



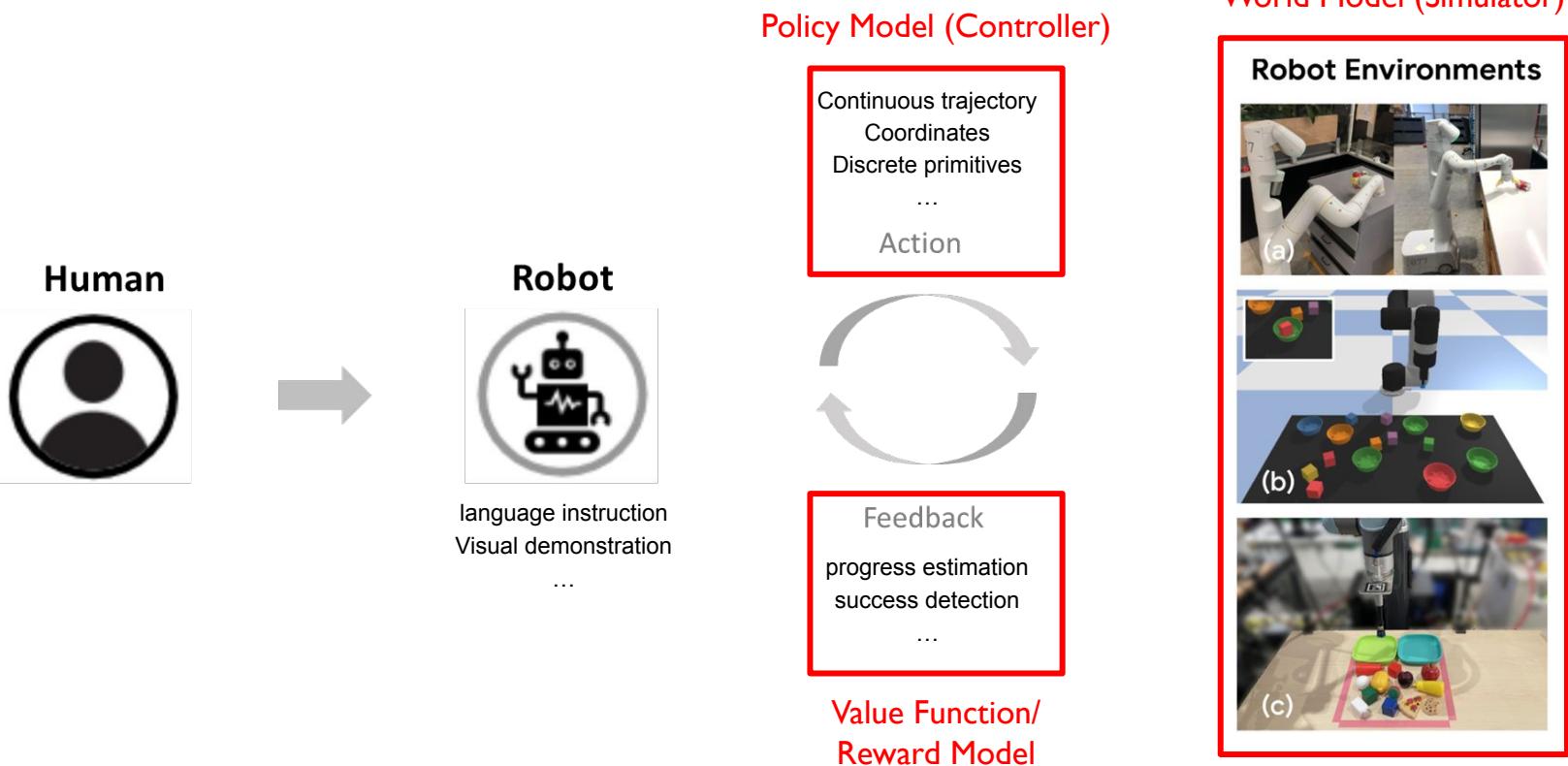
DATA 8005 Advanced Natural Language Processing

LLM/VLMs for Embodied AI

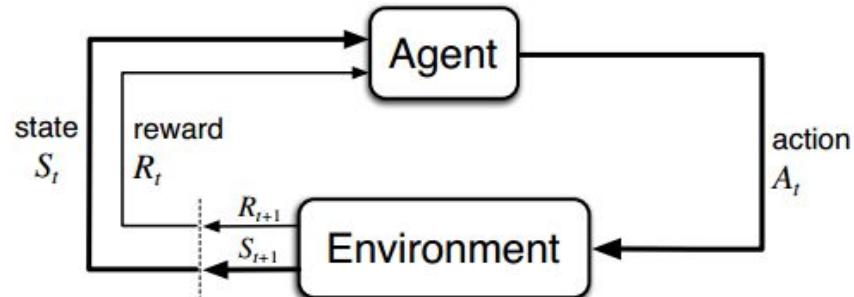
Yi Chen, Lu Qiu

Fall 2024

Background



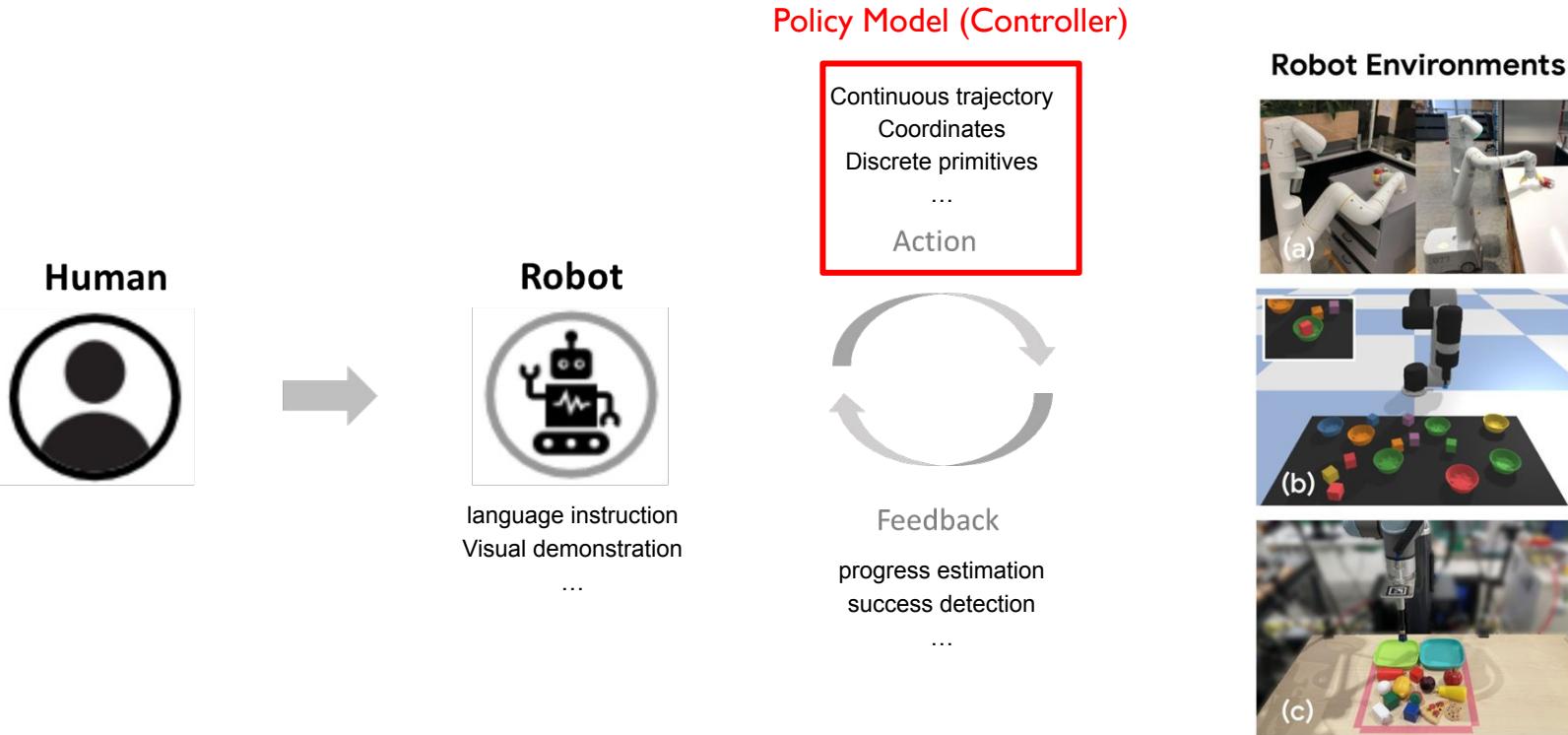
- Policy Model (Controller)
 - Make decision based on the current state and provide the next action
- Reward Model
 - Estimate reward / value to train the policy
- World model (Simulator)
 - Simulate the environment
 - Model-based RL



