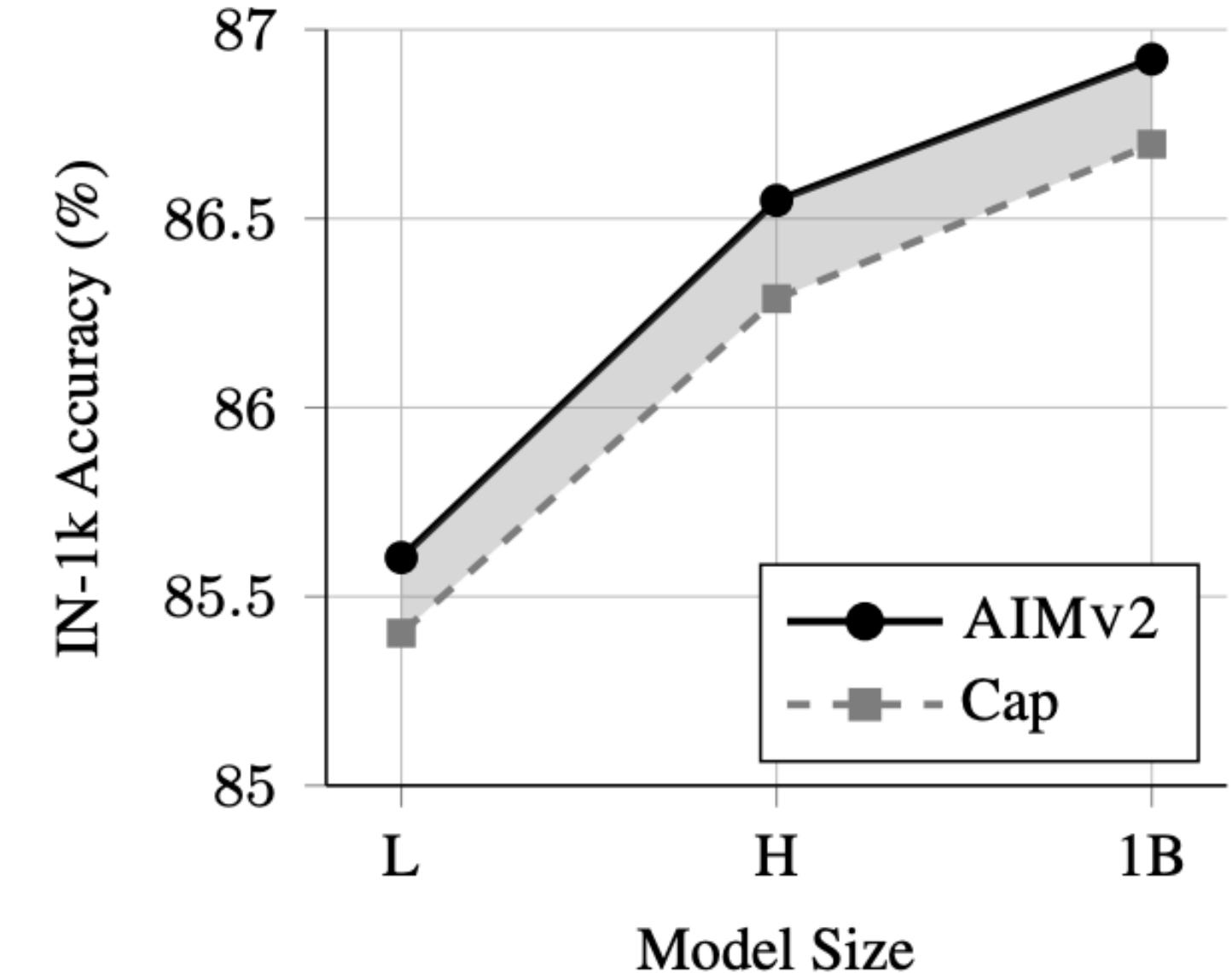
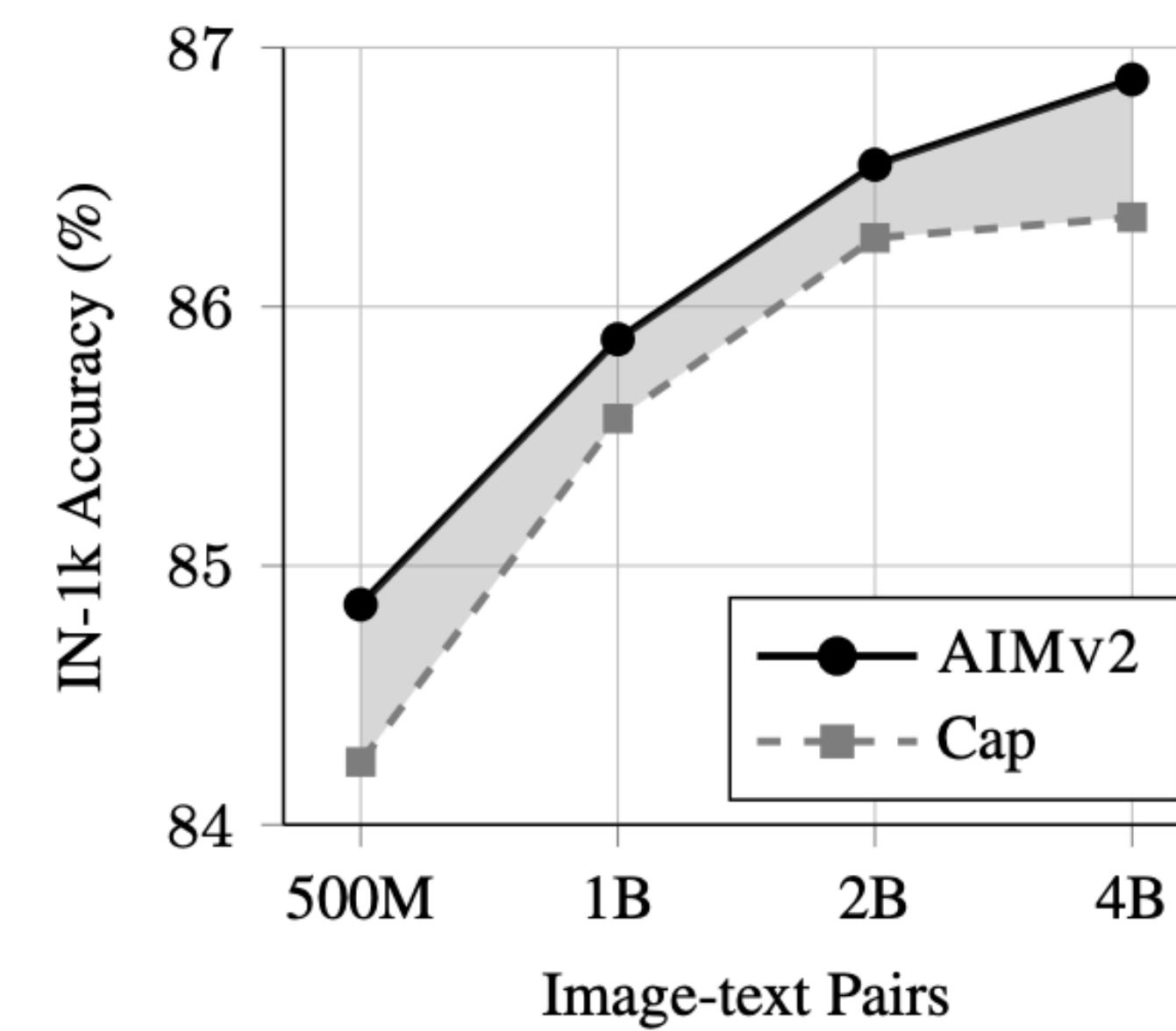
AIMv2 is another example of simple methods that scale well!

Scaling Properties of AIMv2

Scaling in terms of
image resolution



AIMv2 vs Captioning



Other Good Works Out There

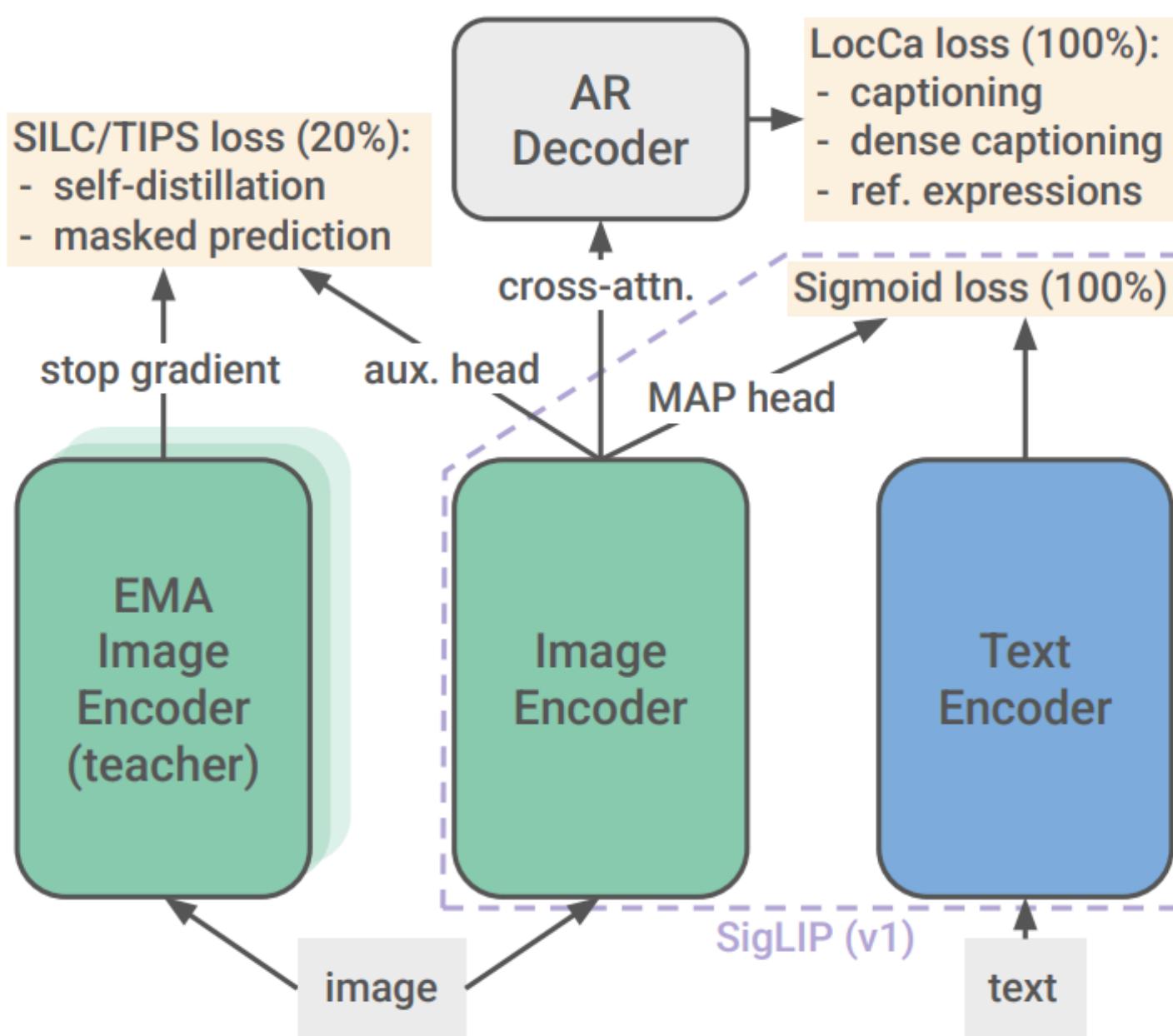


Figure 1 | SigLIP 2 adds the captioning-based pretraining from LocCa [62] as well as self-distillation and masked prediction from SILC [45] and TIPS [38] (during the last 20% of training) to the sigmoid loss from SigLIP [71]. For some variants, the recipe additionally involves fine-tuning with data curation [61] or adaptation to native aspect ratio and variable sequence length [6, 12].

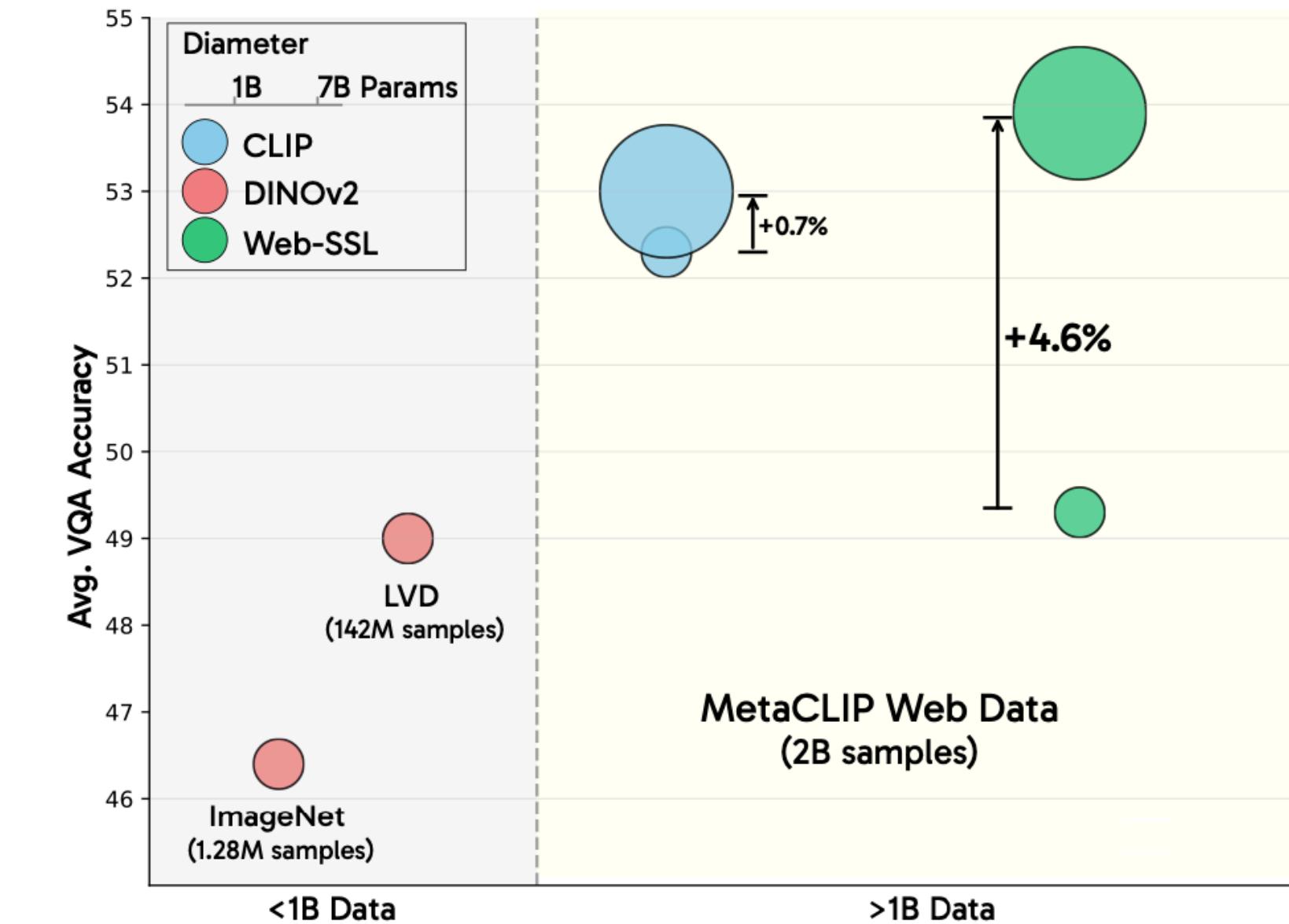


Figure 1 We compare the scaling behavior of visual SSL and CLIP on 16 VQA tasks from the Cambrian-1 suite under different data and model size regimes. Prior visual SSL methods achieved strong performance on classic vision tasks, but have underperformed as encoders for multimodal instruction-tuned VQA tasks. Our results show that with appropriate scaling of models and data, visual SSL can match the performance of language-supervised models across all evaluated domains—even OCR & Chart.

Other Good Works Out There

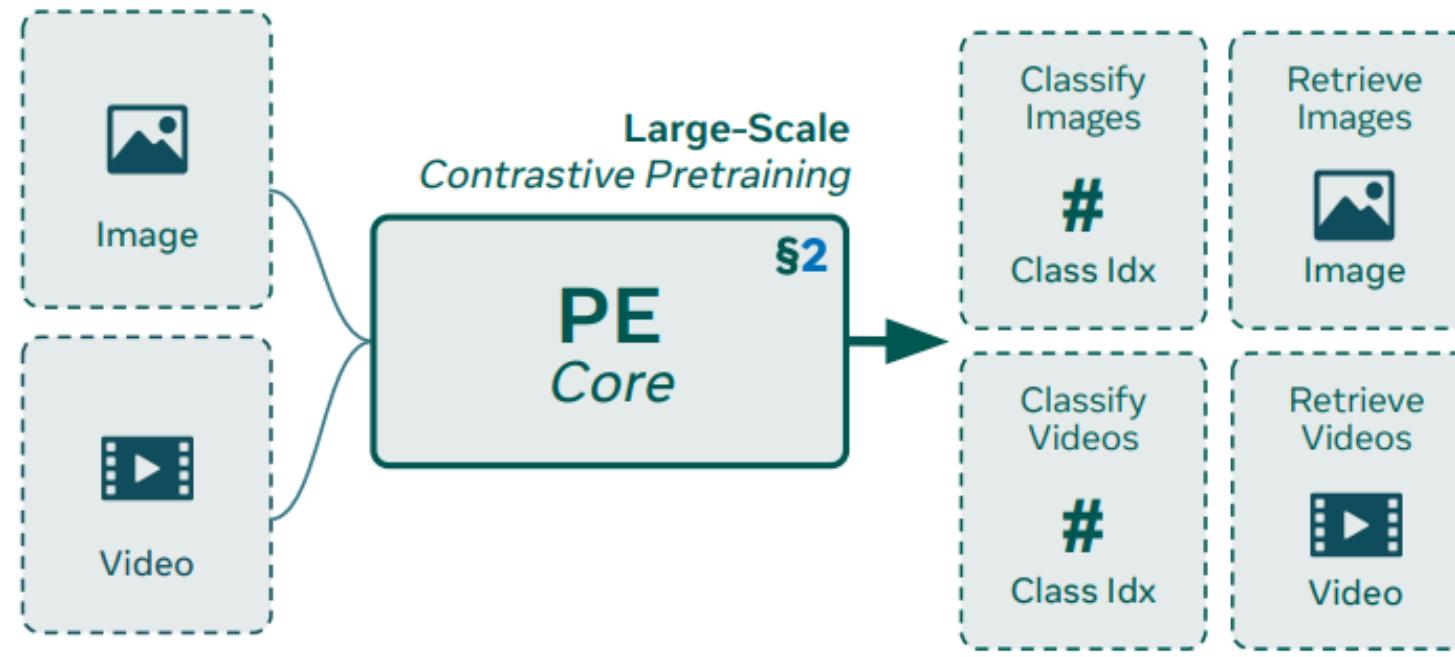


Figure 1 Perception Encoder (PE) is a family of larg



Figure 1: The *top* table compares our OpenVision series to OpenAI's CLIP and Google's SigLIP. The *bottom* figure showcases that OpenVision attain competitive or even superior multimodal performance than OpenAI's CLIP and Google's SigLIP.