Papers

- Policy Model
 - [RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control](#)
 - [Do As I Can, Not As I Say: Grounding Language in Robotic Affordances](#)
- Value Function
 - [Vision Language Models are In-Context Value Learners](#)
- Simulator
 - [Genie: Generative Interactive Environments](#)

Background





DATA 8005 Advanced Natural Language Processing

Do As I Can, Not As I Say: Grounding Language in
Robotic Affordances

Lu Qiu

Fall 2024

Motivation

- LLM may respond with a reasonable narrative
- LLM's response without the context of what the robot is capable of given its abilities, the current state and the environment
- How can embodied agents extract and harness the knowledge of LLMs for physically grounded tasks?



Method Overview

- Do As I Can, Not As I Say (SayCan)
- Low-level controller:
 - The robot equipped with atomic skills
- High-level planner:
 - LLM to split high-level instruction into a series of skills
 - Select the optimal skill based on:
 - Scoring of LLM: probability that a skill is useful for the instruction
 - Affordance function: probability of successfully executing an individual skill

Connecting Large Language Models to Robots

- Prompting engineering: may produce inadmissible actions or language that is not formatted in a way that is easy to parse into individual steps
- Scoring language models
 - $p(w_k | w_{<k})$
 - Score a candidate completion selected from a set of options $p(l_\pi | i)$, where l_π is a given skill, and i is the instruction
 - Optimal skill can be calculated by $l_\pi = \operatorname{argmax}_{l_\pi \in l_\Pi} p(l_\pi | i)$

SayCan Pipeline

Algorithm 1 SayCan

Given: A high level instruction i , state s_0 , and a set of skills Π and their language descriptions ℓ_Π

```
1:  $n = 0, \pi = \emptyset$ 
2: while  $\ell_{\pi_{n-1}} \neq \text{"done"}$  do
3:    $\mathcal{C} = \emptyset$ 
4:   for  $\pi \in \Pi$  and  $\ell_\pi \in \ell_\Pi$  do
5:      $p_\pi^{\text{LLM}} = p(\ell_\pi | i, \ell_{\pi_{n-1}}, \dots, \ell_{\pi_0})$                                  $\triangleright$  Evaluate scoring of LLM
6:      $p_\pi^{\text{affordance}} = p(c_\pi | s_n, \ell_\pi)$                                       $\triangleright$  Evaluate affordance function
7:      $p_\pi^{\text{combined}} = p_\pi^{\text{affordance}} p_\pi^{\text{LLM}}$ 
8:      $\mathcal{C} = \mathcal{C} \cup p_\pi^{\text{combined}}$                                          I. calculate probability for each skill
9:   end for
10:   $\pi_n = \arg \max_{\pi \in \Pi} \mathcal{C}$ 
11:  Execute  $\pi_n(s_n)$  in the environment, updating state  $s_{n+1}$ 
12:   $n = n + 1$                                          2. select optimal skill
13: end while
```

- Scoring of LLM: how the skill makes progress toward completing the high-level instruction
- Affordance function: make the LLM aware of the current state

SayCan Pipeline

Instruction Relevance with LLMs



Combined

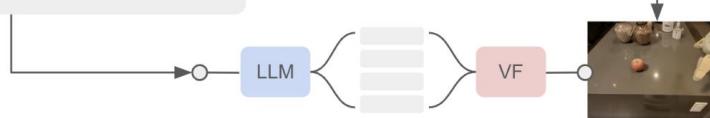
-6	Find an apple	0.6
-30	Find a coke	0.6
-30	Find a sponge	0.6
-4	Pick up the apple	0.2
-30	Pick up the coke	0.2
...
-5	Place the apple	0.1
-30	Place the coke	0.1
-10	Go to the table	0.8
-20	Go to the counter	0.8

Skill Affordances with Value Functions



Value Functions

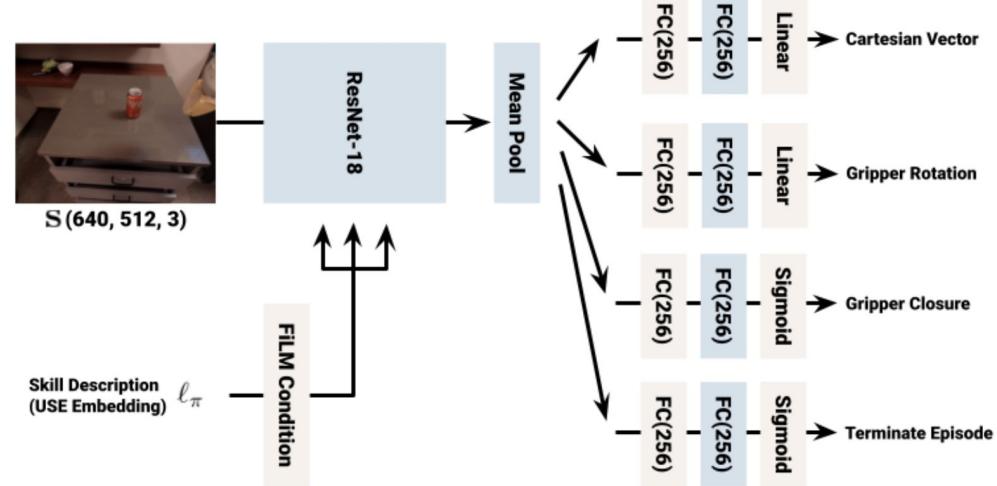
I would: 1. **Find an apple**, 2. _____



3. append skill to i and repeat

SayCan Pipeline

- Affordance model
 - temporal-difference-based RL
- Policy of low-level control
 - image-based BC
 - RL

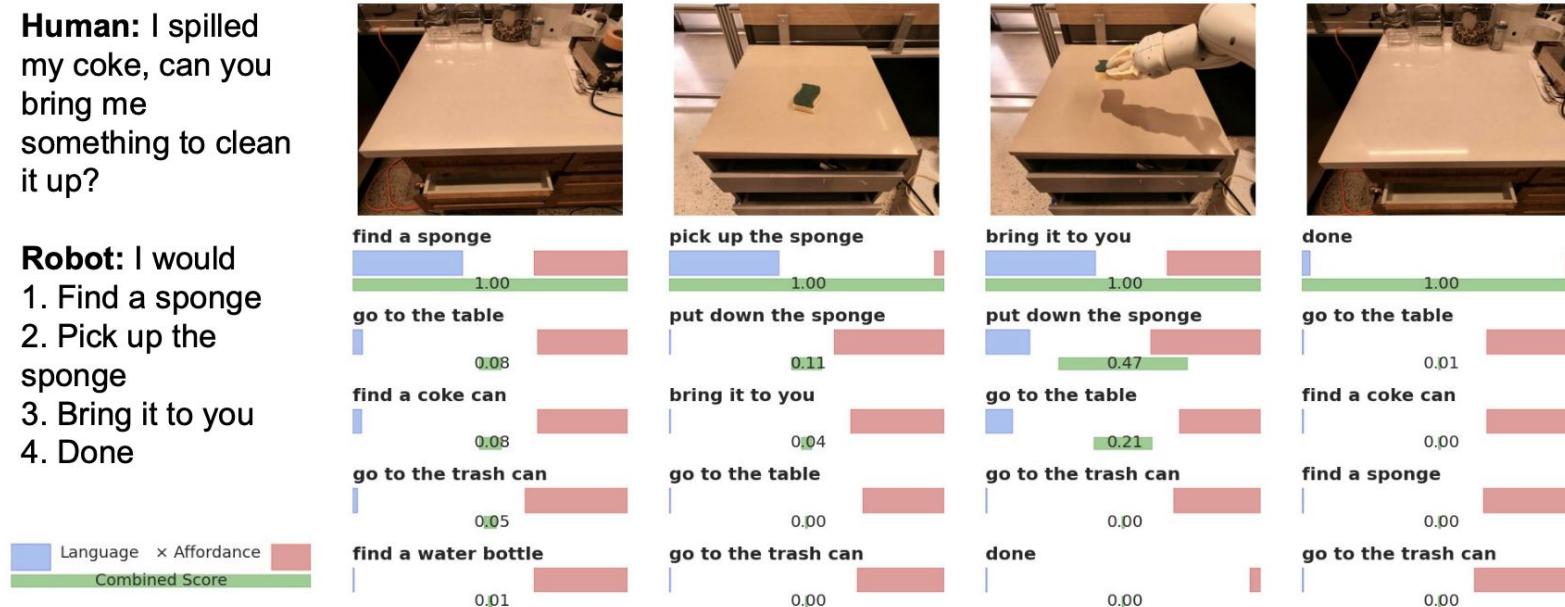


Universal Sentence Encoder

Results

Human: I spilled my coke, can you bring me something to clean it up?

Robot: I would
1. Find a sponge
2. Pick up the sponge
3. Bring it to you
4. Done



Results

- PaLM-SayCan and the underlying policies generalize reasonably well to the full kitchen
- The necessity of the affordance grounding
- Larger models perform better

Family	Num	Mock Kitchen		Kitchen		No Affordance		No LLM	
		PaLM-SayCan	PaLM-SayCan	PaLM-SayCan	PaLM-SayCan	No VF	Gen.	BC NL	BC USE
NL Single	15	100%	100%	93%	87%	73%	87%	0%	60%
NL Nouns	15	67%	47%	60%	40%	53%	53%	0%	0%
NL Verbs	15	100%	93%	93%	73%	87%	93%	0%	0%
Structured	15	93%	87%	93%	47%	93%	100%	0%	0%
Embodiment	11	64%	55%	64%	55%	18%	36%	0%	0%
Crowd Sourced	15	87%	87%	73%	60%	67%	80%	0%	0%
Long-Horizon	15	73%	47%	73%	47%	67%	60%	0%	0%
Total	101	84%	74%	81%	60%	67%	74%	0%	9%

Family	Num	PaLM 540B [9]	PaLM 62B	PaLM 8B	FLAN 137B [8]
NL Single	15	87%	73%	20%	40%
NL Nouns	15	53%	47%	20%	40%
NL Verbs	15	93%	100%	60%	87%
Structured	15	100%	100%	67%	73%
Embodiment	11	36%	27%	27%	0%
Crowd Sourced	15	80%	73%	47%	47%
Long-Horizon	15	60%	73%	20%	0%
Total	101	74%	72%	38%	43%



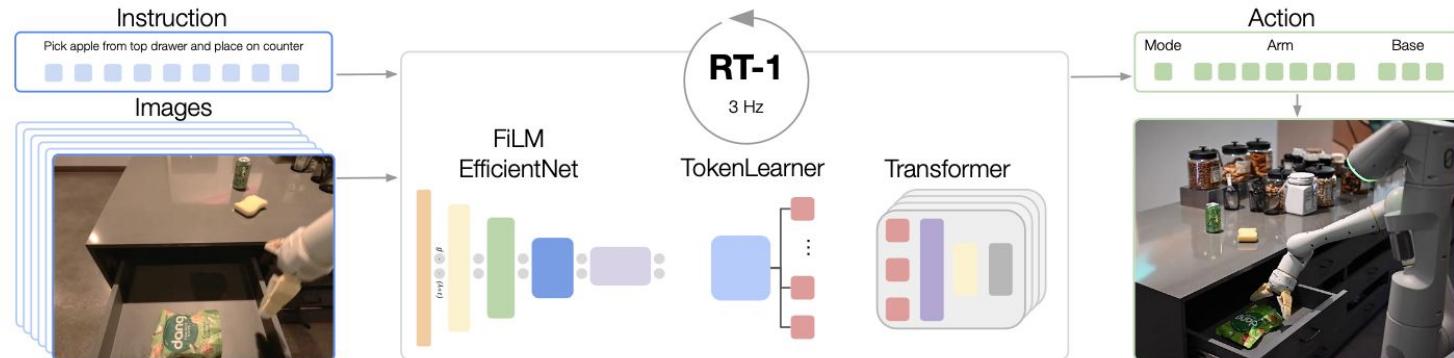
DATA 8005 Advanced Natural Language Processing

RT-2: Vision-Language-Action Models Transfer Web
Knowledge to Robotic Control

Fall 2024

About RTI

- End-to-end vision-language-action (VLA) model, 35M
- A large multi-task backbone model on data consisting of a wide variety of robotic tasks (13 robots, containing ~130k episodes and over 700 tasks)
- Action tokenization + cross-entropy loss

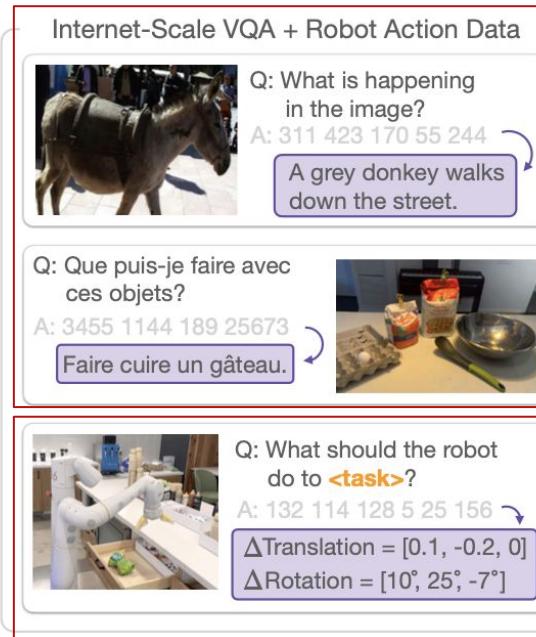


Motivation for RT2

- General robotic model / generalist robots
 - Map robot observations to actions end-to-end
 - Open-ended task-agnostic training
 - Collecting millions of robotic interaction trials
 - Enjoy the benefits of pretraining on Internet-scale data