Understanding: Multimodal LLM

Core Vision Capabilities (Examples from MM1.5)

UI understanding

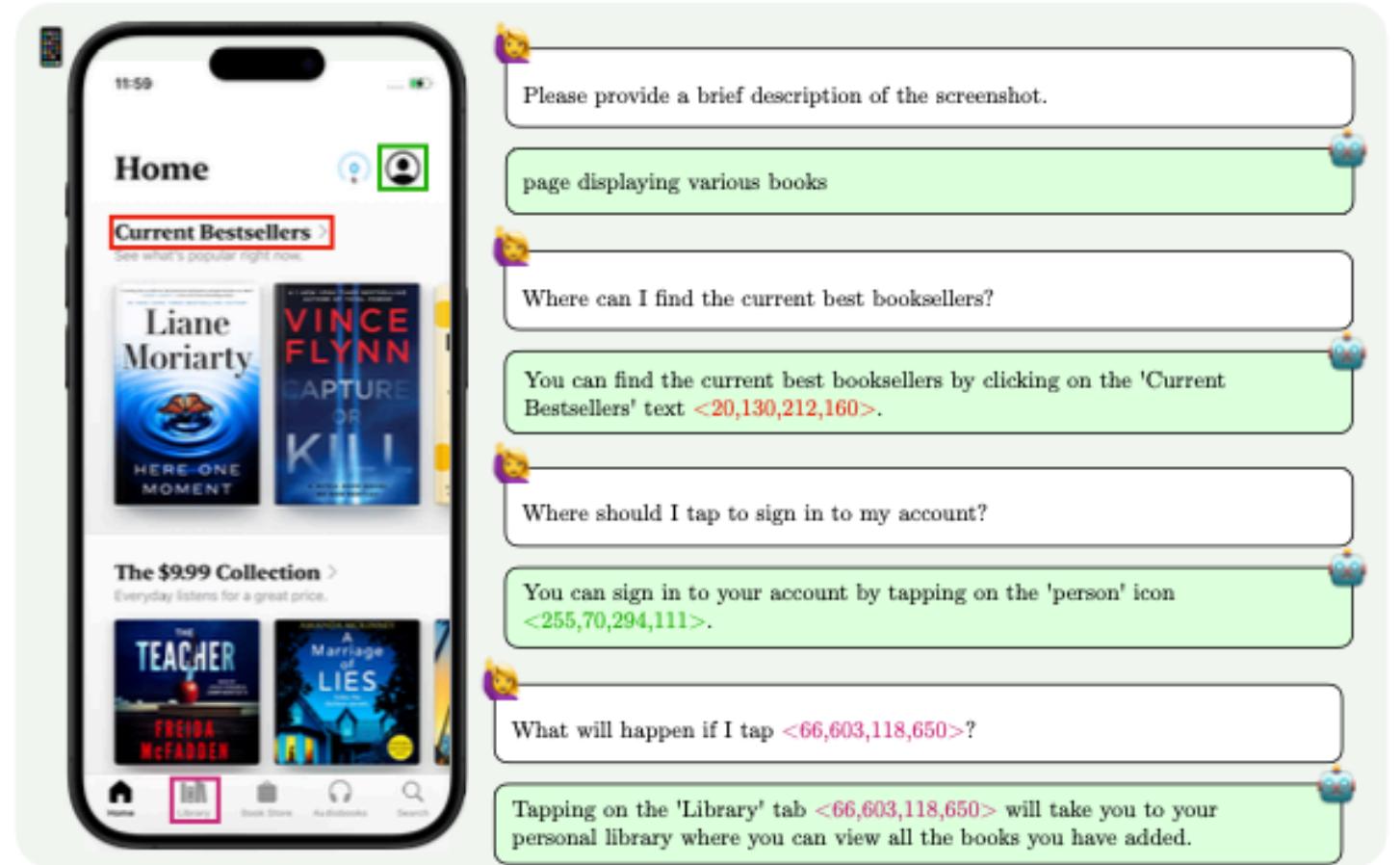
Text-rich image understanding



Text Rich

Read out the abstract of this paper.

In this work, we discuss building performant Multimodal Large Language Models (MLLMs). In particular, we study the importance of various architecture components and data choices. Through careful and comprehensive ablations of the image encoder, the vision language connector, and various pre-training data choices, we identify several crucial design lessons. For example, we demonstrate that for large-scale multimodal pre-training using a careful mix of image-captioning, interleaved image-text, and text-only data is crucial for achieving state-of-the-art (SOTA) few-shot results across multiple benchmarks, compared to other published multimodal pre-training results. Further, we show that the image encoder together with image resolution and the image token count has substantial impact, while the vision-language connector design is of comparable negligible importance. By scaling up the presented recipe, we build MM1, a family of multimodal models, including both dense variants up to 30B and mixture-of-experts (MoE) variants up to 64B, that are SOTA in pre-training metrics and achieve competitive performance after supervised fine-tuning on a range of established multimodal benchmarks. Thanks to large-scale pre-training, MM1 enjoys appealing properties such as enhanced in-context learning, and multi-image reasoning, enabling few-shot chain-of-thought prompting.



Video understanding

Referring & Grounding

What can I make with these ingredients <111,24,315,229> <308,235,460,314> <383,372,571,485>?

The ingredients <111,24,315,229> <308,235,460,314> <383,372,571,485> can be used to make a delicious cake. The first ingredient is cake flour, the second one is eggs, and the third one is butter. These are all essential components for baking a cake.

Should I add this stuff <101,306> to my cake too?

Yes, you should definitely add the canned cherries <54,211,165,388> to your cake. They will add a sweet and tangy flavor to the cake, enhancing its taste.

If we want to make a cake, choose the best ingredients in the image.

To make a cake, you would need flour, sugar, eggs, butter, and baking soda from the image.

Can you show me their coordinates?

Sure! The flour <112,26,316,239>, sugar <306,49,484,251>, eggs <312,239,392,312> <386,243,461,316>, butter <383,371,571,481>, and baking soda <457,155,567,308> are the ingredients you would need for a cake.



Here are two videos. Please provide the analysis on their differences.

The first video and the second video are similar in that they both depict a person gardening. However, there are some differences between the two videos. In the first video, the person is seen picking berries from a bush, while in the second video, the person is seen planting flowers in a garden. This difference in activity suggests that the two videos may have been taken at different times or in different locations. Additionally, the first video is taken from a bird's-eye view, while the second video is taken from a lower angle. This difference in perspective gives the two videos a different feel and may affect the viewer's experience. Overall, while the two videos share some similarities, they also have some differences in terms of the activity being performed and the perspective from which the video is taken.

Core Vision Capabilities (Examples from MM1.5)

Multi-Image



Help me generate a Twitter post providing these images.

Step 1: Identify the main subject of each image.

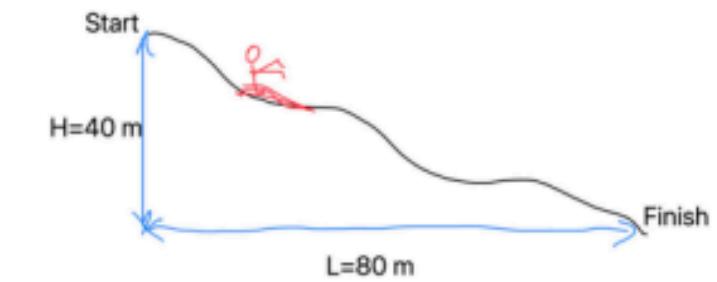
- Image 1: The main subject is a crowd of people at a concert.
 - Image 2: The main subject is the Sagrada Familia, a large Gothic-style church in Barcelona.
 - Image 3: The main subject is a cup of coffee with a heart-shaped foam design.
- Step 2: Determine the tone and theme of each image.
- Image 1: The tone is energetic and lively, with a focus on the vibrant atmosphere of a concert.
 - Image 2: The tone is historical and architectural, showcasing the grandeur of the Sagrada Familia.
 - Image 3: The tone is cozy and inviting, with a focus on the comforting act of enjoying a cup of coffee.

Step 3: Craft a Twitter post that incorporates the main subject and tone of each image.

"Experience the energy of a live concert at [Concert Venue] or take a moment to appreciate the architectural beauty of [Sagrada Familia] with [Tourist Attraction]. Enjoy a warm cup of coffee with a heart-shaped foam design from [Coffee Shop]."

Math/Reasoning

23. A skier slides down the frictionless slope as shown. What is the skier's speed at the bottom?



Solution:

Try to reason about the question step by step.



To find the skier's speed at the bottom, we can use the conservation of energy principle. At the start, the skier has potential energy, and at the end, the skier has kinetic energy. Since the energy is conserved, we can set up the following equation:

$$\text{Start: } E = mgh \text{ End: } E = (1/2)mv^2$$

where E is the energy, m is the mass of the skier, g is the acceleration due to gravity (9.81 m/s^2), h is the height, and v is the final velocity.

We are given the initial height (40 m) and the final height (0 m) since the skier reaches the ground. We can plug these values into the equation and solve for the final velocity (v):

$$(1/2)mv^2 = mgh$$

Divide both sides by m:

$$(1/2)v^2 = gh$$

Now, plug in the given values for g and h:

$$(1/2)v^2 = (9.81 \text{ m/s}^2)(40 \text{ m})$$

$$(1/2)v^2 = 392.4 \text{ J}$$

Multiply both sides by 2:

$$v^2 = 784.8 \text{ J}$$

Now, take the square root of both sides to find the final velocity (v):

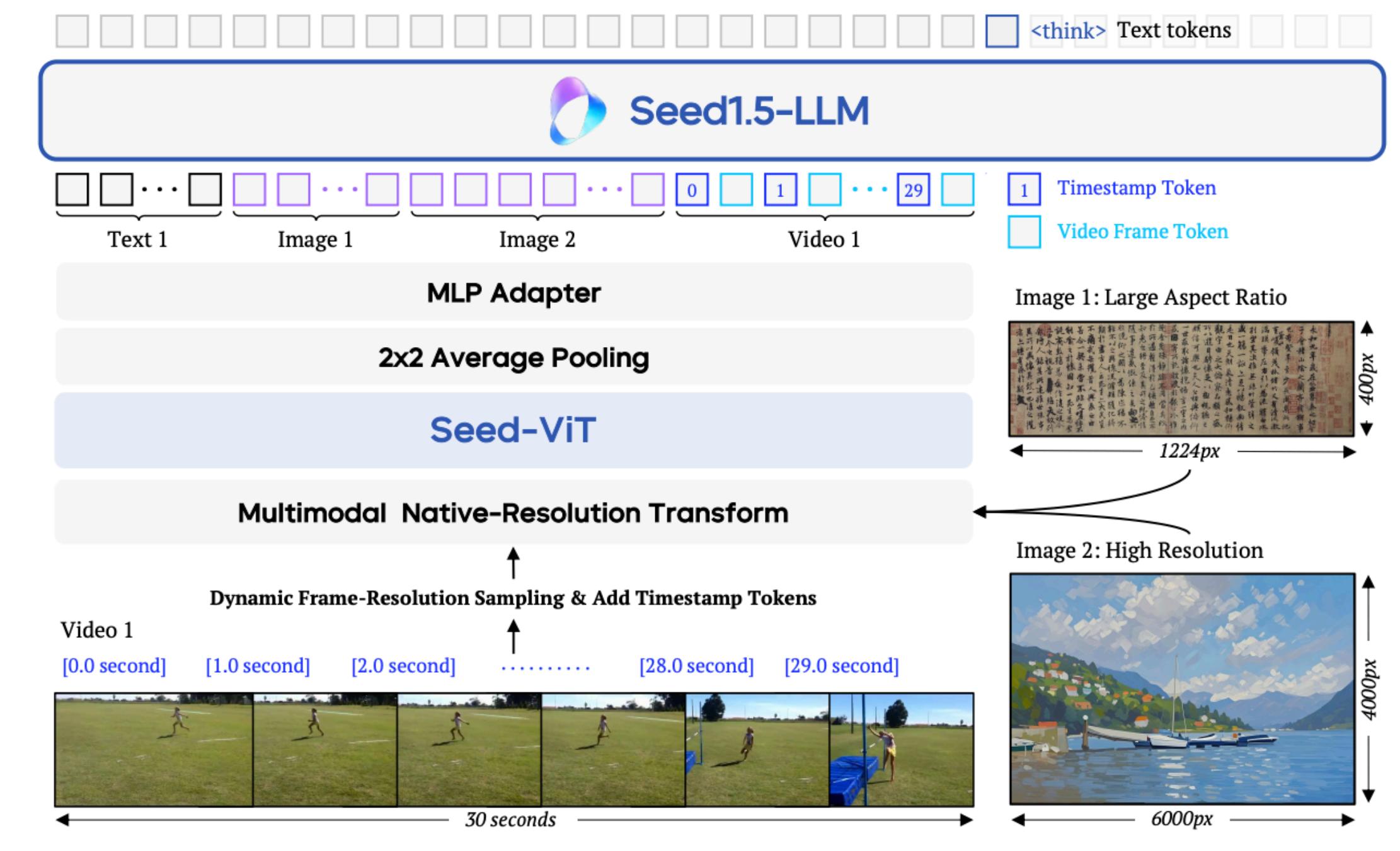
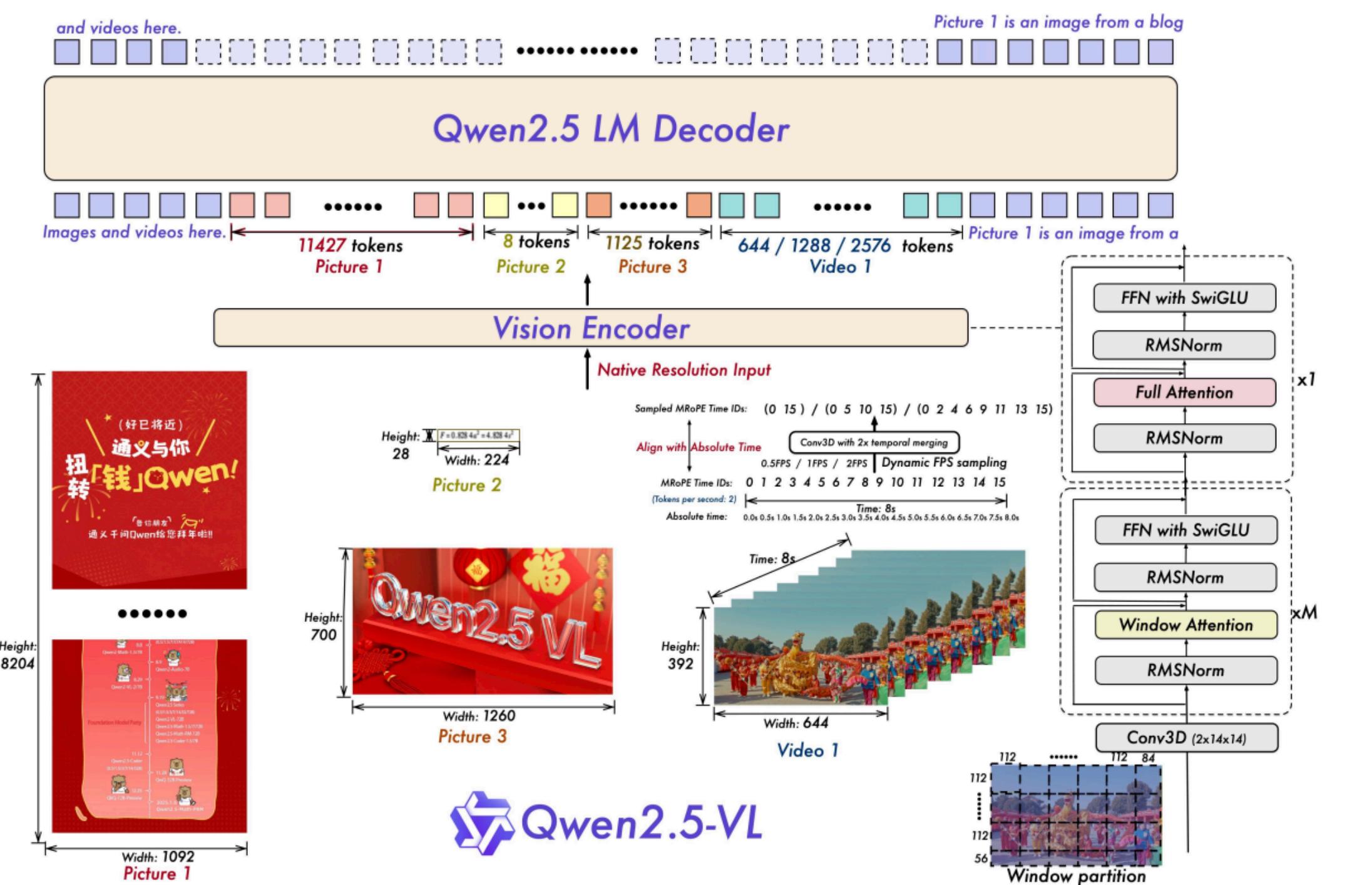
$$v = \sqrt{784.8} \text{ J}$$

$$v \approx 28 \text{ m/s}$$

So, the skier's speed at the bottom is approximately 28 m/s.

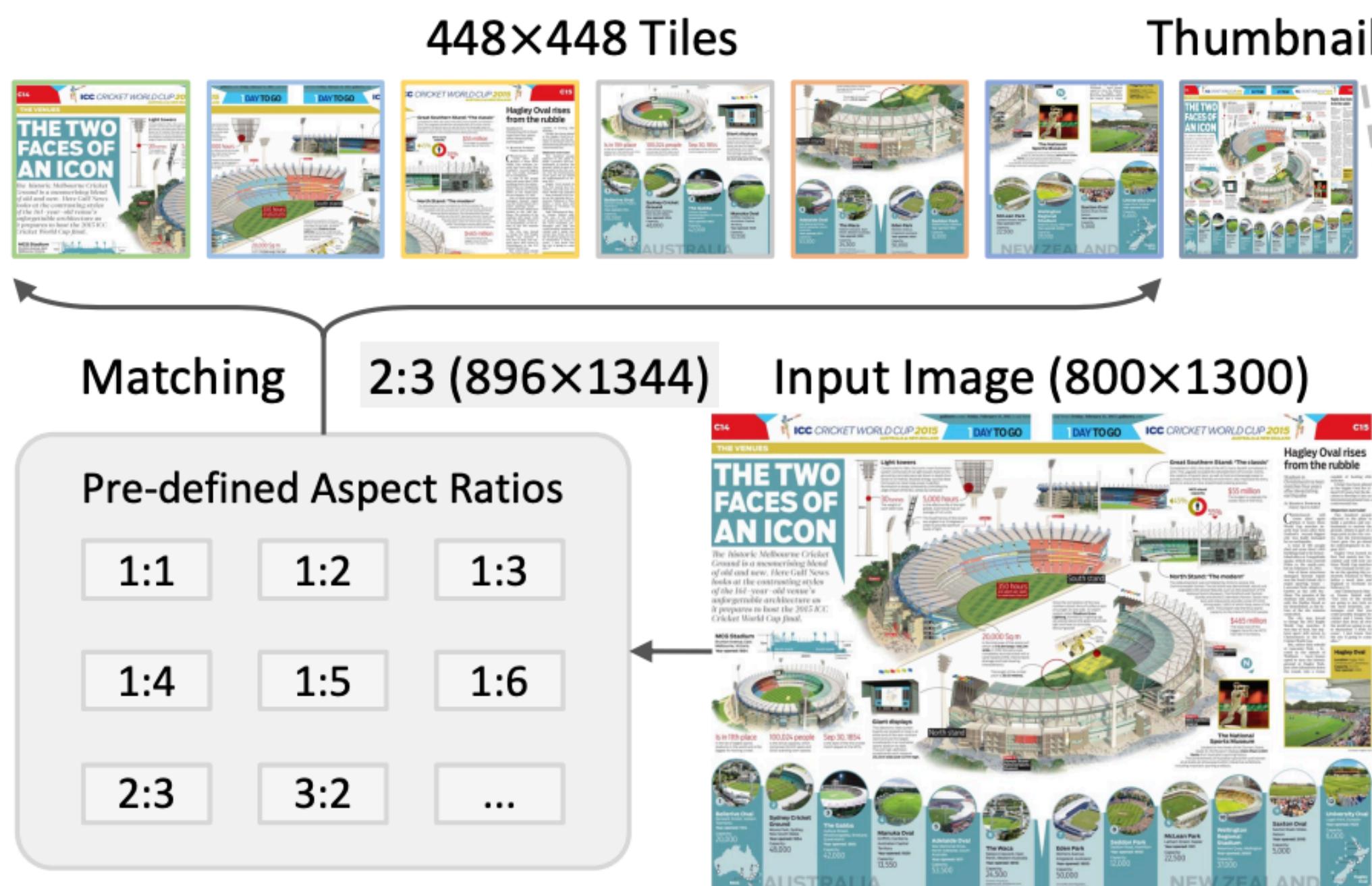
Qwen2.5-VL vs. Seed1.5-VL

- Modern multimodal LLMs look increasingly similar
- It's all about data, no matter it's text-rich, UI, or video understanding