Method

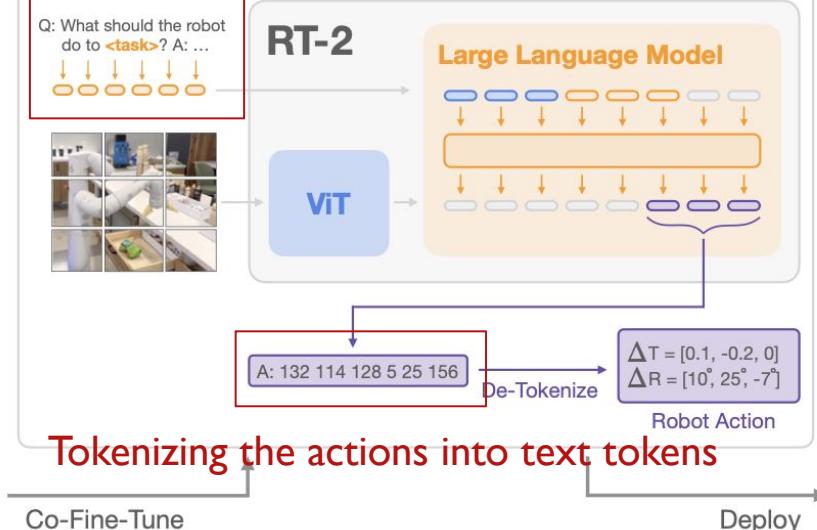
Internet-scale vision-language tasks



Robot action task

Robot action task in VQA format

Vision-Language-Action Models for Robot Control



Closed-Loop Robot Control

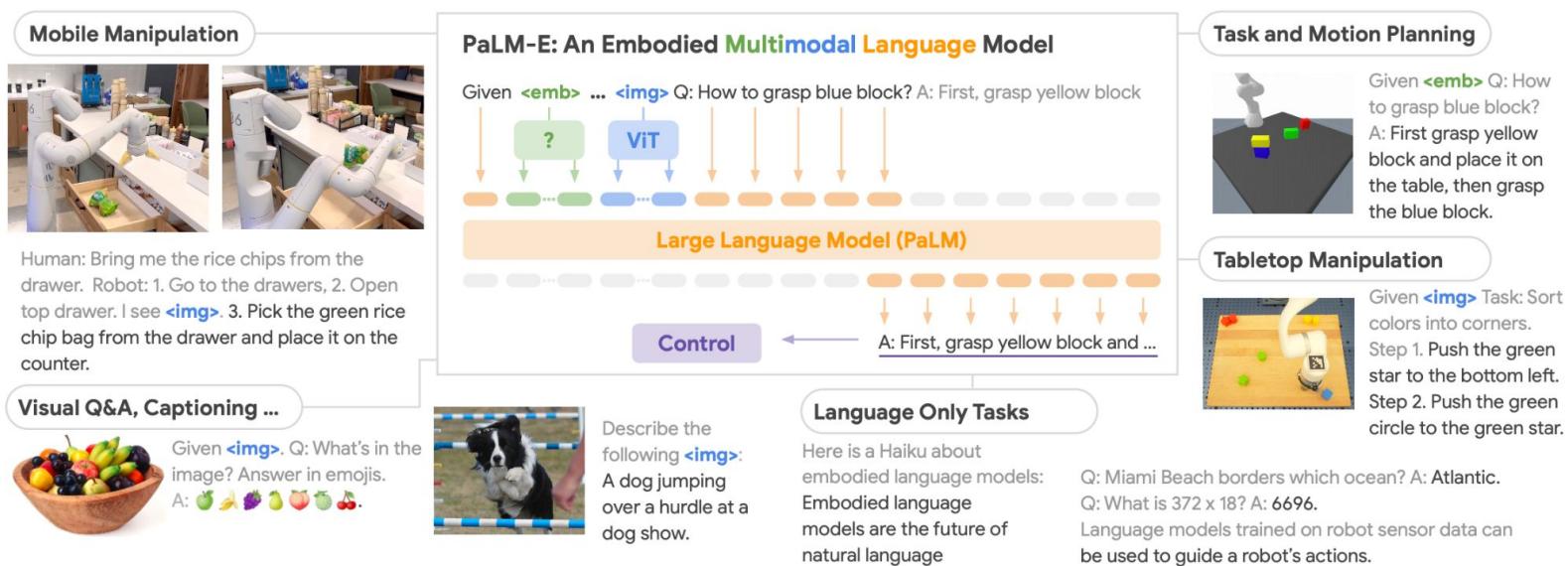


Method

- 1st: Adapt a previously proposed VLM to act as the VLA model
- 2nd: Tokenizing the actions into text tokens and creating “multimodal sentences”
- 3rd: Co-Fine-Tuning, output low-level robot actions + open-vocabulary VQA

Pre-Trained Vision-Language Models (PaLM-E)

- PaLM + Embodied observation = PaLM-E
- 540B LLM + 22B ViT
- High-level planner



Task and Motion Planning



Given <emb> Q: How to grasp blue block?
A: First grasp yellow block and place it on the table, then grasp the blue block.

Tabletop Manipulation



Given Task: Sort colors into corners.
Step 1. Push the green star to the bottom left.
Step 2. Push the green circle to the green star.

Language Only Tasks

Here is a Haiku about embodied language models:
Embodied language models are the future of natural language

Q: Miami Beach borders which ocean? A: Atlantic.
Q: What is 372×18 ? A: 6696.
Language models trained on robot sensor data can be used to guide a robot's actions.

Action Tokenization

- Continuous dimensions are discretized into 256 bins uniformly
- Action space
 - 6-DoF positional and rotational displacement of the robot end-effector
 - the level of extension of the robot gripper
 - a special discrete token for terminating the episode
- “terminate Δpos_x Δpos_y Δpos_z Δrot_x Δrot_y Δrot_z gripper_extension”.
- Overwrite the least frequently used tokens to represent the action vocabulary

Co-Fine-Tuning

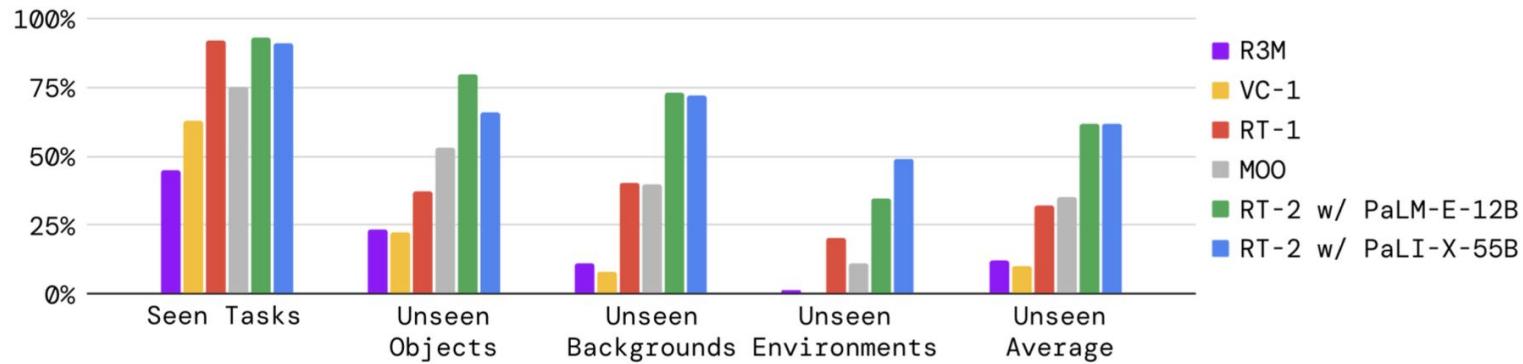
- Datasets:
 - Robotics data (RTI)
 - Web data (PaLI-X, PaLM-E)
- Increasing the sampling weight on the robot dataset
- Output Constraint:
 - Only sampling valid action tokens

Experiments

- I. How does RT-2 perform on seen tasks and more importantly, generalize over new environments?
- II. Can we observe and measure any emergent capabilities of RT-2?
- III. How does the generalization vary with parameter count and other design decisions?
- IV. Can RT-2 exhibit signs of chain-of-thought reasoning similarly to vision-language models?

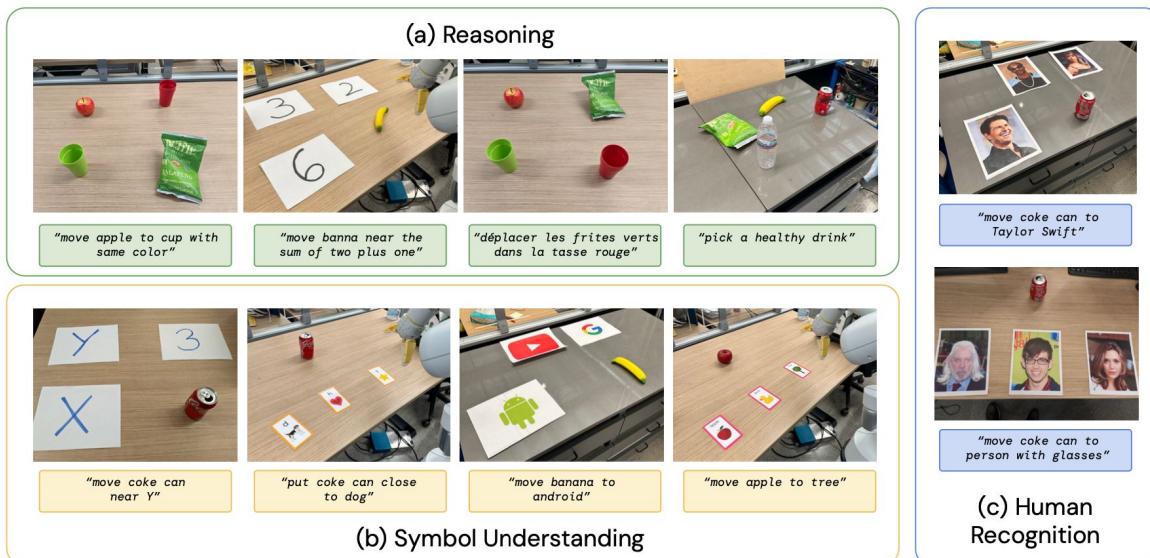
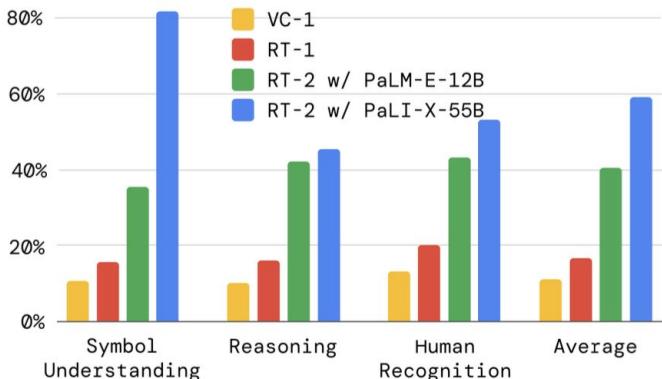
Results - I

- The strength of VLA models lies in transferring more general semantic concepts from the Internet-scale pretraining data

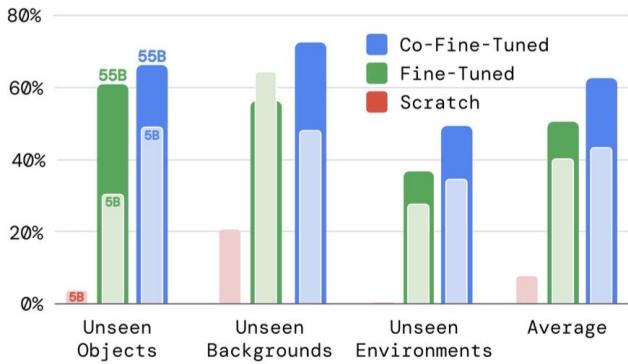


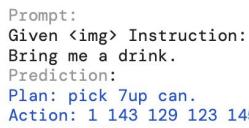
Results - II

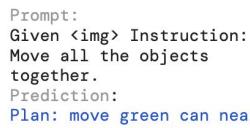
- Emergent: to evaluate the degree to which RT2 can enable new capabilities beyond those demonstrated in the robot data by transferring knowledge from the web

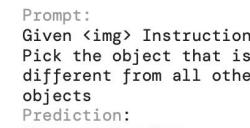


Results – III & IV



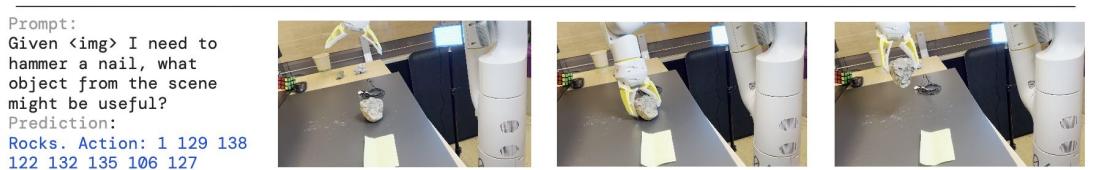
Prompt:
Given  Instruction:
Bring me a drink.
Prediction:
Plan: pick 7up can.
Action: 1 143 129 123 145
114 115 127

Prompt:
Given  Instruction:
Move all the objects together.
Prediction:
Plan: move green can near green rice chip bag.
Action: 1 128 126 127 135
123 119 127

Prompt:
Given  Instruction:
Pick the object that is different from all other objects
Prediction:
Plan: pick rxbar chocolate.
Action: 1 128 129 121 131
123 125 131 125 128 127

Prompt:
Given  Instruction:
Move the green objects together.
Prediction:
Plan: move green can near green rice chip bag.
Action: 1 130 129 121 131
127 128 127

Prompt:
Given  Instruction:
I need to hammer a nail, what object from the scene might be useful?
Prediction:
Rocks. Action: 1 129 138
122 132 135 106 127

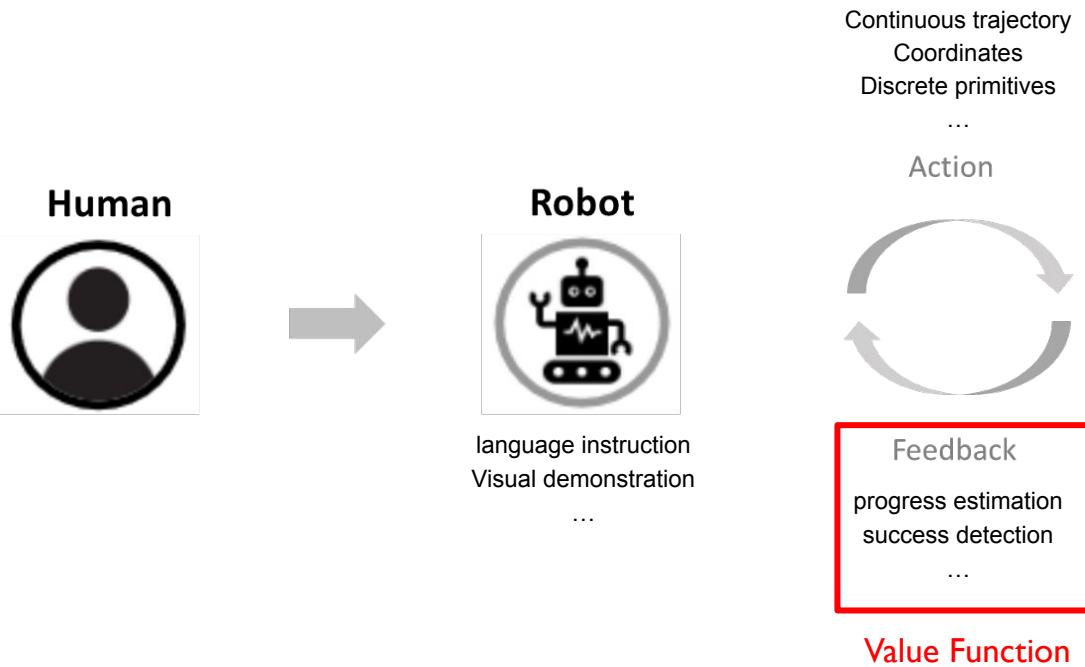
Discussions

- Question1: Compared two-stage framework with the end-to-end VLA model, which one is better?
- Question2: Challenges stemming from the scarcity of action-labeled robotic datasets.

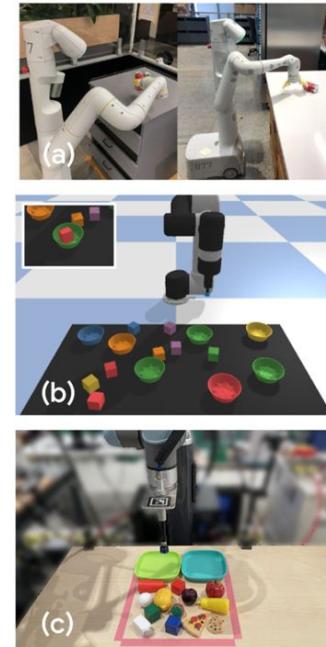
Discussions

- Compared two-stage framework with the end-to-end VLA model, which one is better?
 - Two-stage framework
 - hierarchically decompose the long-horizon tasks into a sequence of sub-goals
 - easy to train planner and controller separately
 - VLA
 - no need to train an additional low-level controller by RL or BC
 - require more training data
- Challenges stemming from the scarcity of action-labeled robotic datasets.
 - Learn general knowledge (world dynamic, temporal reasoning...) from pretraining
 - more pretraining task: video generation...
 - more datasets: human-object interaction videos, simulation, multiview images...