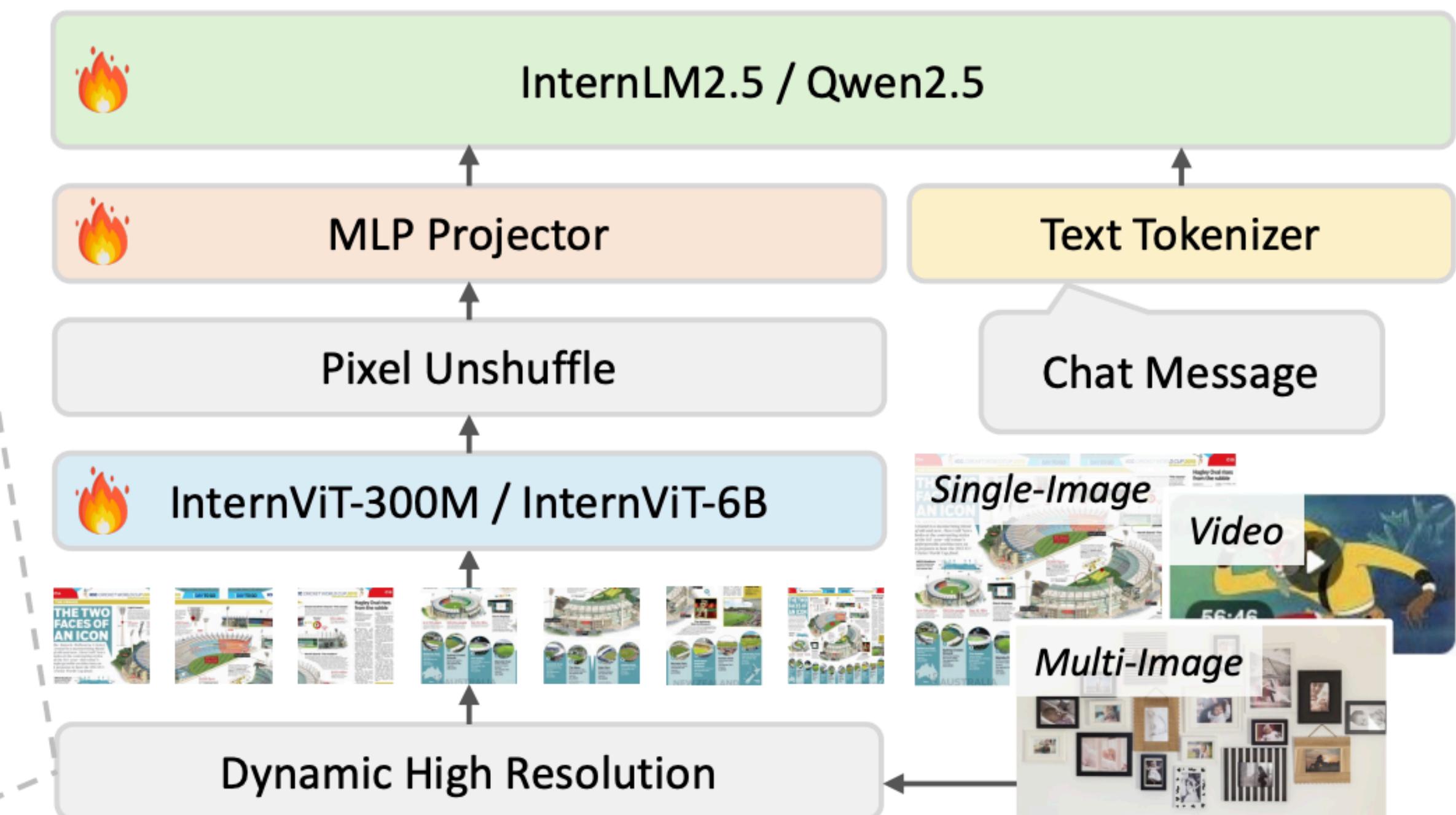


Example 1: Text-rich Image Understanding (InternVL-2.5)

- Native resolution support as in Qwen2.5-VL using NaViT like methods
- High-resolution support via dynamic image splitting as in InternVL-2.5 and many others



(a) Data Preprocessing



(b) Model Architecture

Example 2: UI Understanding (Ferret UI 2)

- GUI grounding and navigation becomes an increasingly hot topic
- Many good works out there, e.g., U-Ground, Show-UI, UI-TARS and all recent R1-like methods

The figure displays four screenshots of Apple devices illustrating UI elements:

- iPhone Screenshot:** Shows the "Display & Brightness" settings screen. It features three green boxes labeled [Box0] through [Box3].
- iPad Screenshot:** Shows the weather forecast for Cupertino. It features three yellow boxes labeled [Box0] through [Box3].
- Mac Screenshot:** Shows the "Accessories" store page for AirTags. It features four red boxes labeled [Box0] through [Box4].
- Apple TV Screenshot:** Shows the home screen of the Apple TV app. It features three cyan boxes labeled [Box0] through [Box2].

User Questions and Answers:

Device / Screen	Question	Answer
iPhone Screenshot	What kind of UI widget is in [Box0]?	This UI widget is a toggle.
	What is the current state of the appearance toggle and how to switch it?	The current theme is set to Light Mode [Box1]. To switch to Dark Mode, you would tap on the radio button [Box2] next to "Dark", which is currently unchecked.
	How would you interact with the brightness slider to decrease the screen brightness?	The current brightness level is slightly above mid-level. To decrease the brightness, drag the slider [Box3] towards the left, where the lower brightness icon is located.
	What does the text in [Box0] say?	The text says "Complete your return online or take it to an Apple Store".
	What is the [Box0] icon?	This icon is for "App Store" on Apple's devices.
iPad Screenshot	What does the text in [Box0] indicates?	The text that reads "10-DAY FORECAST" indicates the list below shows the weather forecast of the next 10 days.
	What is the current temperature in Cupertino, and what is the weather condition?	The current temperature in Cupertino is 91°F [Box1], and the weather condition is "Mostly Sunny" [Box2].
	How to view the weather conditions of later hours?	You can scroll horizontally on the hourly forecast row [Box3] to see weather conditions for later hours.
	What does the large "For All Mankind" banner suggest about the context of this screen?	The large banner suggests that "For All Mankind" [Box1] is either a featured show or content available on the Apple TV app. The banner acts as promotional content, encouraging users to watch or explore the show.
	Where do I open the Apple TV app?	To open the Apple TV app, tap or select the black tile with the Apple TV logo [Box2].
Mac Screenshot	How to explore different color options for the AirTag FineWoven Key Ring?	Tap or click on the dots [Box1] below the image of the AirTag FineWoven Key Ring.
	What are the prices of the three visible products on the screen?	The prices are AirTag for \$29.00 [Box2] AirTag 4-pack for \$99.00 [Box3] and AirTag FineWoven Key Ring – Blackberry for \$35.00 [Box4].
	What is the [Box0] icon?	This icon is for "App Store" on Apple's devices.
	What does the text in [Box0] say?	The text says "Complete your return online or take it to an Apple Store".
	What is the [Box0] icon?	This icon is for "App Store" on Apple's devices.
Apple TV Screenshot	What does the text in [Box0] say?	The text says "Complete your return online or take it to an Apple Store".
	What is the [Box0] icon?	This icon is for "App Store" on Apple's devices.

Example 3: How About Egocentric Intelligence?

“What can I see?” – Long Video Understanding

“Where did I leave my phone?” – Long Video Memory

Interaction

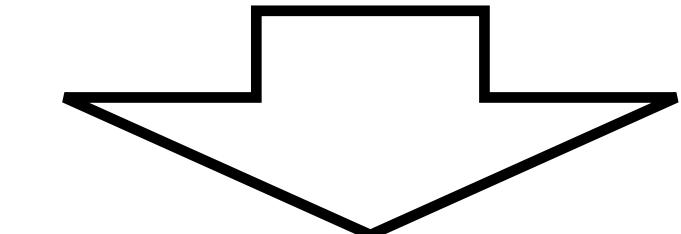
MM-Ego: Data

Human Annotated Narrations
from existing dataset (Ego4D)

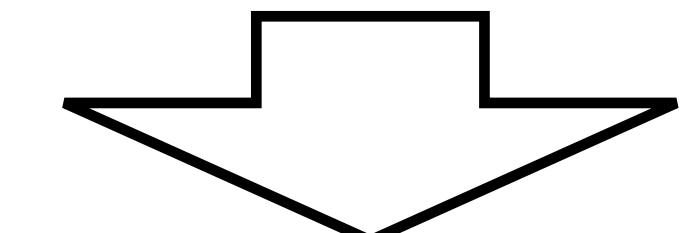
Video Clip 1: “I sit down on the sofa.”

Video Clip 2: “I put the wallet and phone on the table.”

Text-only LLM



Prompt: “Design a question about visual details based on the narrations.”



Memory QA

Video: [Video Clip 1, Video Clip 2, Video Clip 3]

Question: “Given this video, where did I leave my phone?”

Answer: “I left my phone on the table.”

Long Video Memory Dataset

Conversation counts

- Train split: 942 K
- Test split: 32 K

Question counts

- Train split: 7 M
- Test split: 235 K

Long Video Understanding Dataset

- Conversation/Question counts
- Train split: 999k



Q: Which hand did the man place on his chest?

A: The man placed both hands on his chest.

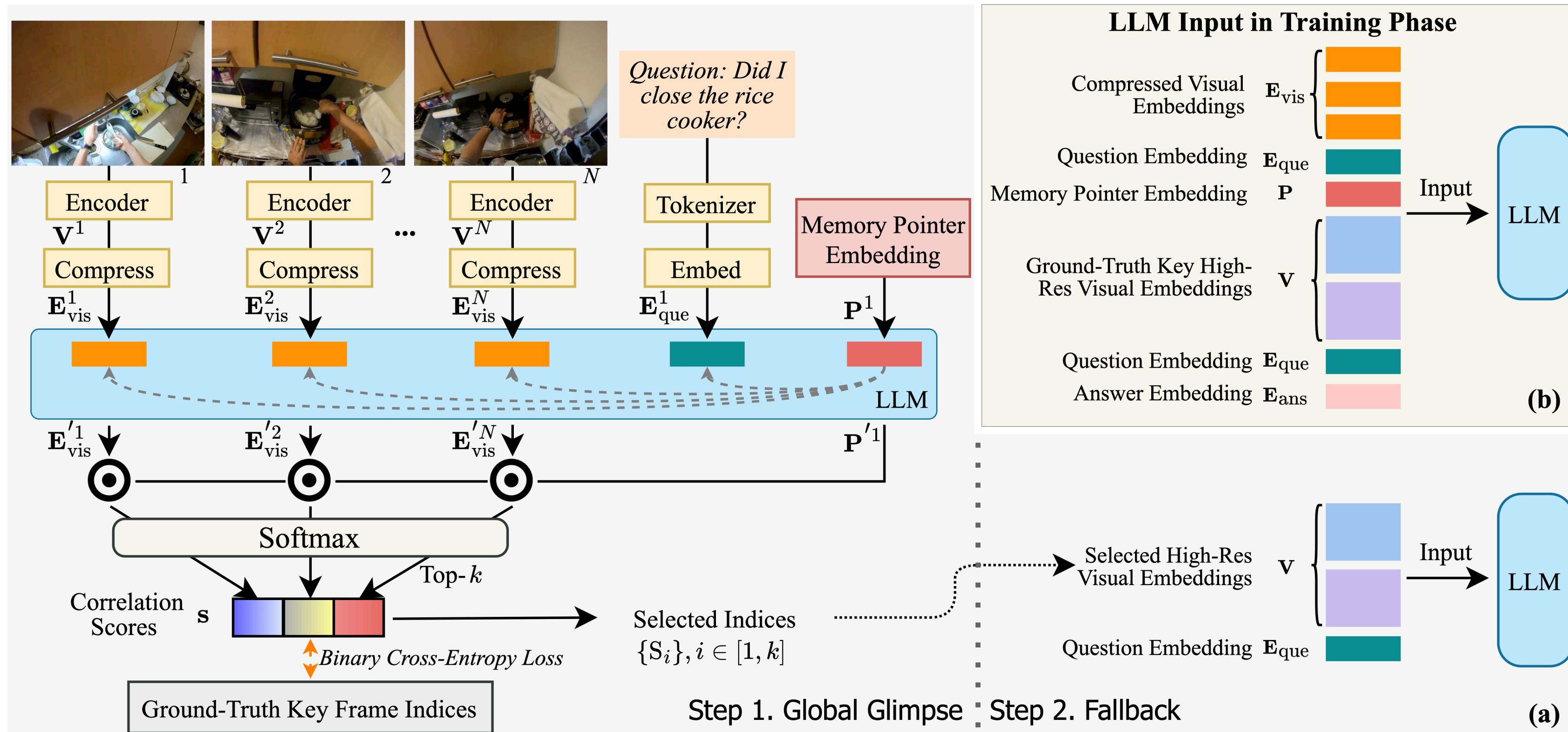


Q: What did I do with the pear after slicing it?

A: I moved the pear on the tray with the knife in my right hand.

MM-Ego: Model Architecture

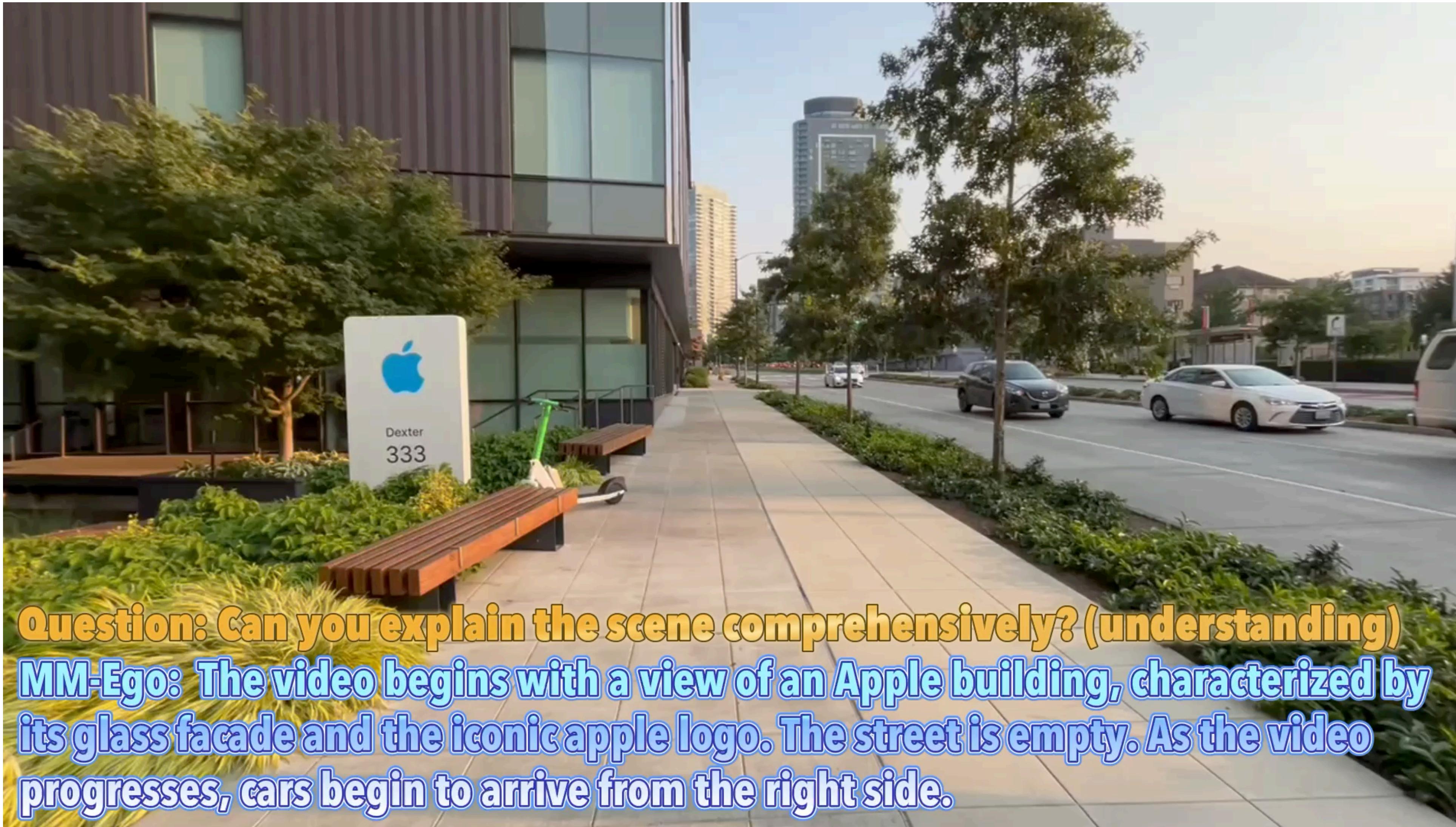
- Memory Pointer Prompting



Global Glimpse - the correlation scores between the memory pointer and all compressed visual embeddings

Fallback - high-resolution visual embeddings corresponding to the selected indices

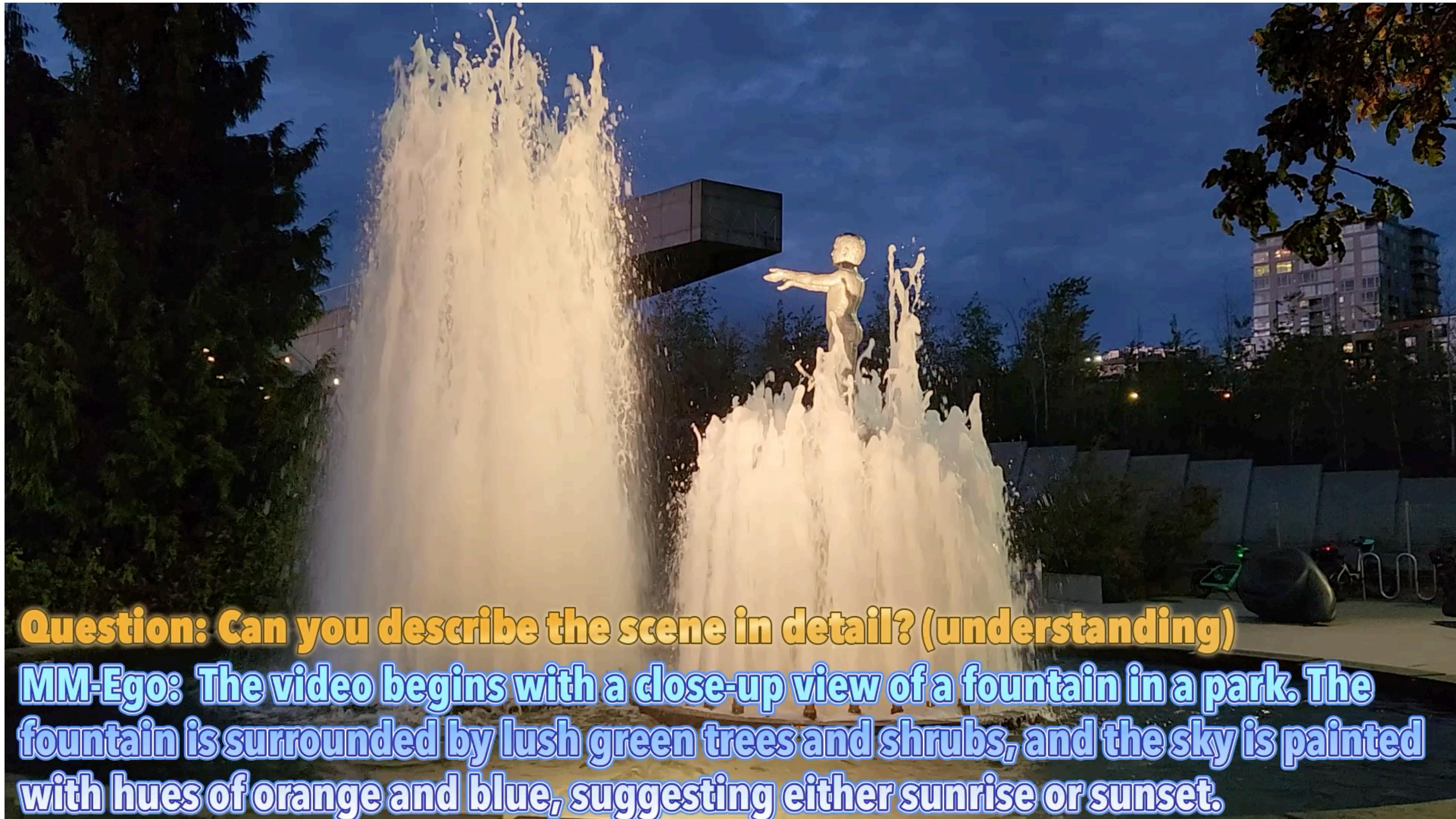
MM-Ego: Apple Office at Seattle



Question: Can you explain the scene comprehensively? (understanding)

MM-Ego: The video begins with a view of an Apple building, characterized by its glass facade and the iconic apple logo. The street is empty. As the video progresses, cars begin to arrive from the right side.

MM-Ego: Another View from Seattle

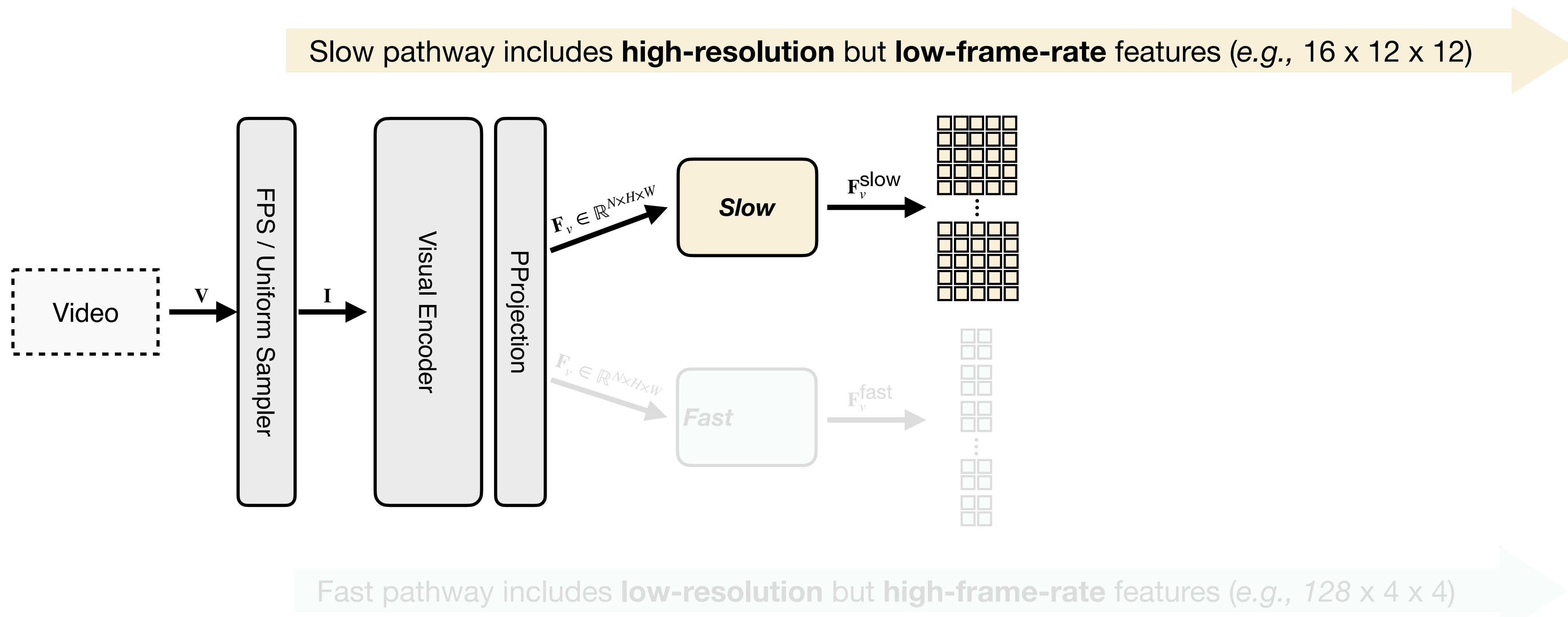


Question: Can you describe the scene in detail? (understanding)

MM-Ego: The video begins with a close-up view of a fountain in a park. The fountain is surrounded by lush green trees and shrubs, and the sky is painted with hues of orange and blue, suggesting either sunrise or sunset.

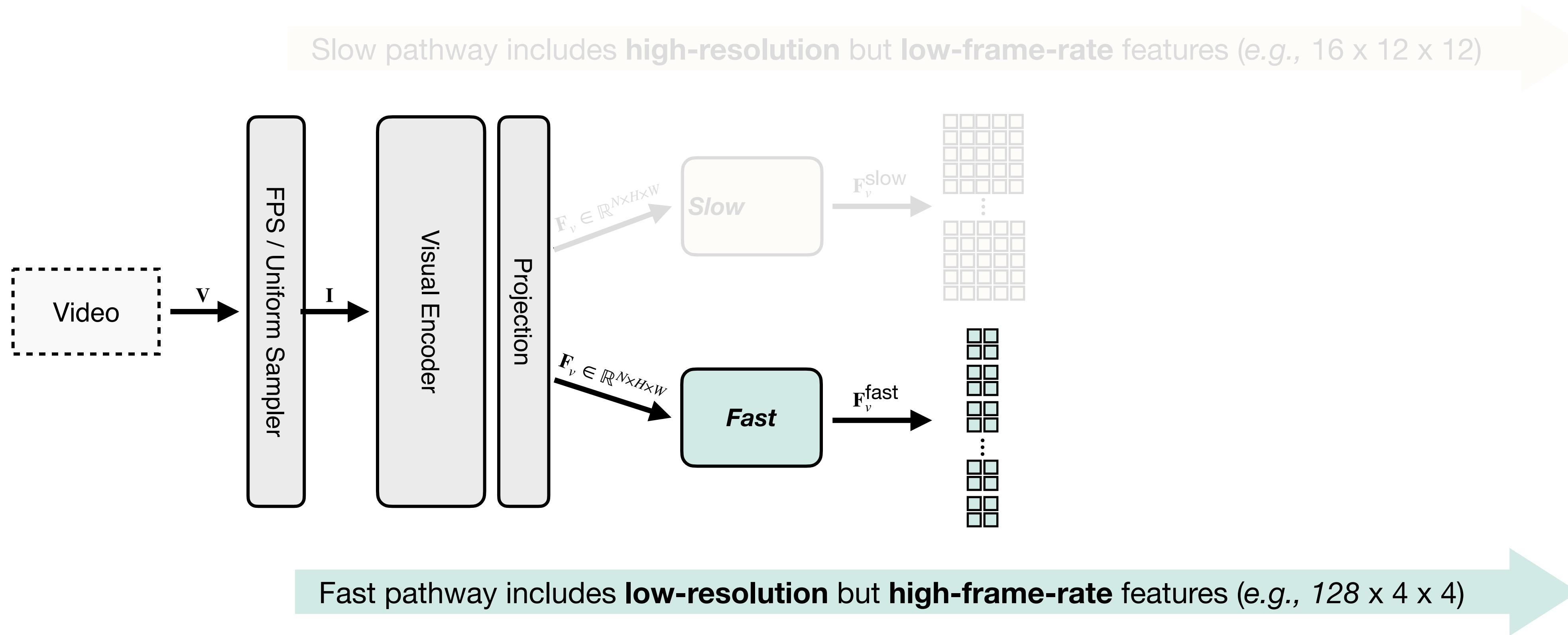
Slow Fast Thinking for Video Understanding

- Slow pathway deals with high-resolution features



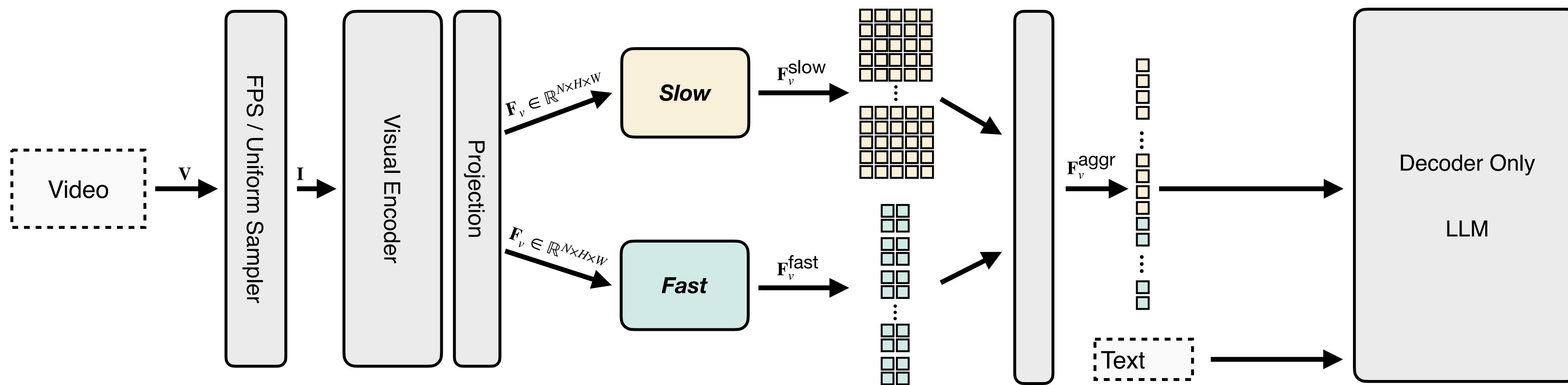
Slow Fast Thinking for Video Understanding

- Fast pathway deals with high-frame-rate features



Slow Fast Thinking for Video Understanding

- SlowFast-LLaVA-1.5 achieves strong performance across all image and video benchmarks when trained on the joint SFT mixture



Video Results: 3B Models

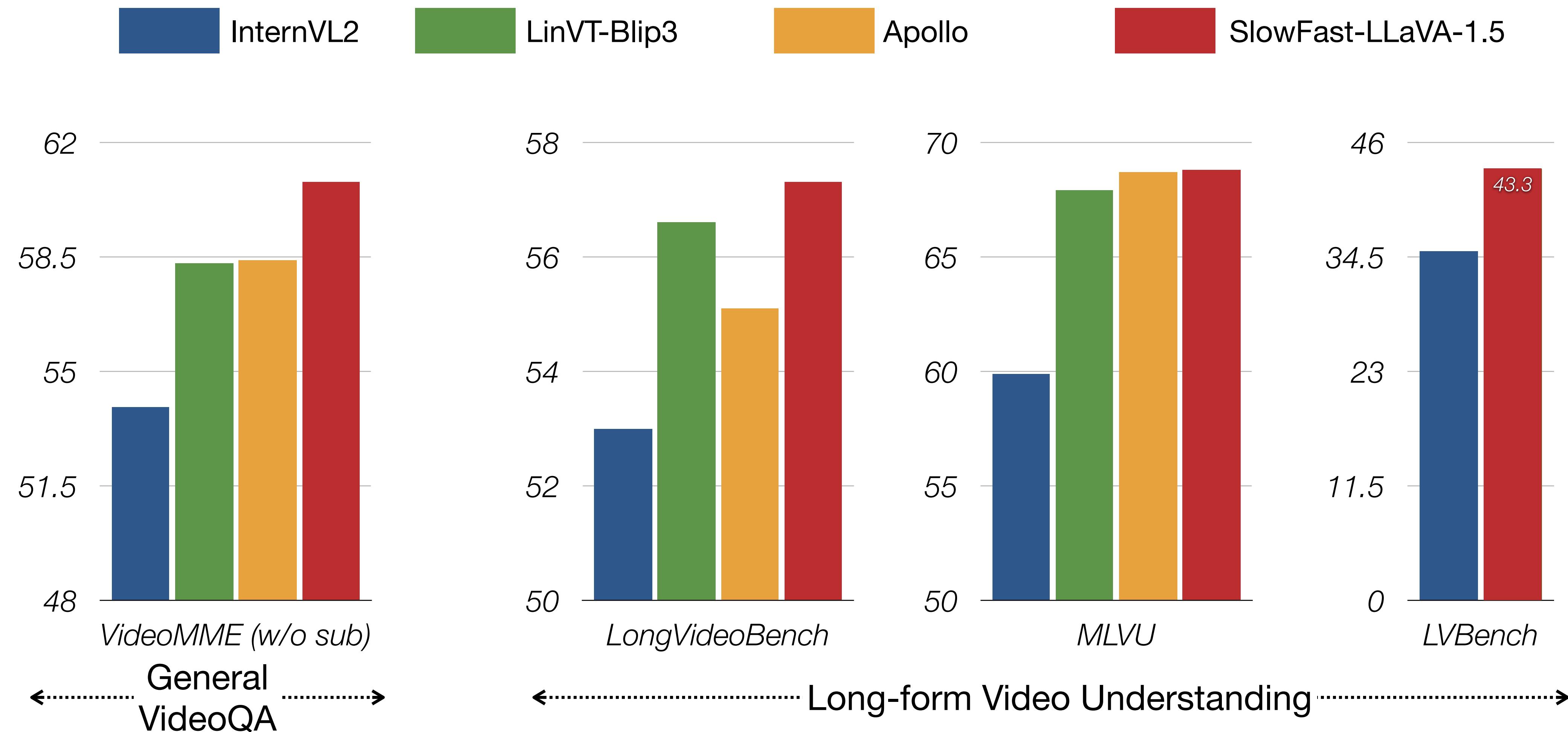
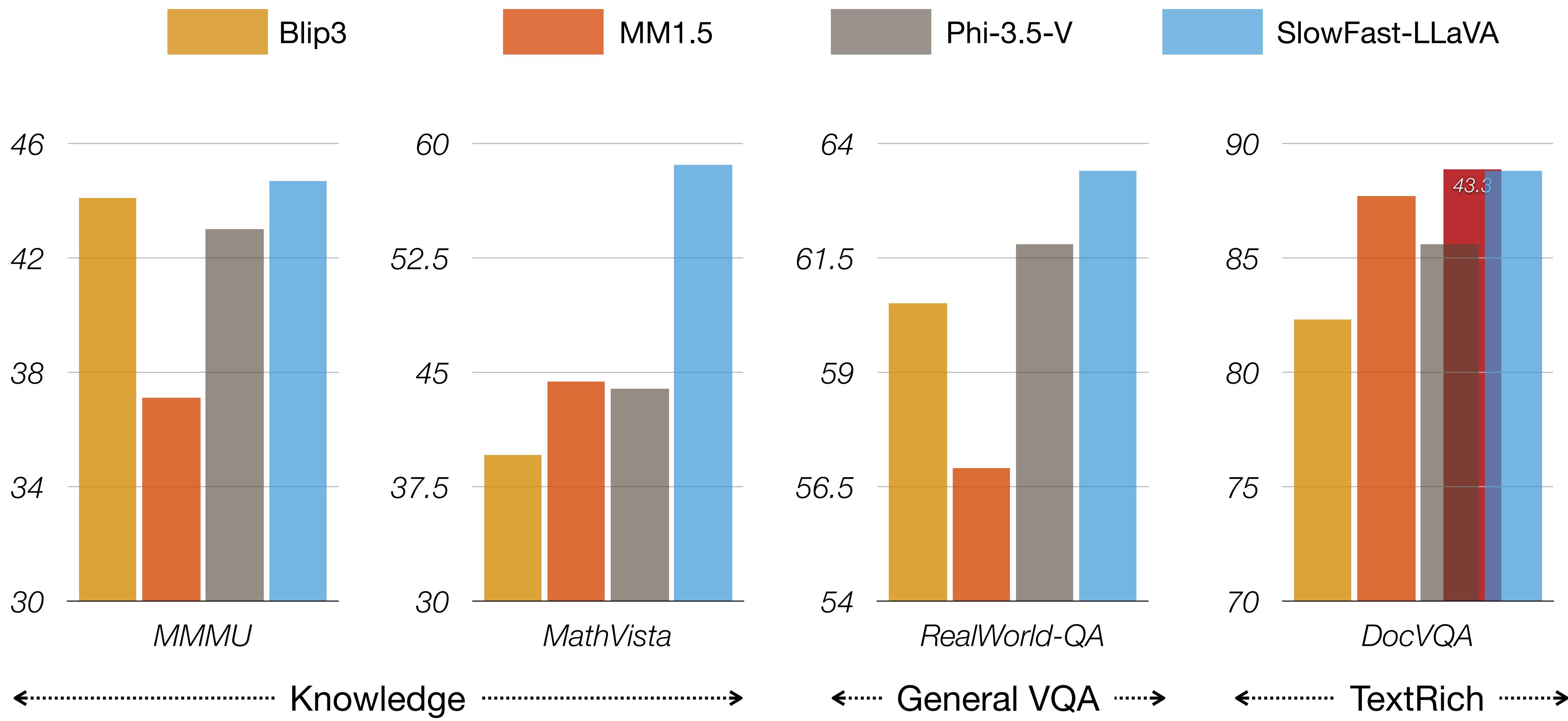
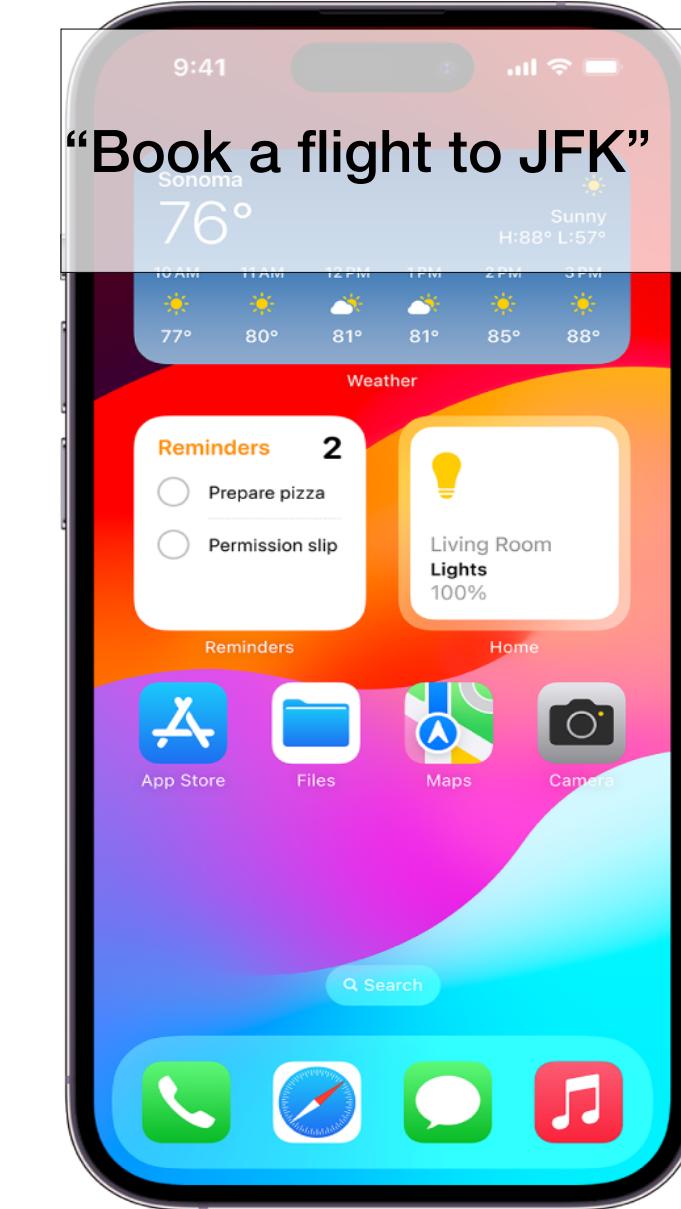