Background



Robot Environments





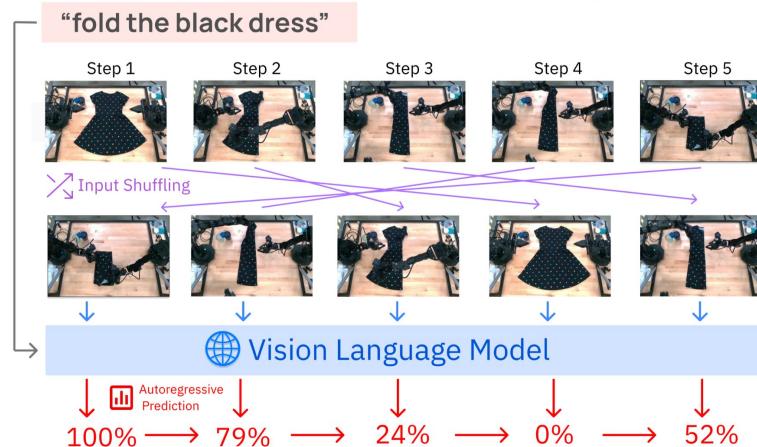
DATA 8005 Advanced Natural Language Processing

Vision Language Models are In-Context Value
Learners

Yi Chen

Fall 2024

- **Generative Value Learning (GVL)** auto-regressively predicts task completion percentage over shuffled frames, enabling impressive in-context value learning.



VLM in-context value learning on
50 OXE datasets and 250 challenging bimanual tasks



Diverse Downstream Applications

Dataset Filtering Success Detection



Bi-Manual Policy Learning

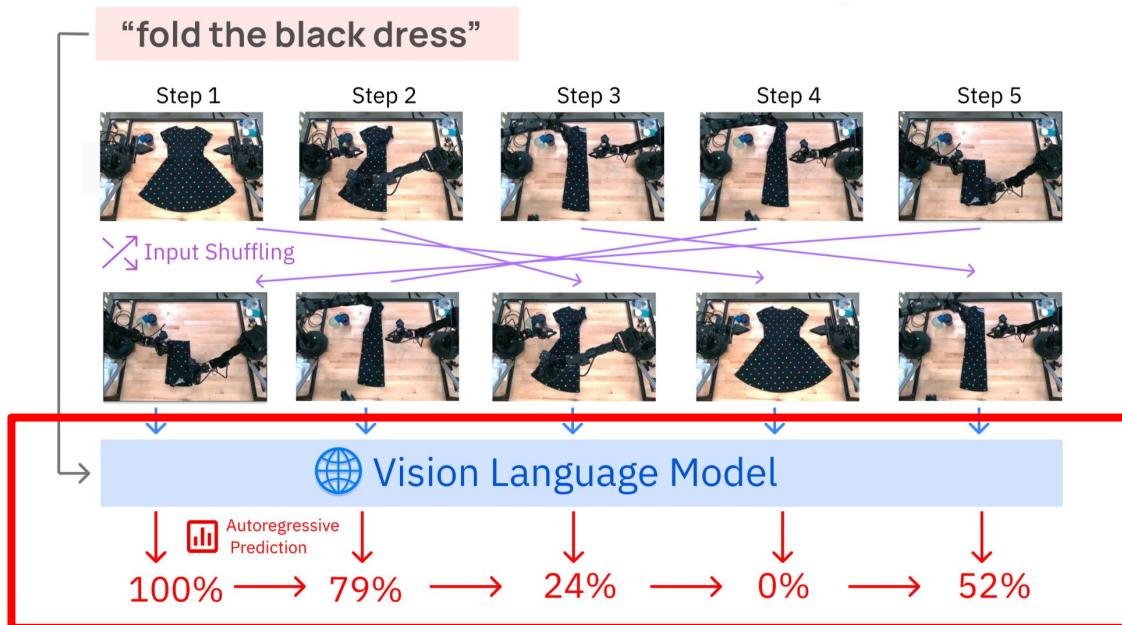
Motivation

- SOTA VLMs have exhibited **strong spatial reasoning and temporal understanding** capabilities, allowing them to generalize to novel scenarios.
- Large transformer-based VLMs have the requisite **context window** to reason over long historical information.
- VLMs commit to their own outputs as inputs for subsequent predictions, imposing **consistency** constraints on long generations.

Methodology

- Autoregressive value prediction → produce globally consistent estimates

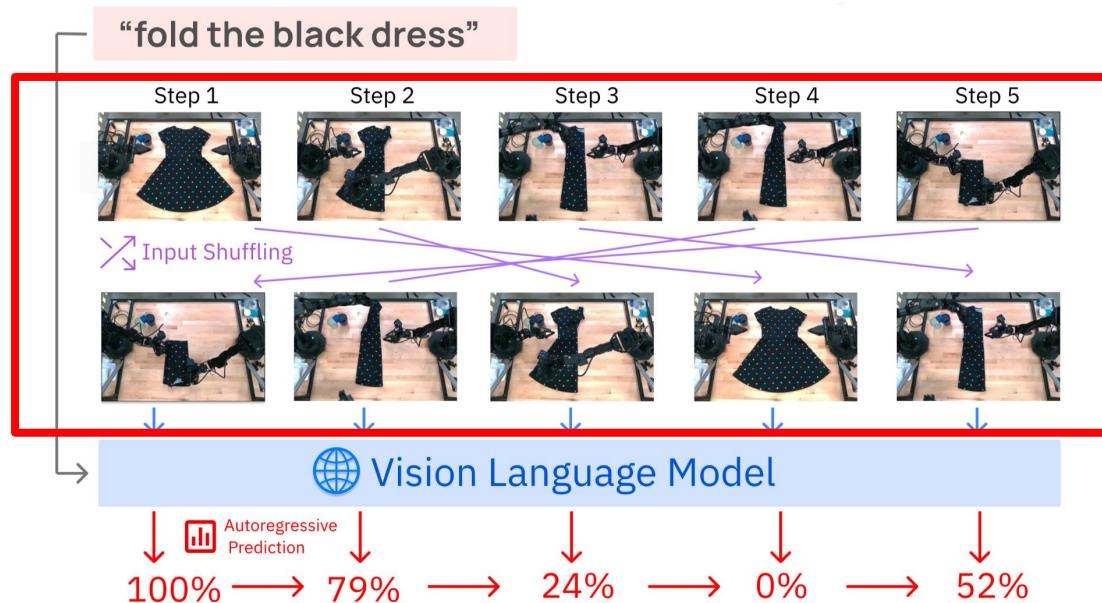
$$v_t = \text{VLM}(o_1, \dots, o_T; v_1, \dots, v_{t-1}; l_{\text{task}}), \forall t \in [2, T]$$



Methodology

- Input observation shuffling → avoid the short-cut solution of outputting monotonically increasing values

$$v_{\tilde{1}}, \dots, v_{\tilde{T}} = \text{VLM}(o_{\tilde{1}}, \dots, o_{\tilde{T}}; l_{\text{task}}, o_1), \quad \text{where} \quad (\tilde{1}, \dots, \tilde{T}) = \text{permute}(1, \dots, T)$$



Methodology

- The full prompt provided to Gemini-1.5-Pro for GVL predictions

You are an expert roboticist tasked to predict task completion percentages for frames of a robot for the task of {task_description}.

The task completion percentages are between 0 and 100, where 100 corresponds to full task completion. We provide several examples of the robot performing the task at various stages and their corresponding task completion percentages. Note that these frames are in random order, so please pay attention to the individual frames when reasoning about task completion percentage.

Initial robot scene: [IMG]

In the initial robot scene, the task completion percentage is 0.

Now, for the task of {task_description}, output the task completion percentage for the following frames that are presented in random order. For each frame, format your response as follow: Frame {i}:
Frame Description: {}, Task Completion Percentages:{}%

Frame 1: [IMG]

...

Frame n: [IMG]

Methodology

- In-context value learning → improve value accuracy with in-context examples

$$v_{\tilde{1}}, \dots, v_{\tilde{T}} = \text{VLM} (o_{\tilde{1}}, \dots, o_{\tilde{T}}, l_{\text{task}} \mid \text{permute} ((o_1, v_1), (o_2, v_2), \dots, (o_M, v_M)))$$



(a) In-context examples from human videos



(b) Target prediction robot videos

- Question 1: Can we use other approaches to improve VLMs for better value prediction?

Experimental Questions

1. Can GVL produce zero-shot value predictions for a broad range of tasks and embodiments?
2. Can GVL improve from in-context learning?
3. Can GVL be used for other downstream robot learning applications?

Evaluation Metric

- **Value-Order Correlation (VOC)** computes the rank correlation between the predicted values and the chronological order of the input expert video:

$$\text{VOC} = \text{rank-correlation}(\text{argsort}(v_{\tilde{1}}, \dots, v_{\tilde{T}}); \text{arange}(T))$$

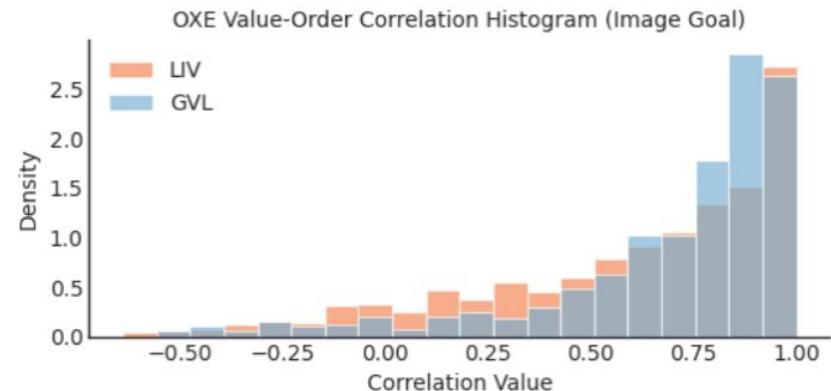
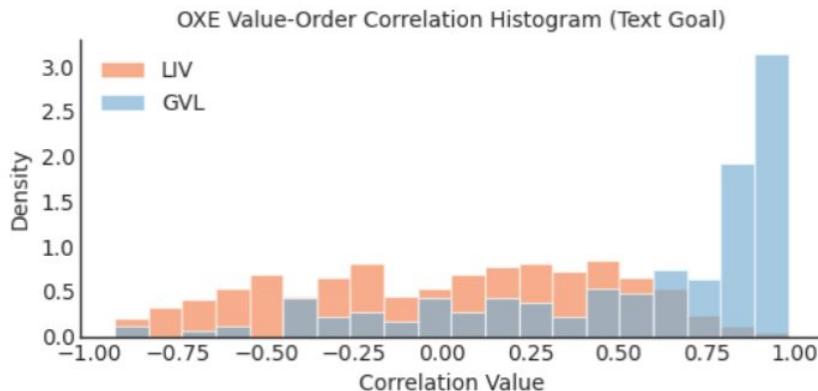
VOC ranges from -1 to 1 , with 1 indicates perfect alignment between two orderings.

Evaluation Metric

- Expert quality demonstrations, by construction, have values that monotonically increase with time. → high VOC scores
- Low-quality trajectories should often contain high repetition of visually similar frames due to the presence of redundant, re-attempt actions or poorly-placed cameras. → low VOC scores
- Question 2: What are the potential drawbacks of VOC?

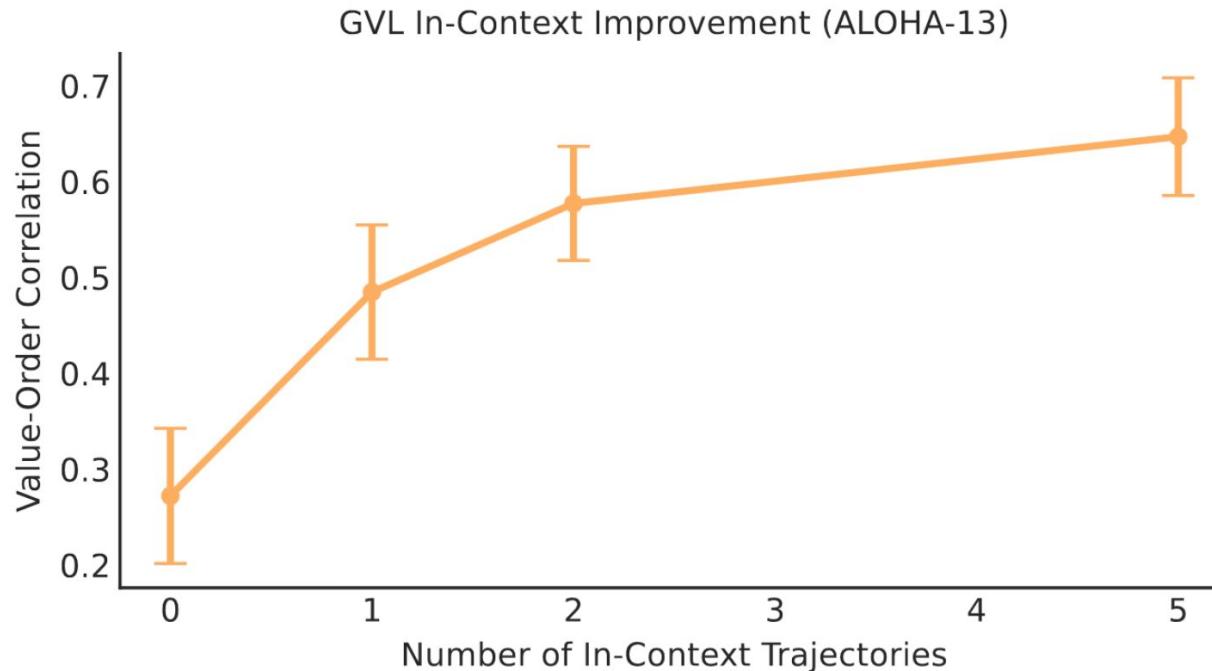
Large-scale real-world evaluation (Q1)

- GVL generates VOC scores that heavily skew to the right, indicating that it is able to zero-shot recover the temporal structure hidden in shuffled demonstration videos.



Multi-Modal In-Context Value Learning (Q2)

- **Few-shot in-context learning.** GVL is able to utilize its full context and exhibit strong generalization with up to 5 in-context trajectories.



Multi-Modal In-Context Value Learning (Q2)

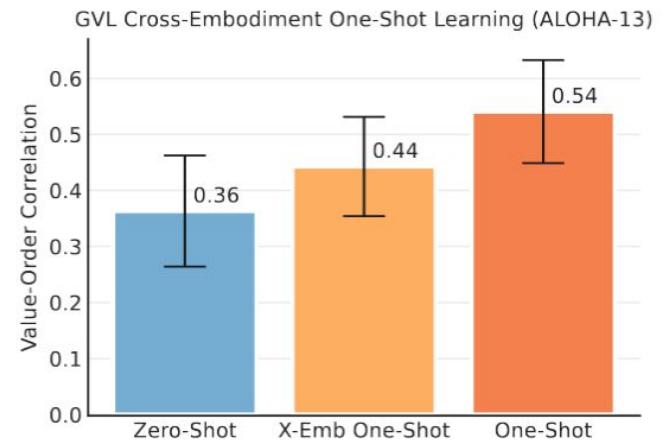
- **Cross-embodiment in-context learning.** GVL's value predictions can be improved by examples from human videos.



(a) In-context examples from human videos

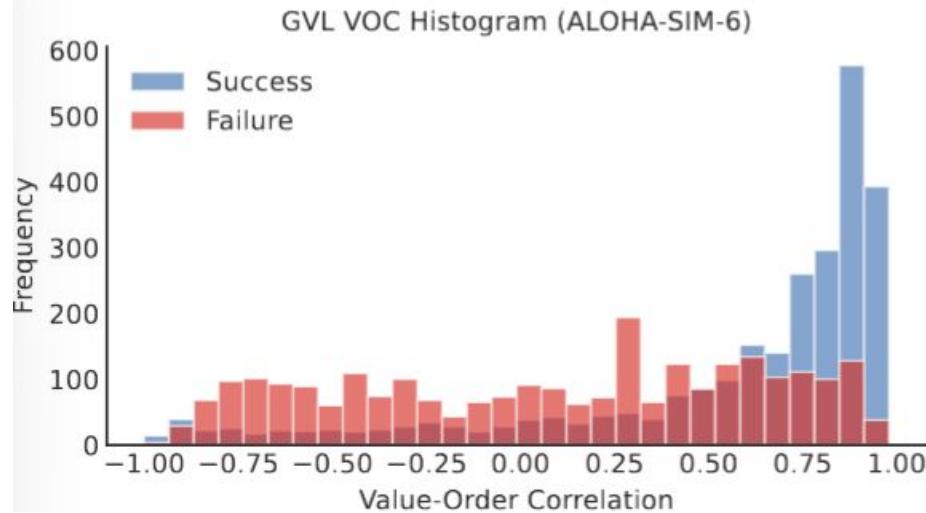


(b) Target prediction robot videos



GVL Applications (Q3)

- Success detection.