Image Results: 3B Models



Acting: Multimodal Agent

Slides in this section are from Andrew Szot



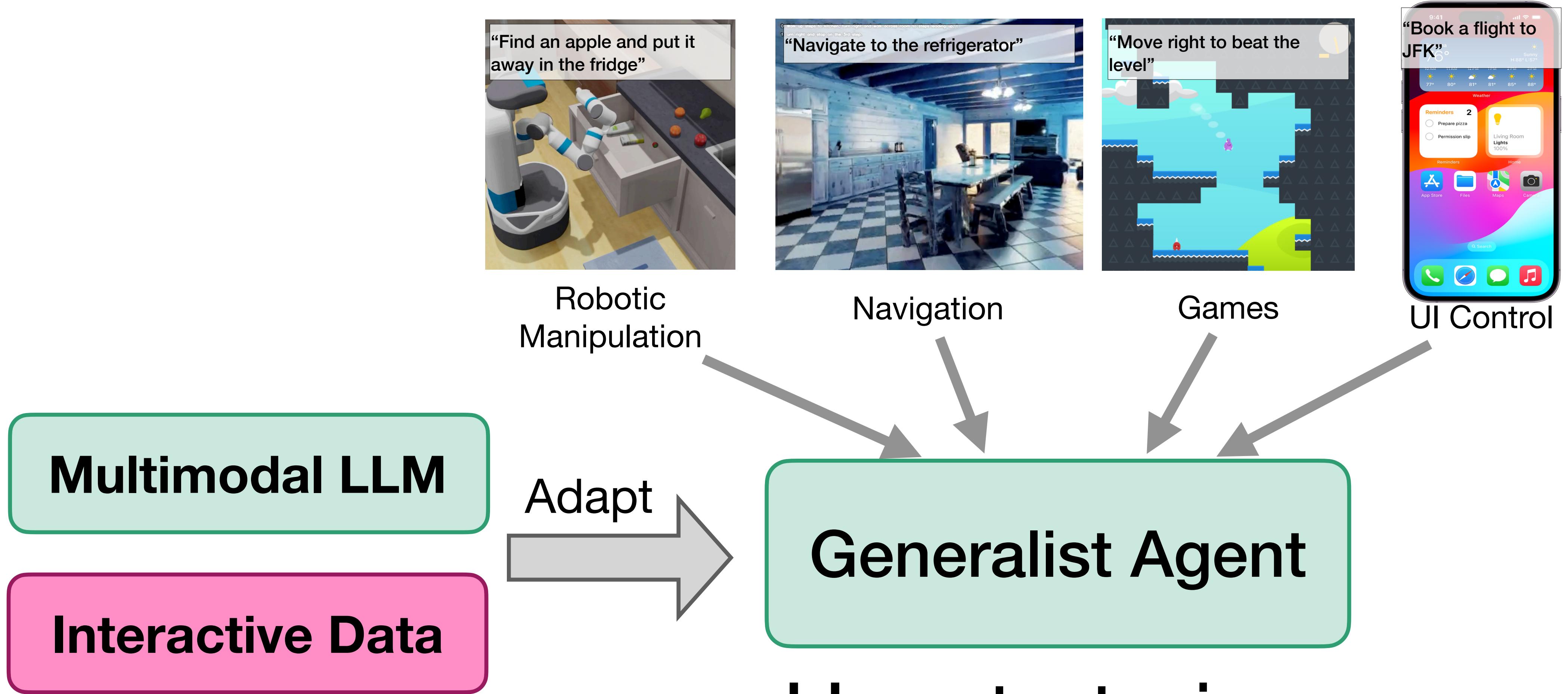
Robotic
Manipulation

Navigation

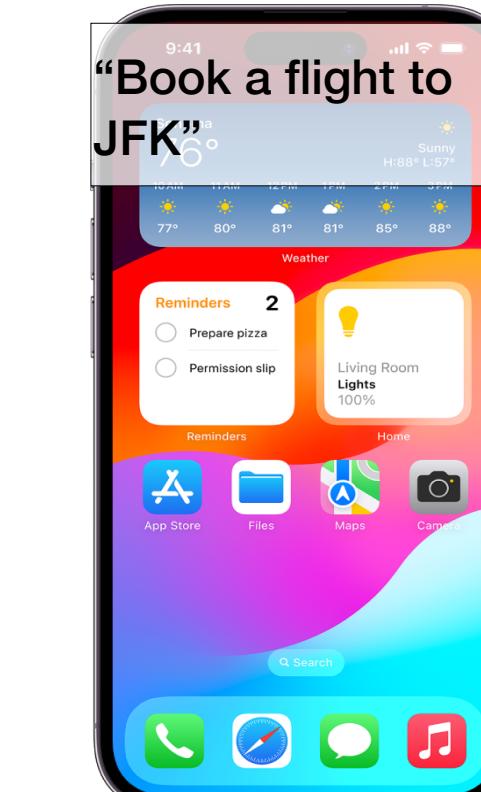
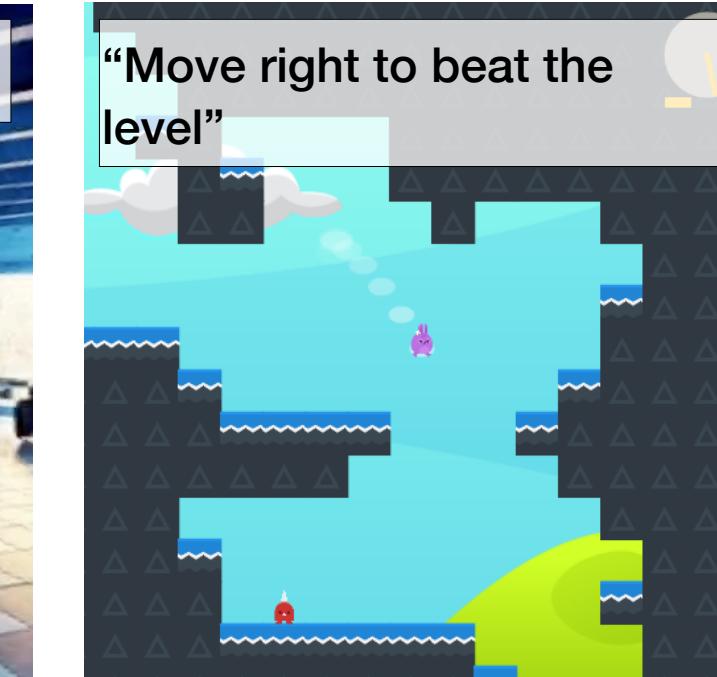
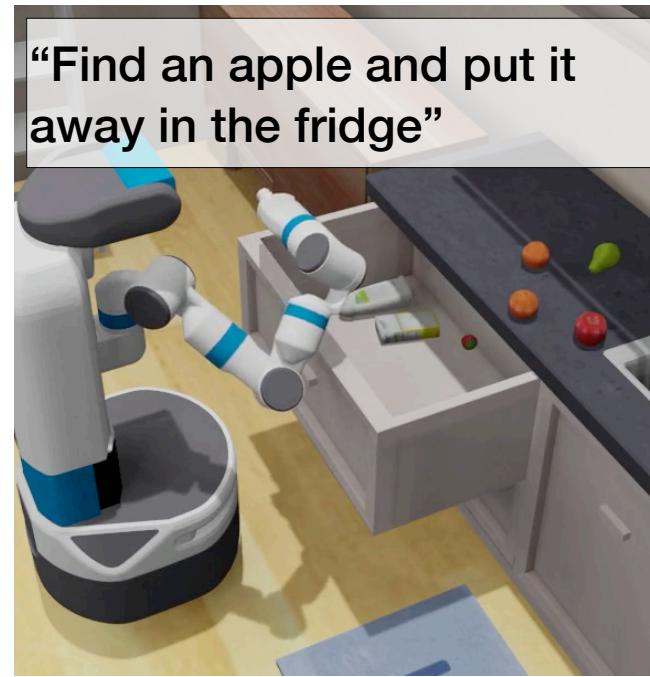
Games

UI Control

Generalist Agent



How to train a generalist agent?



Robotic
Manipulation

Navigation

Games

UI Control

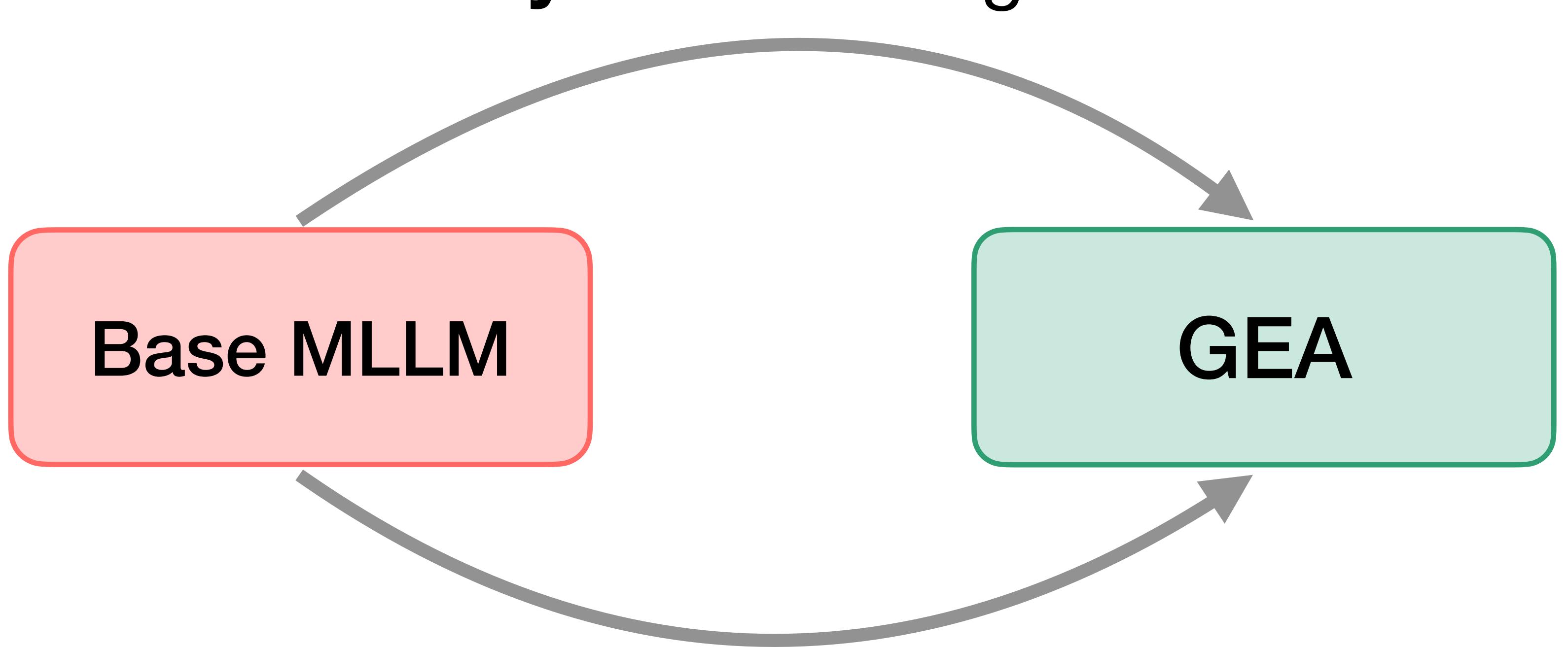
Generalist Agent

Rewards

Actions

**Reinforcement
Learning**

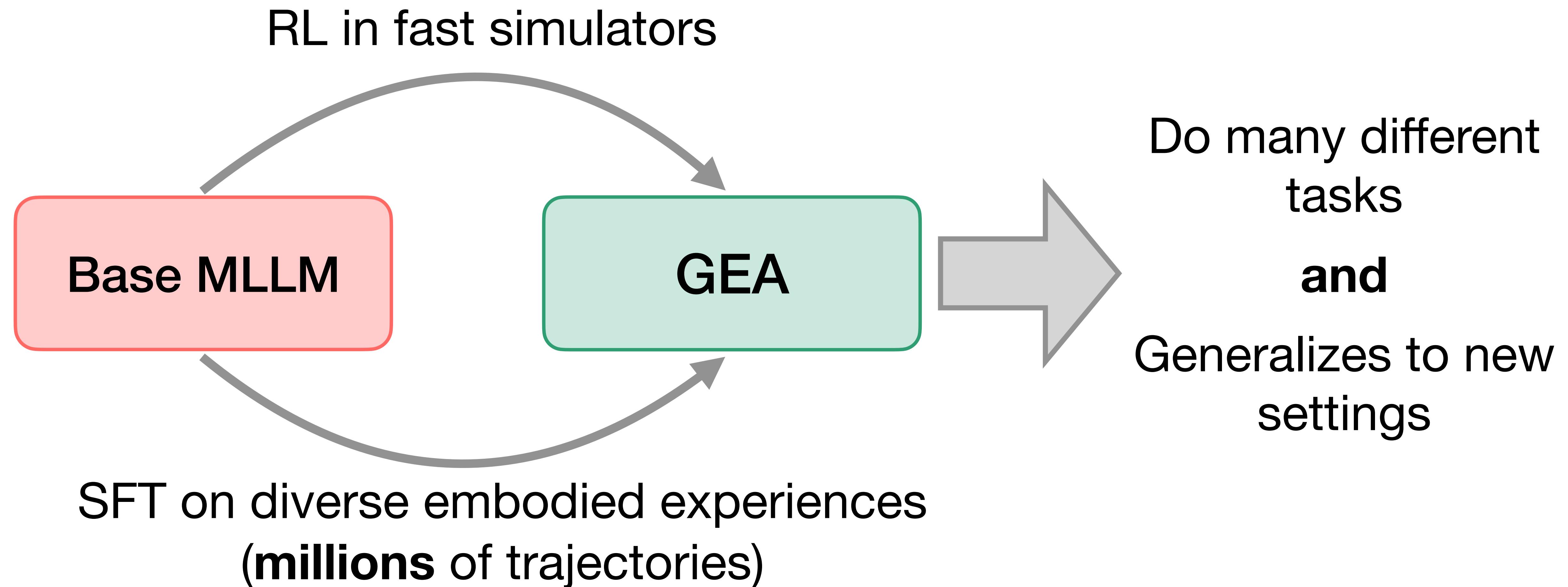
Generalist Embodied Agent (GEA)



RL in many simulated agentic tasks

Supervised Fine Tuning (SFT) on diverse
embodied experiences (**millions** of trajectories)

Generalist Embodied Agent (GEA)



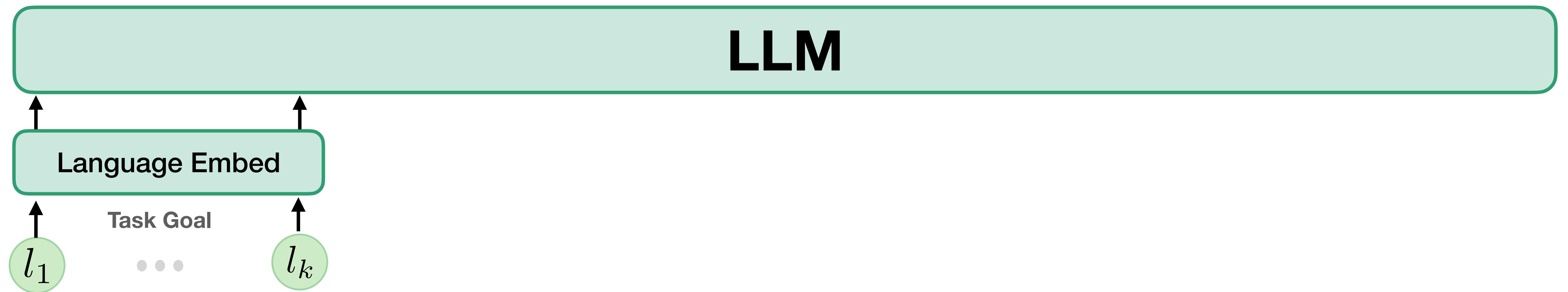
GEA Architecture

Start with a pretrained MLLM

LLM

GEA Architecture

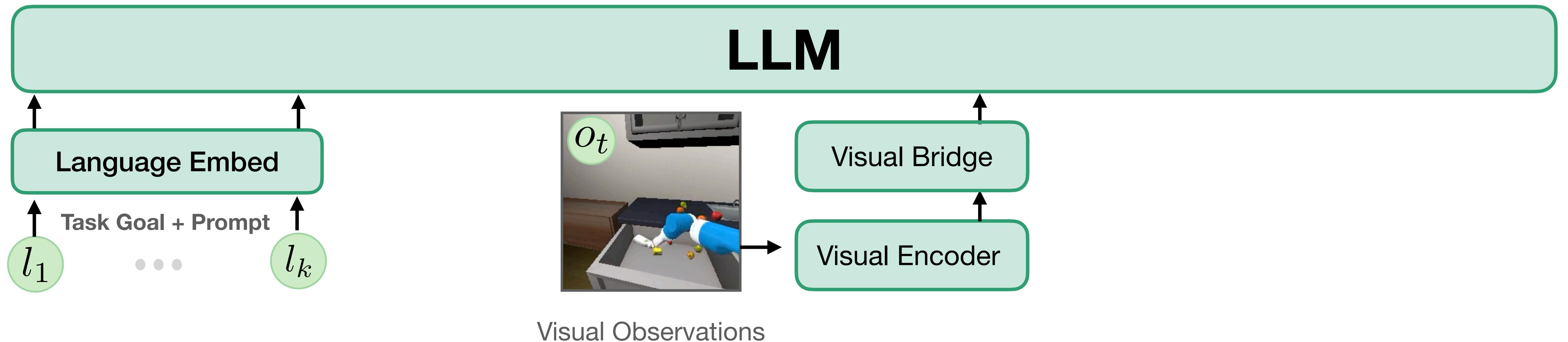
Tokenize and input task instruction to MLLM



Task Instruction: “Move all the fruit to the fridge”

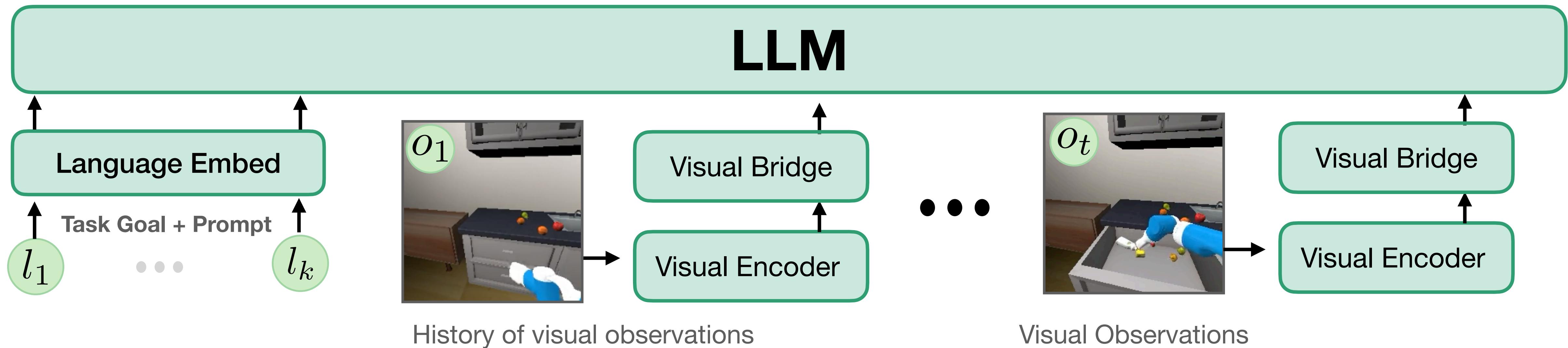
GEA Architecture

Encode visual observations



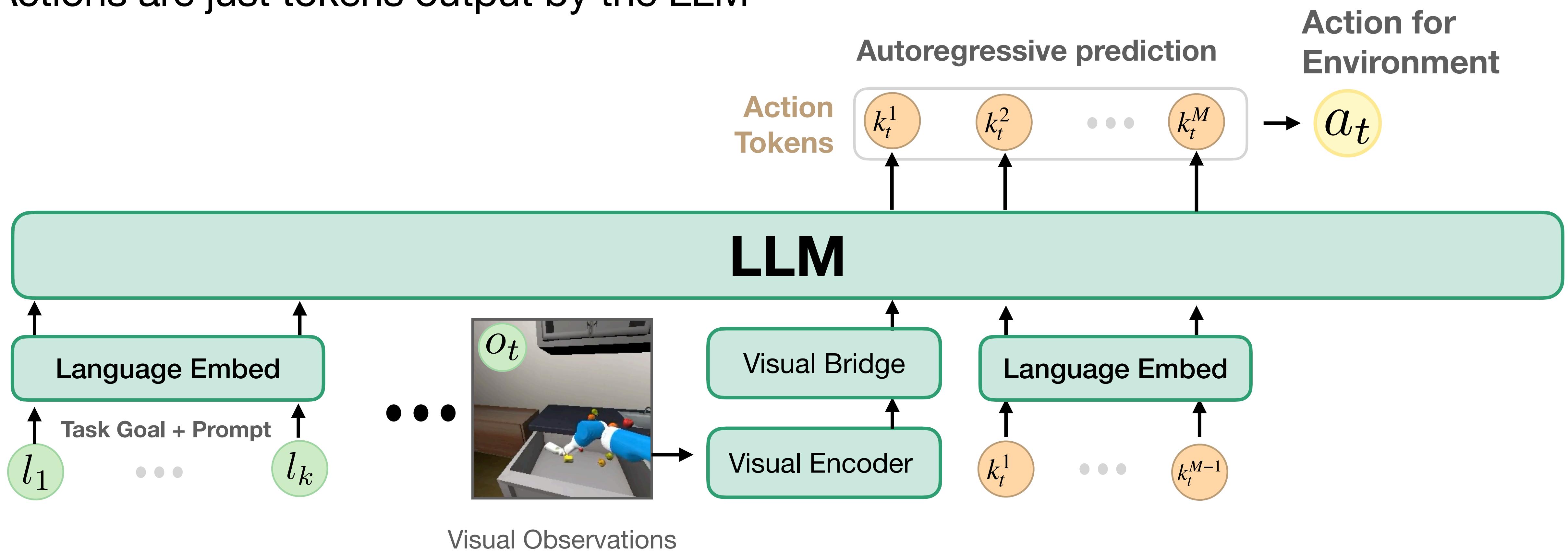
GEA Architecture

Encode visual observations and history of observations for memory



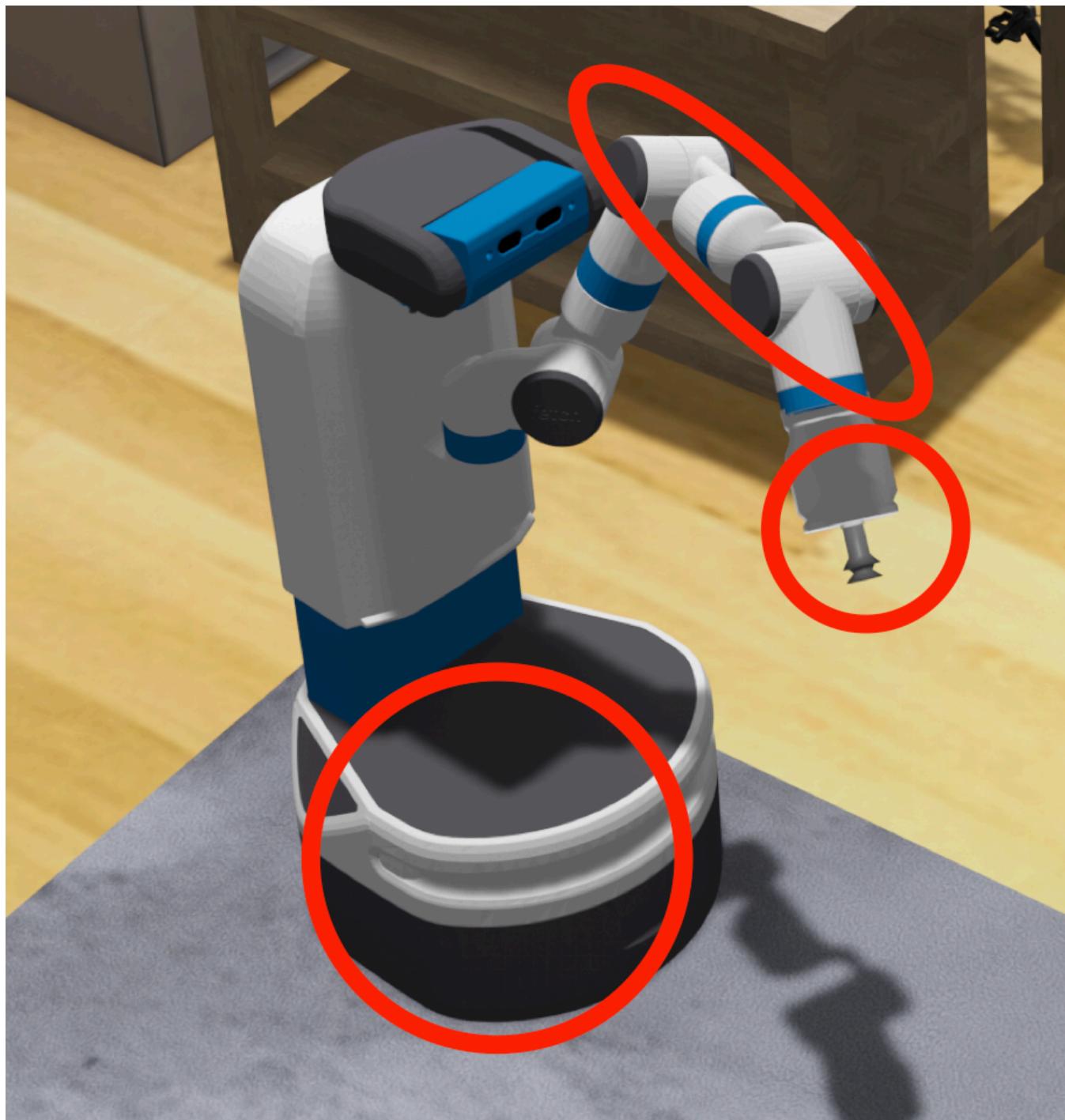
GEA Architecture

Actions are just tokens output by the LLM



But LLMs are trained to output **text**, yet agents require **actions**?

Continuous Low-Level Motor Control



[0.72, 0.24, 0.43, -0.21, ...]

Navigation Control Actions



Turn left
Turn right
Go straight

UI Interaction Actions



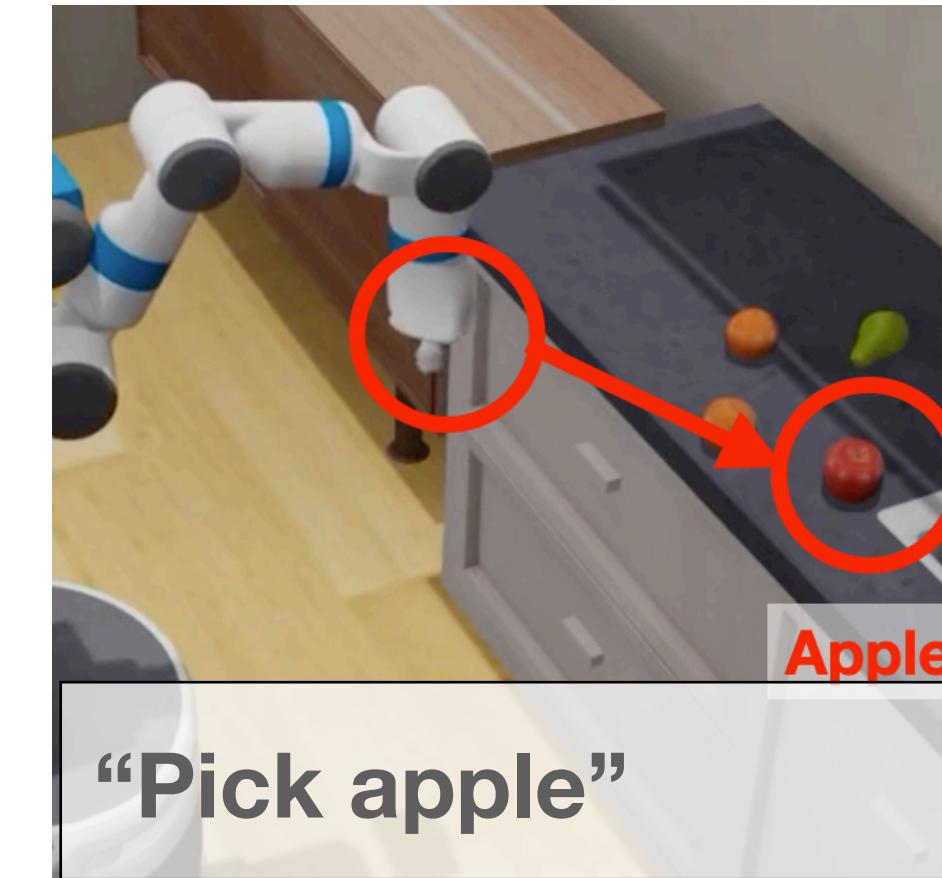
Open Safari
Tap 231 492
Search “food near me”
...

GEA Action Tokenization



Action is a continuous vector

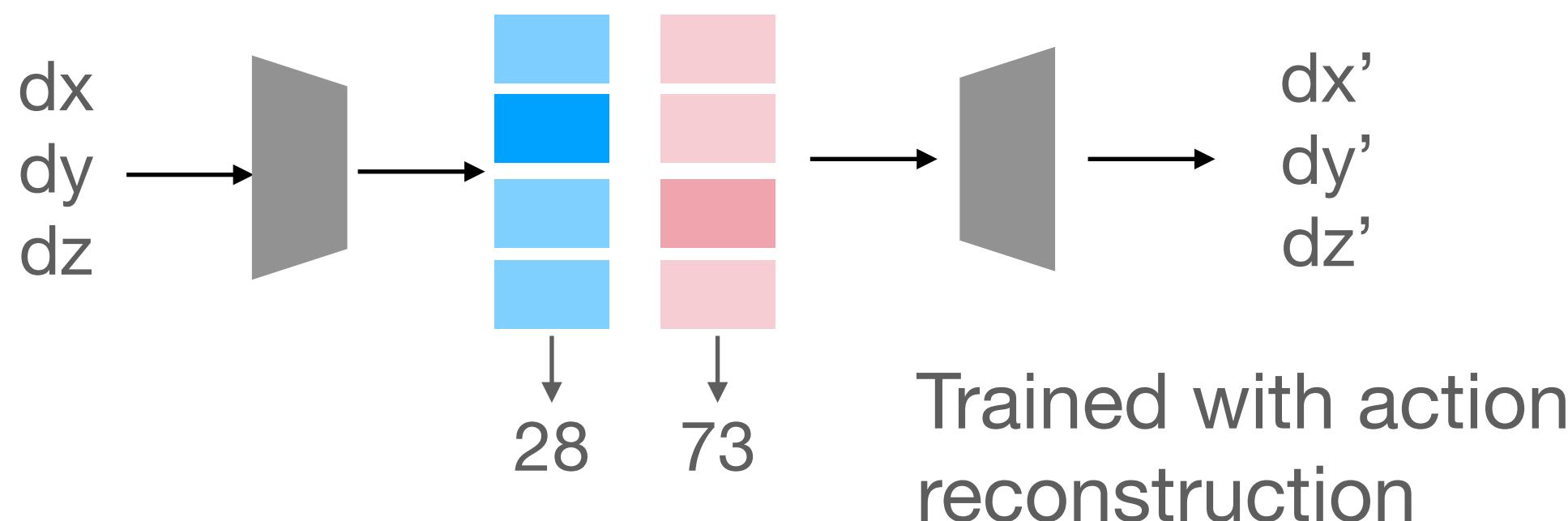
Example: end effector control
[dx, dy, dz]



Action is a selection from a set of discrete choices

Learned Tokenization

Residual VQ-VAE for discrete action tokenization



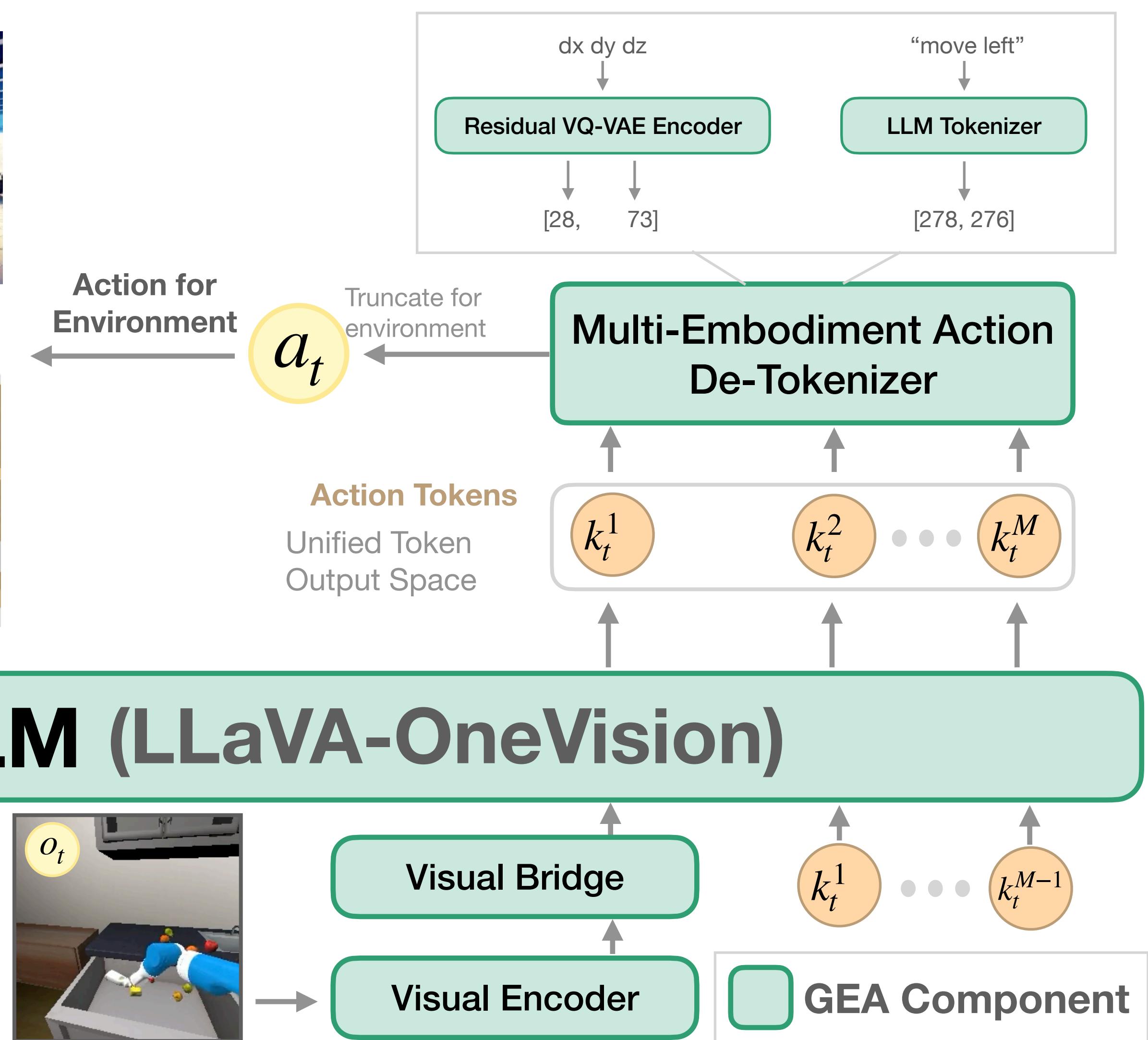
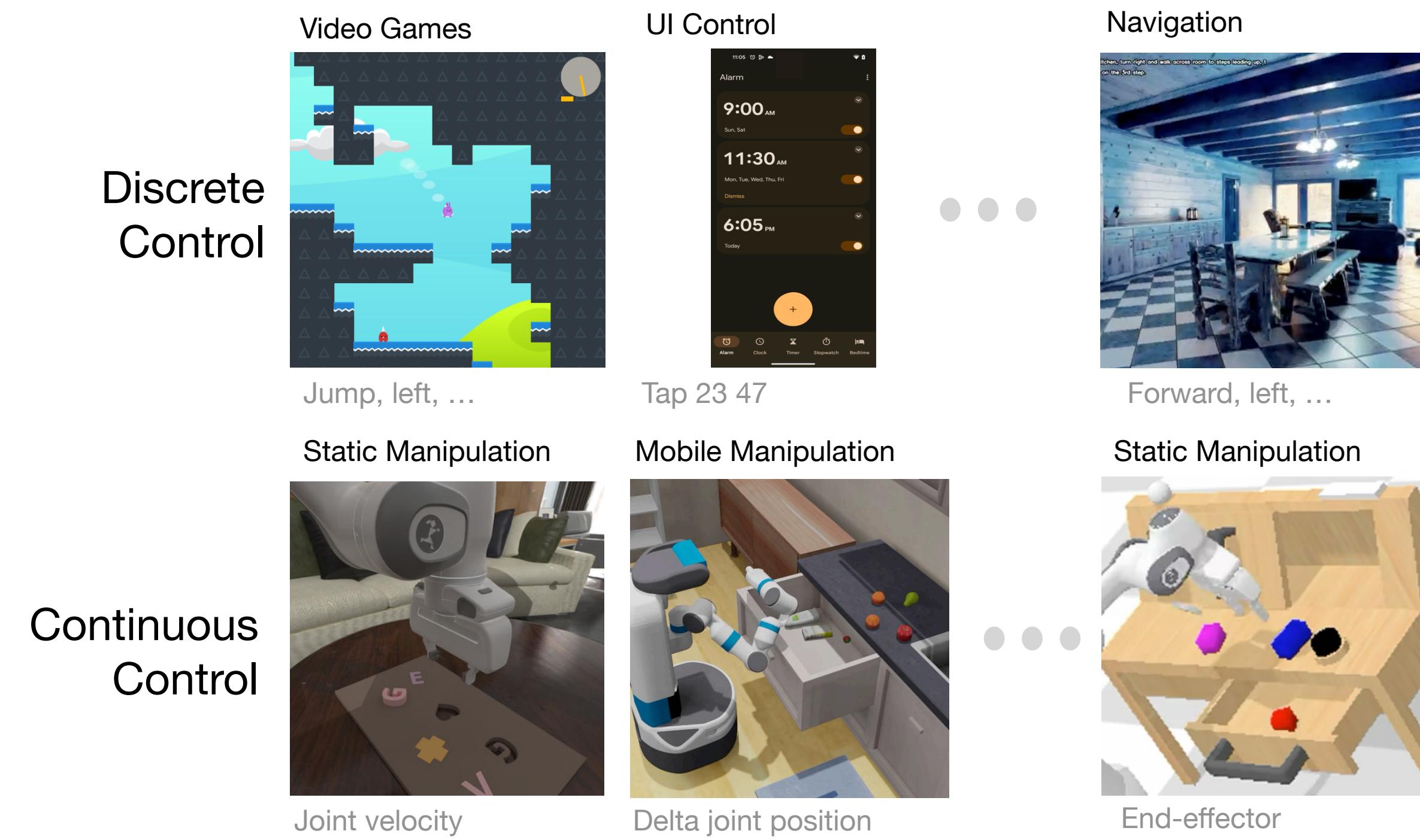
Semantic Tokenization

“pick apple”

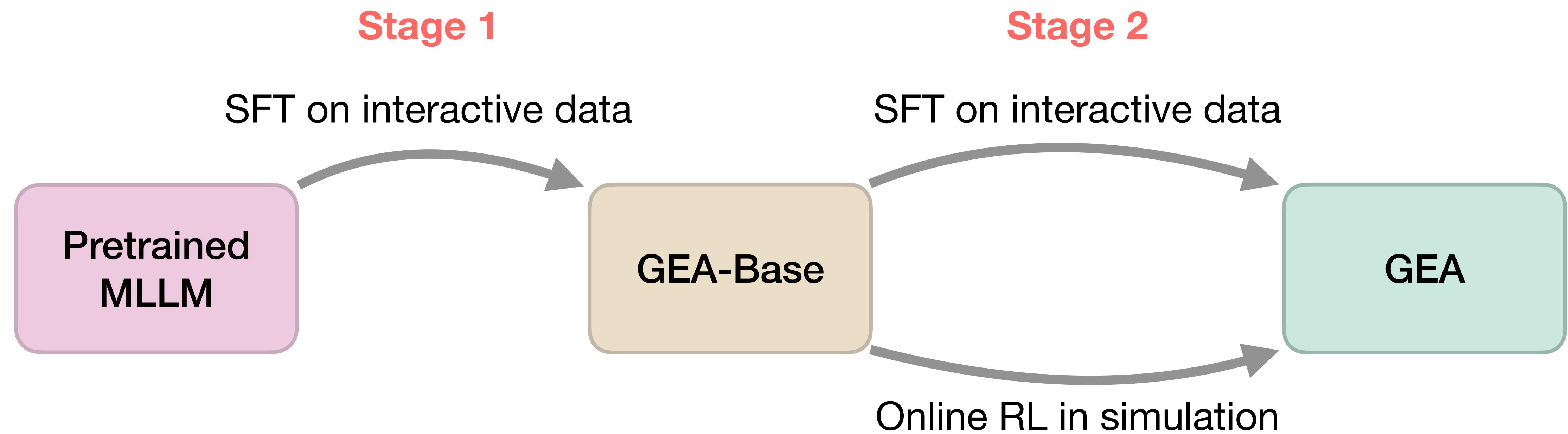


[278,276]

Describe action with language and tokenize with LLM vocabulary



Training GEA

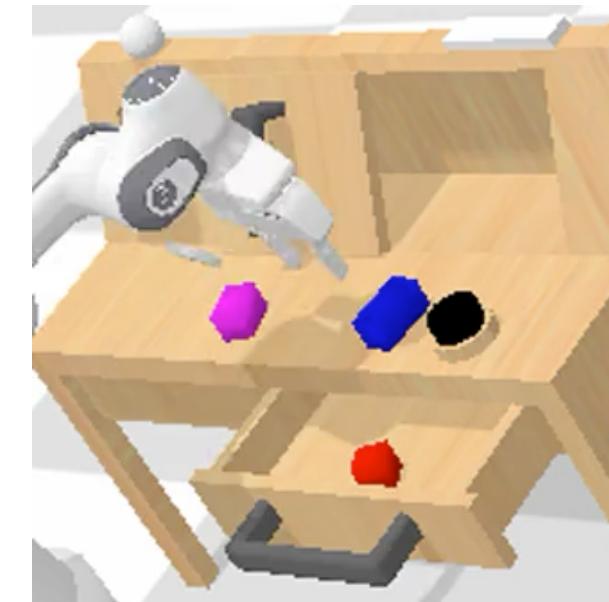
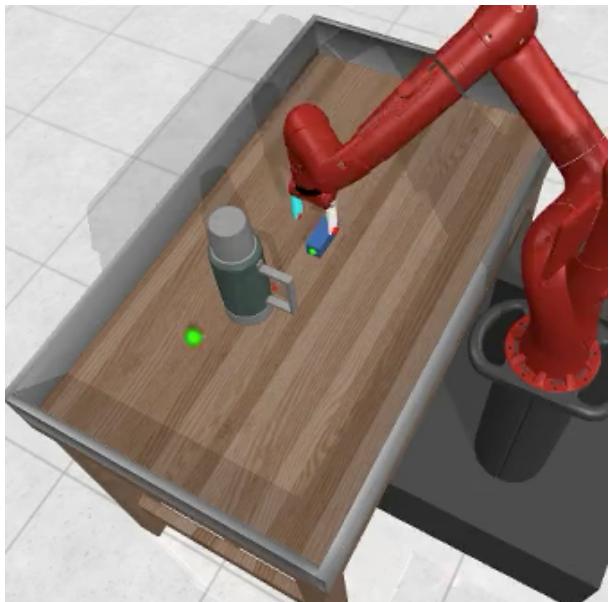


GEA Stage 1: SFT

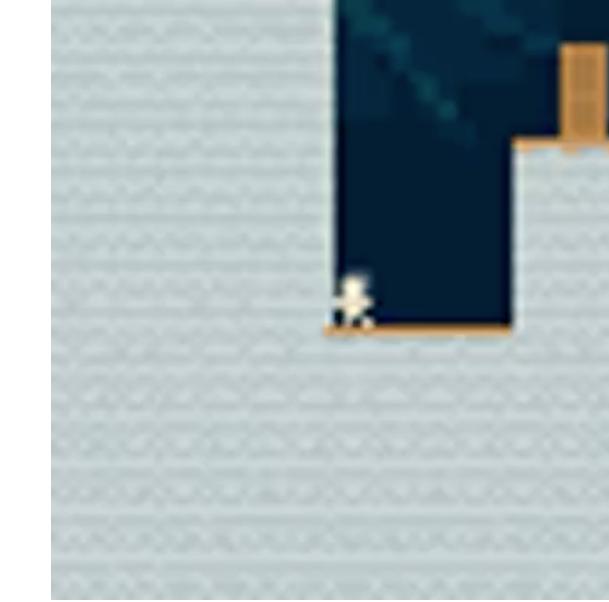
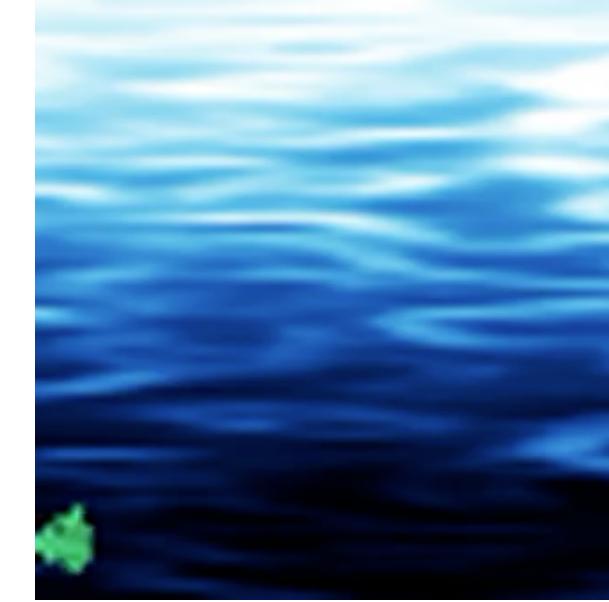
Collect expert demonstrations in diverse domains for training

From diverse sources, like scripted policies, humans, or RL policies

Static Manipulation



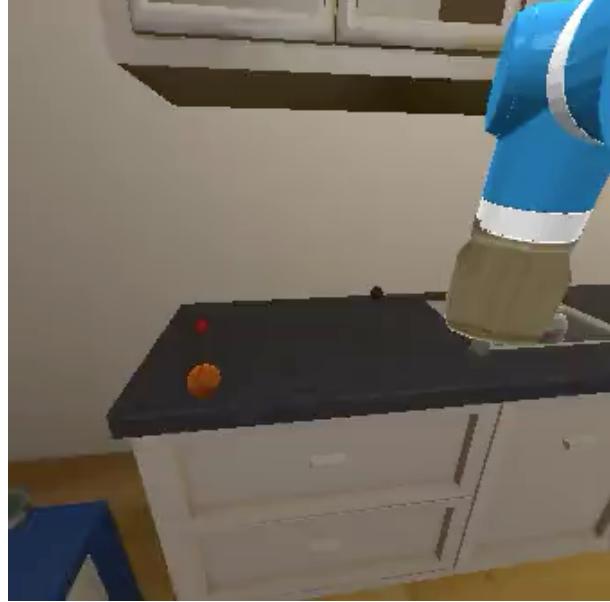
Games



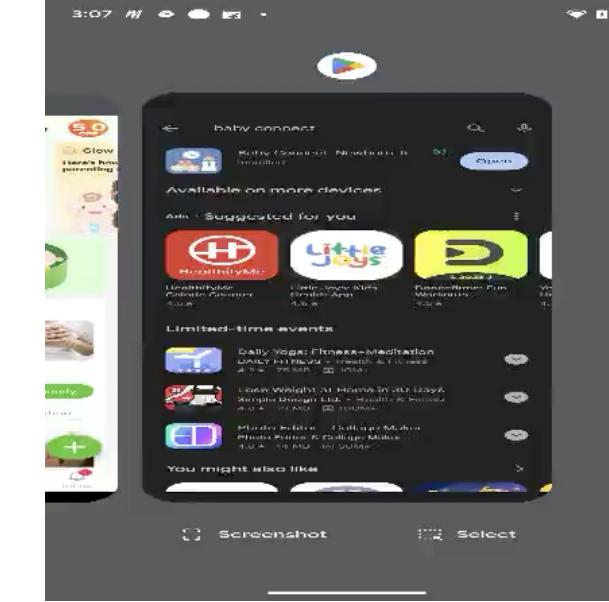
Navigation



Mobile Manipulation



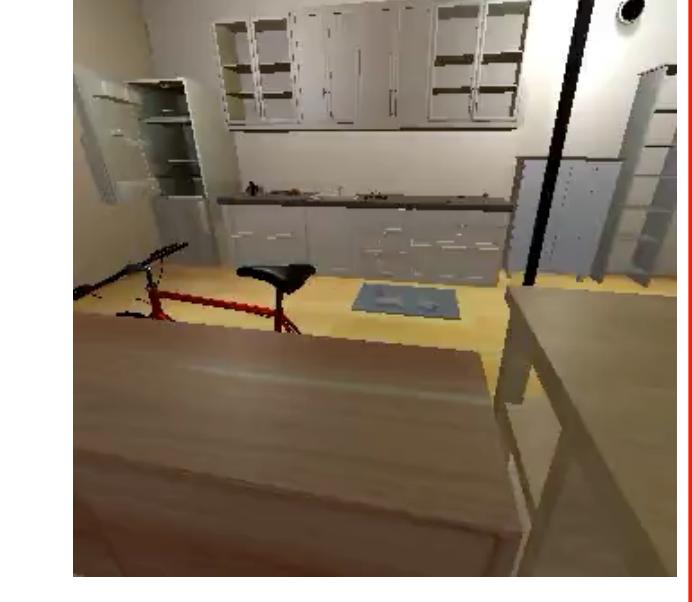
UI Control



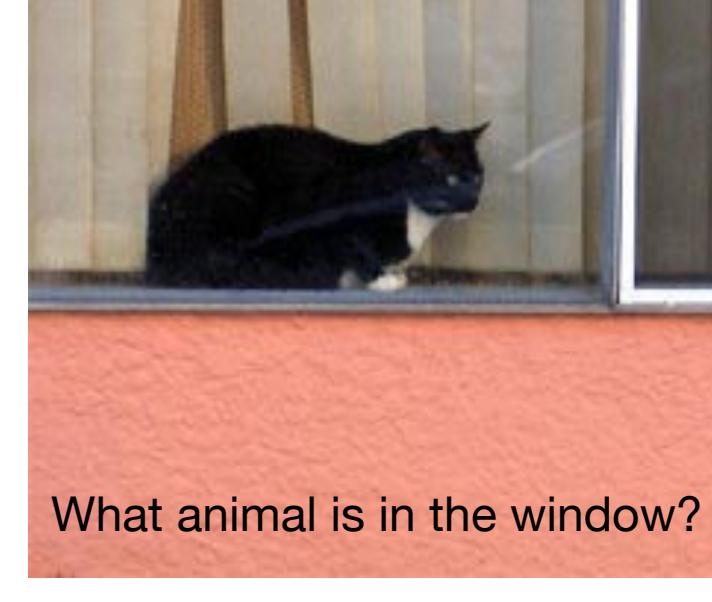
Real Robots



Planning



VL Data

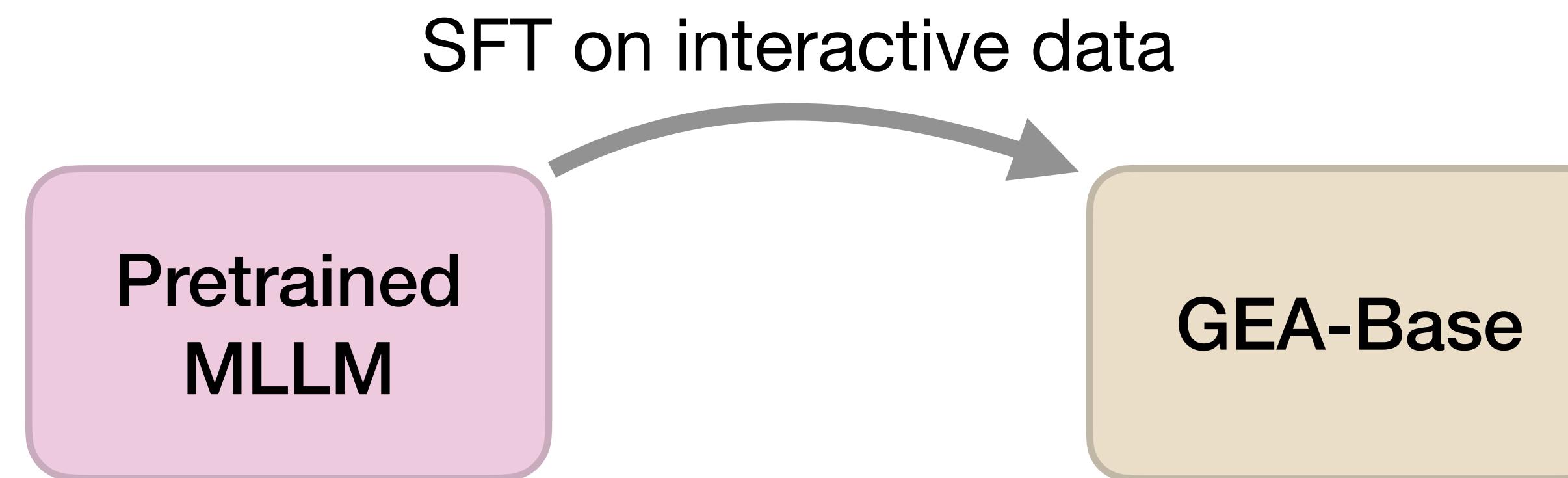


What animal is in the window?

GEA Stage 1: SFT

Collect expert demonstrations in diverse domains for training

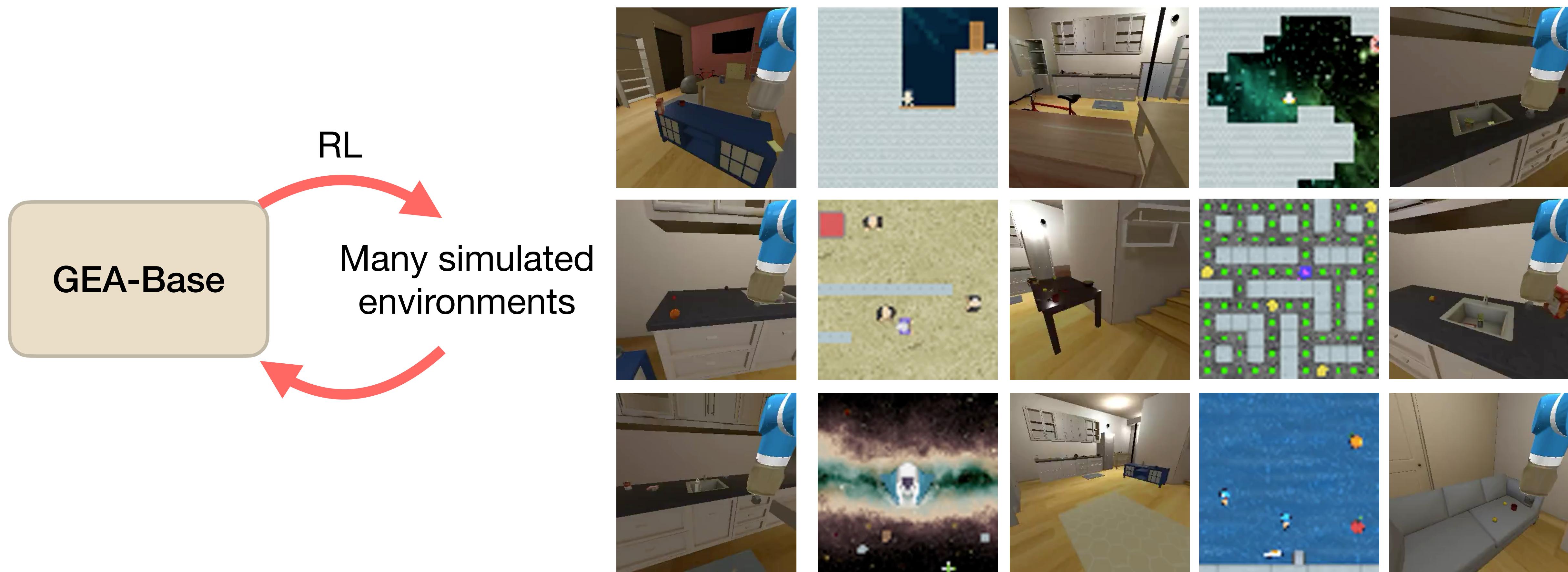
From diverse sources, like scripted policies, humans, or RL policies



2.2M trajectories, 90 embodiments

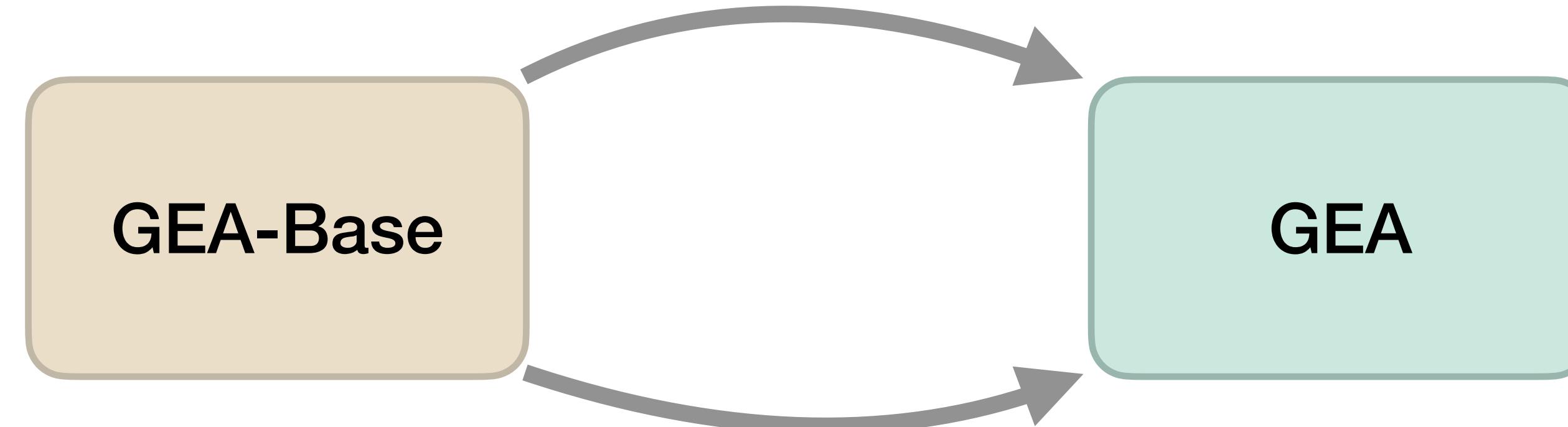
GEA Stage 2: RL + SFT

Continue training GEA-Base with RL in interactive tasks



Train with PPO (200M environment steps)

Online RL in simulation



SFT on interactive data

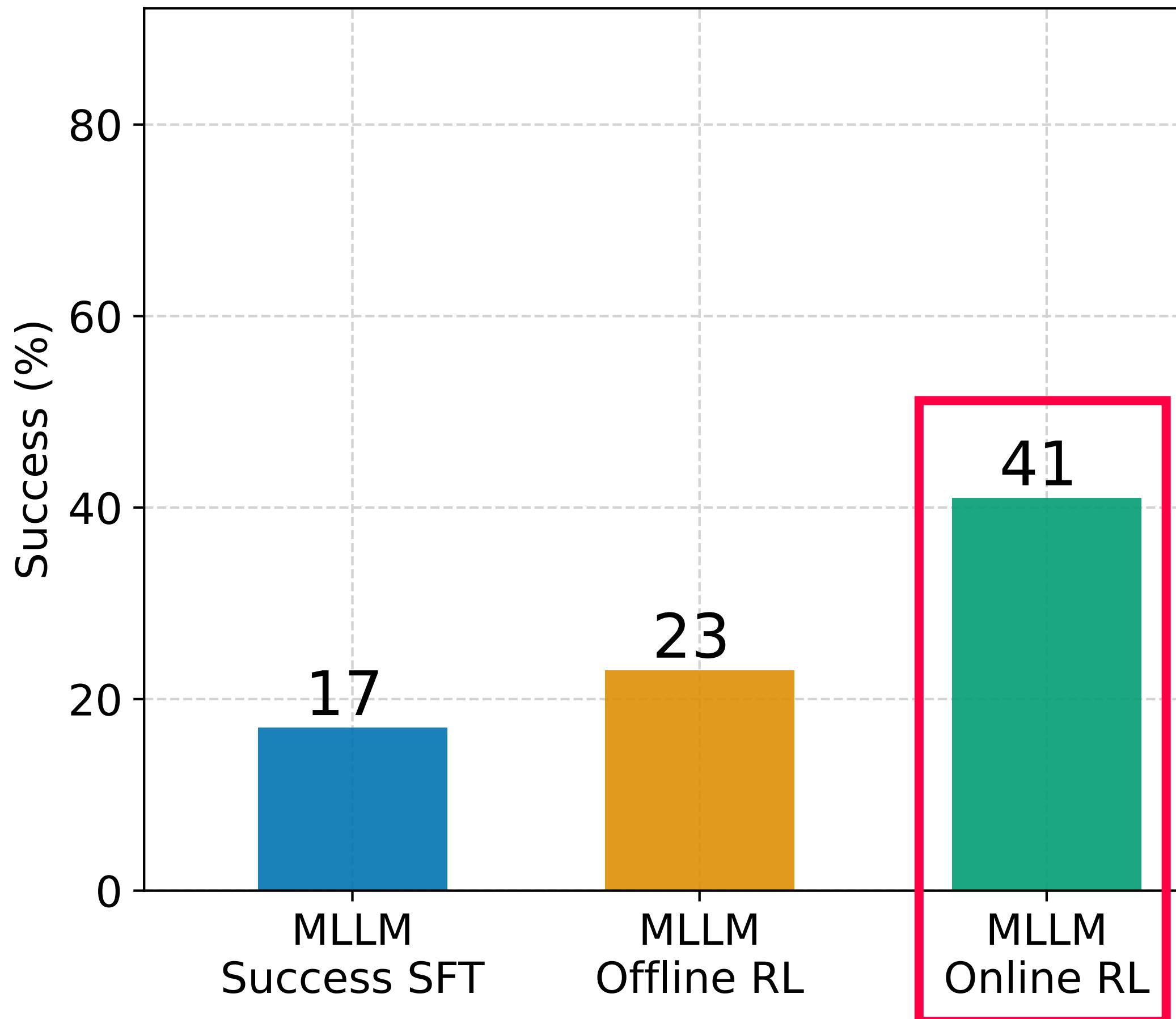
Importance of RL

- **Success SFT:** Collect data from policy, train on only successes
- **Offline RL:** Collect data from policy, train on both success and failure
- **Online RL:** Interact with the current policy in the environment

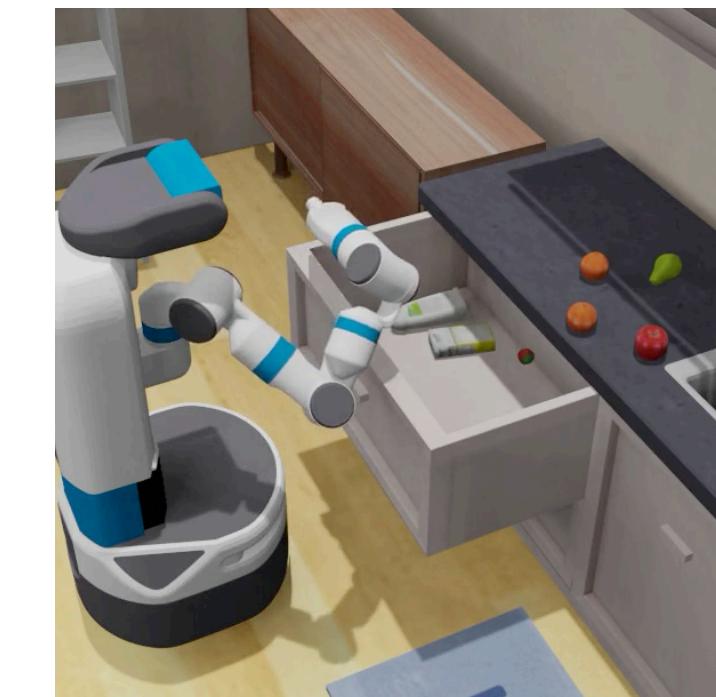
Importance of RL: A Glimpse of Results

On top of base MLLM (not GEA)

Online RL outperforms SFT and offline RL



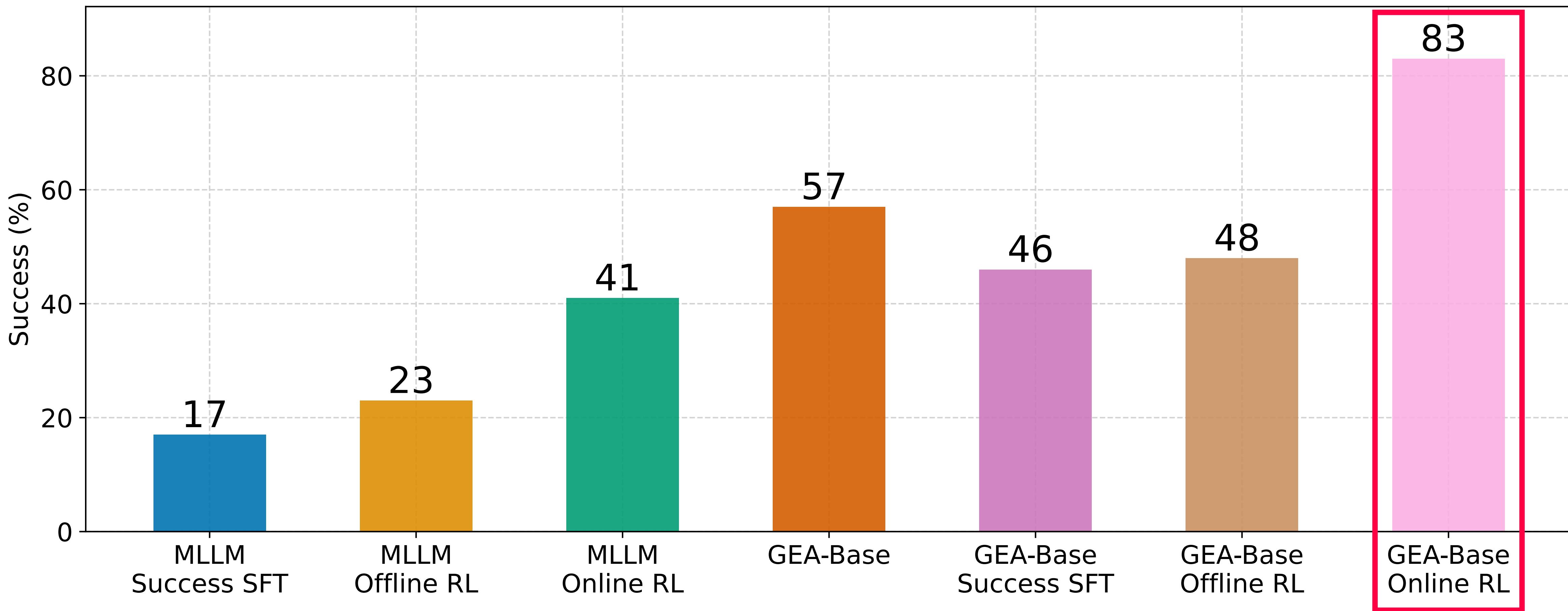
Analysis on a single task (Habitat Pick)



Importance of RL: A Glimpse of Results

Repeat starting from GEA-Base

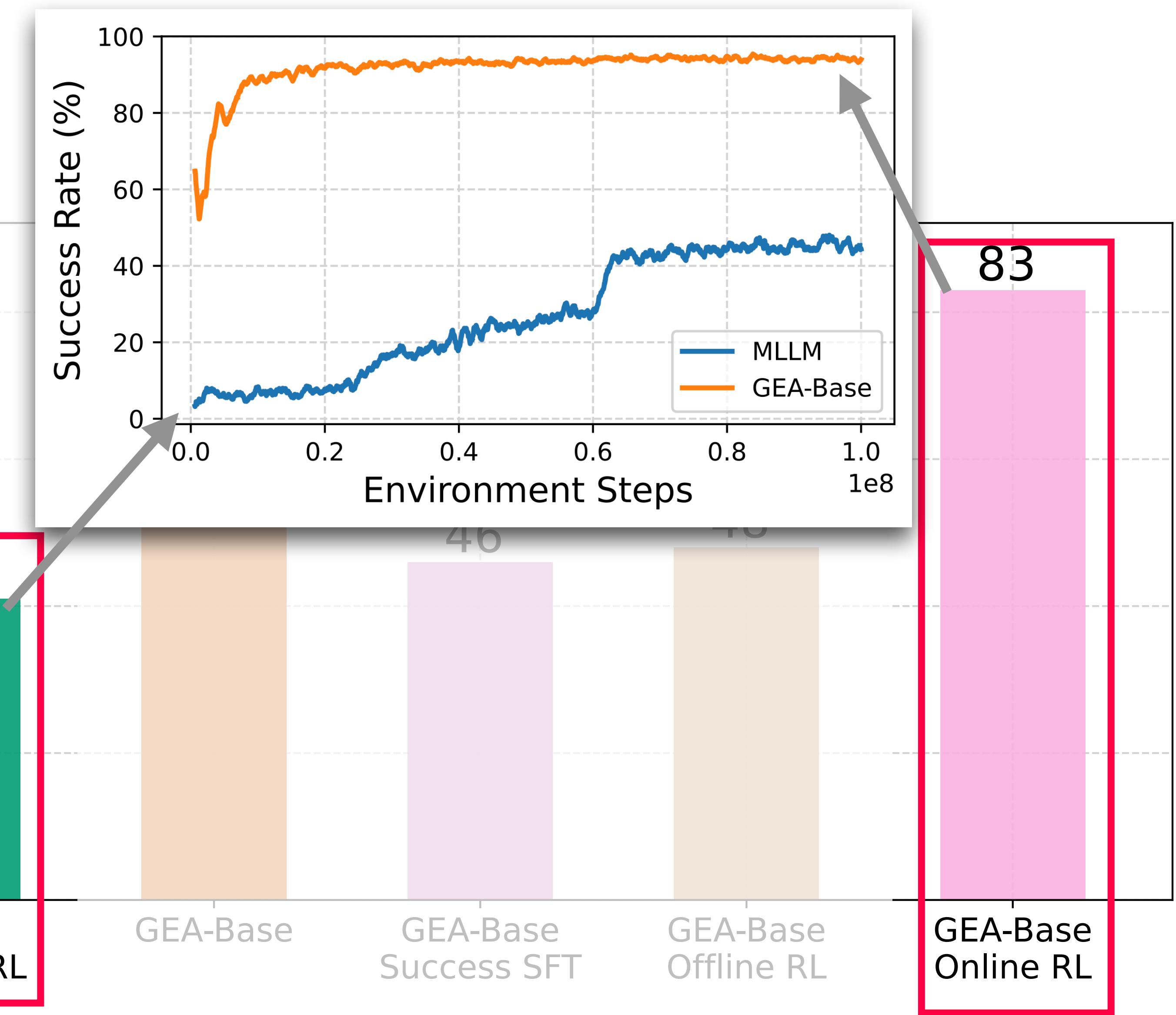
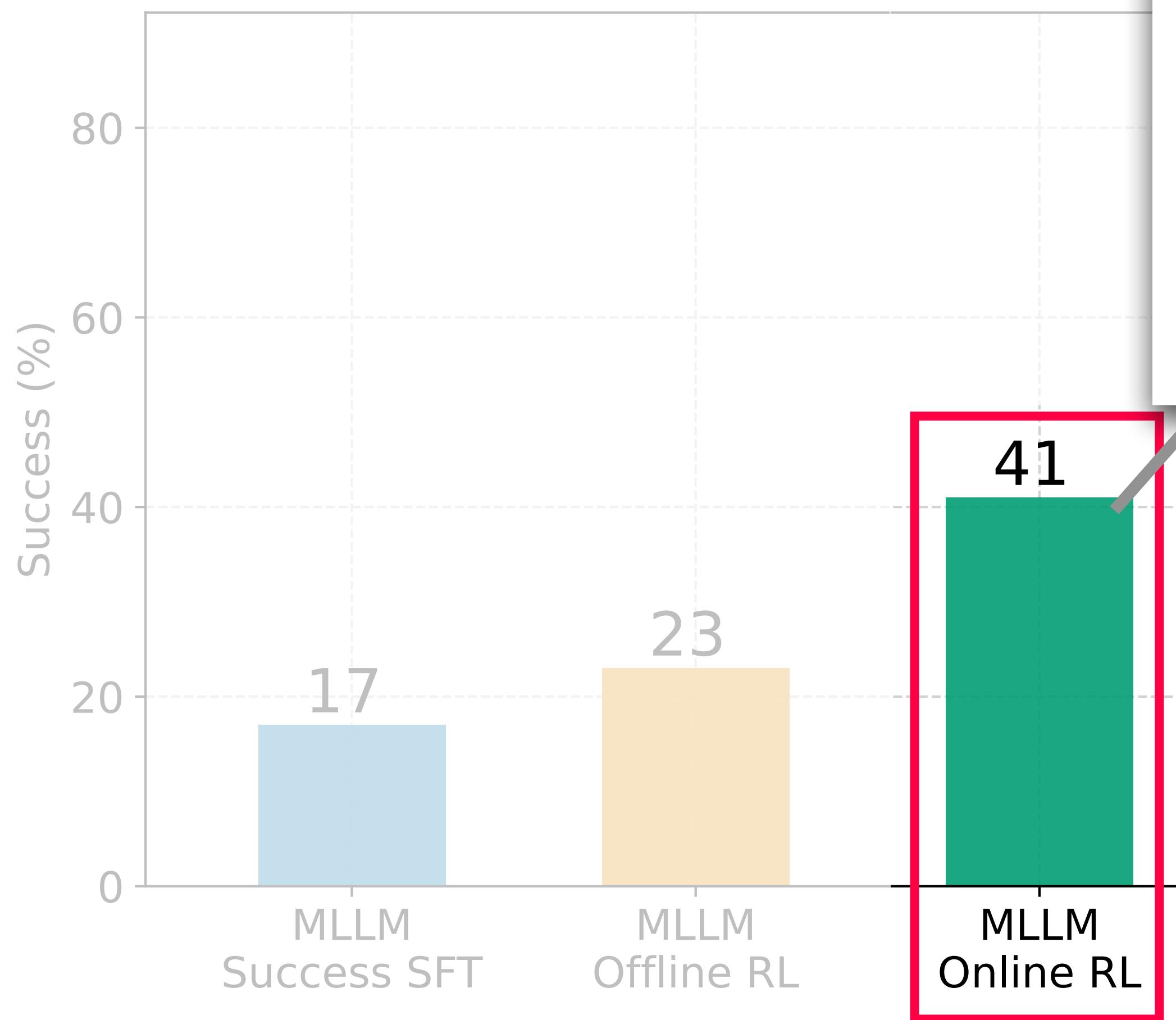
Online RL crucial for GEA



Importance of RL: A Glimpse of Results

GEA initialization accelerates RL

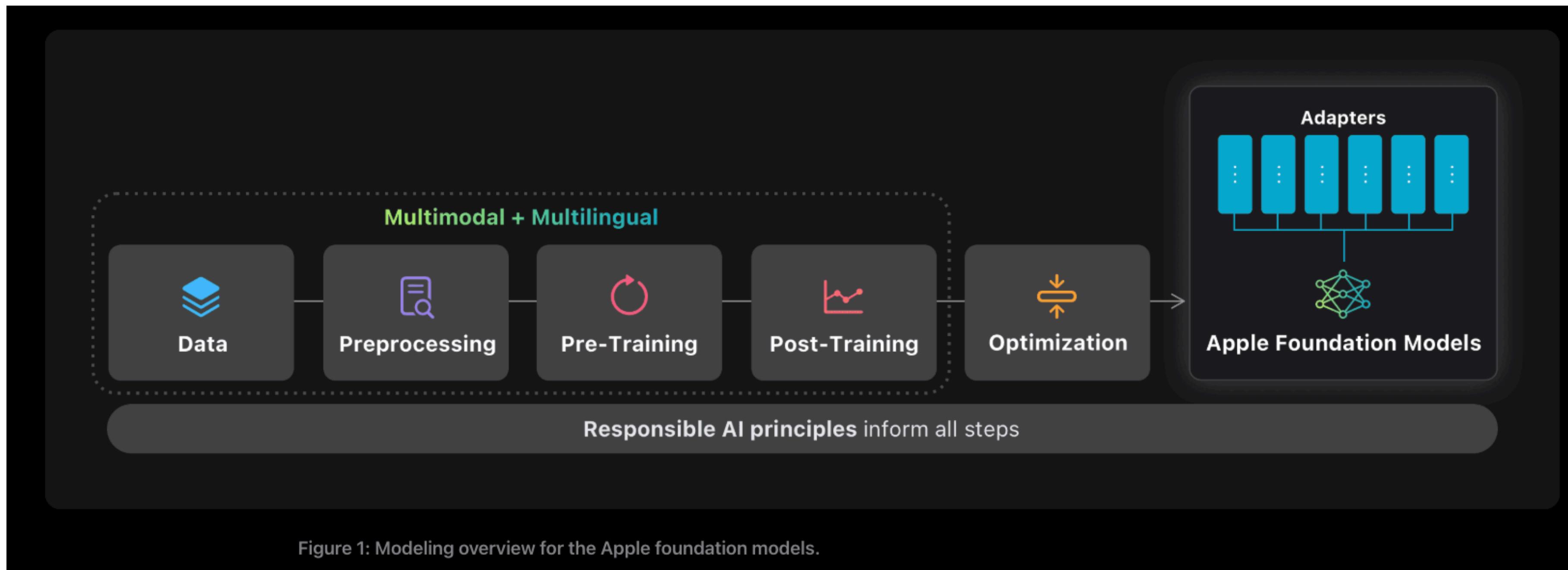
Two stage training is important



Summary

- **Image Encoder**: Simple method that scales well
 - Using MLLM eval suite as standard protocol for image encoder development
- **Multimodal LLM**: It's all about data
- **Generalist Agent**: RL is the key
- Future directions
 - Unified tokenizer for image understanding and generation
 - Reasoning
 - GUI Agents

Apple Foundation Models



<https://machinelearning.apple.com/research/apple-foundation-models-2025-updates>



TM and © 2025 Apple Inc. All rights reserved.