Method	Accuracy	Precision	Recall
GVL-SD (Zero-Shot)	0.71	0.71	0.71
GVL-SD (One-Shot)	0.75	0.85	0.70
SuccessVQA [15]	0.62	0.33	0.73
SuccessVQA-CoT	0.63	0.44	0.68

GVL Applications (Q3)

- **Advantage-weighted regression (AWR) for real-world visuomotor control.**

$$\mathcal{L}(\theta) := -\mathbb{E} [\exp (\tau \cdot (v_{k+1} - v_k)) \cdot \log \pi_\theta(a_k \mid o_k)]$$

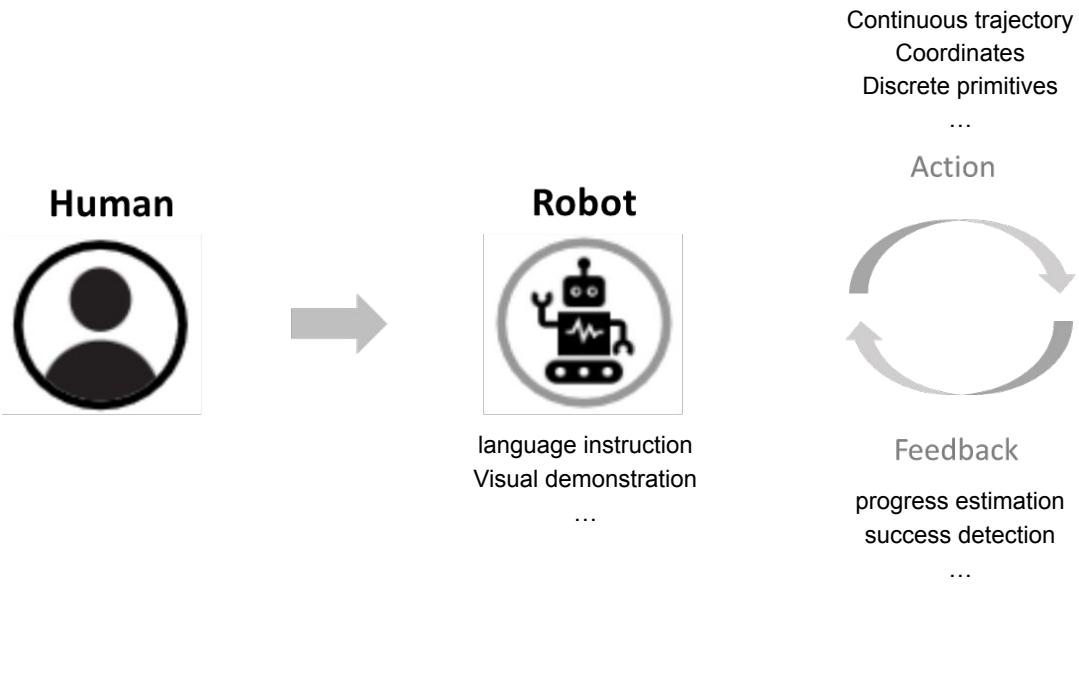
- AWR with GVL outperforms IL baselines when the predicted values have high VOCs.

Real-World ALOHA Tasks	GVL + DP	DP	Avg. VOC
bowl-in-rack	7/10	6/10	0.57
banana-handover	7/10	5/10	0.73
close-laptop	9/10	6.5/10	0.59
open-drawer	4/10	6/10	0.09
remove-gears	4.67/10	7/10	0.19
pen-handover	1.5/10	0/10	0.43
fold-dress	7/10	7/10	0.66

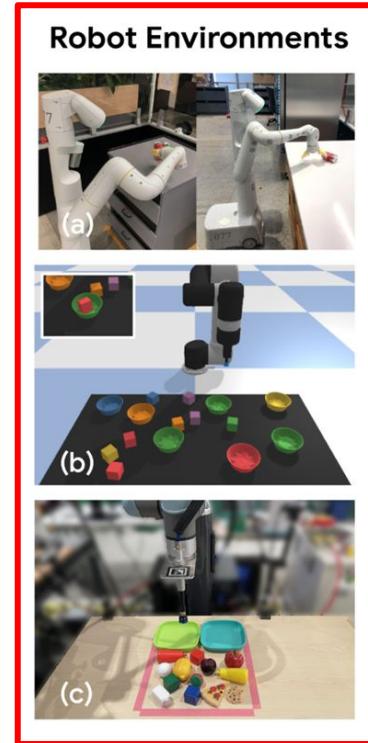
Discussions

- Question 1: Can we fine-tune VLMs to perform better value predictions?
- Question 2: Periodic tasks such as wiping or stirring may be hard to discern with Value-Order Correlation. Can we design a better evaluation metric?

Background



World Model (Simulator)





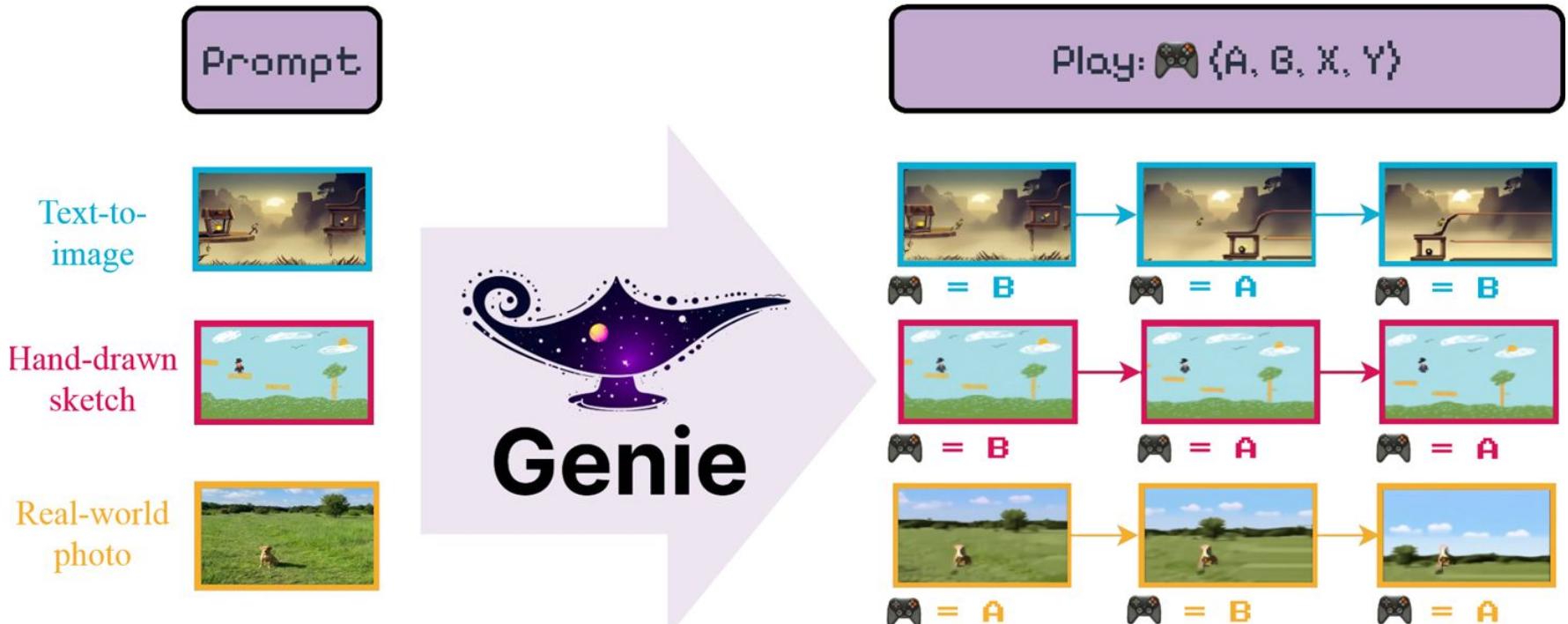
DATA 8005 Advanced Natural Language Processing

Genie: Generative Interactive Environments

Yi Chen

Fall 2024

- Genie is the first generative **interactive** environment trained in an unsupervised manner from **unlabelled** Internet videos.

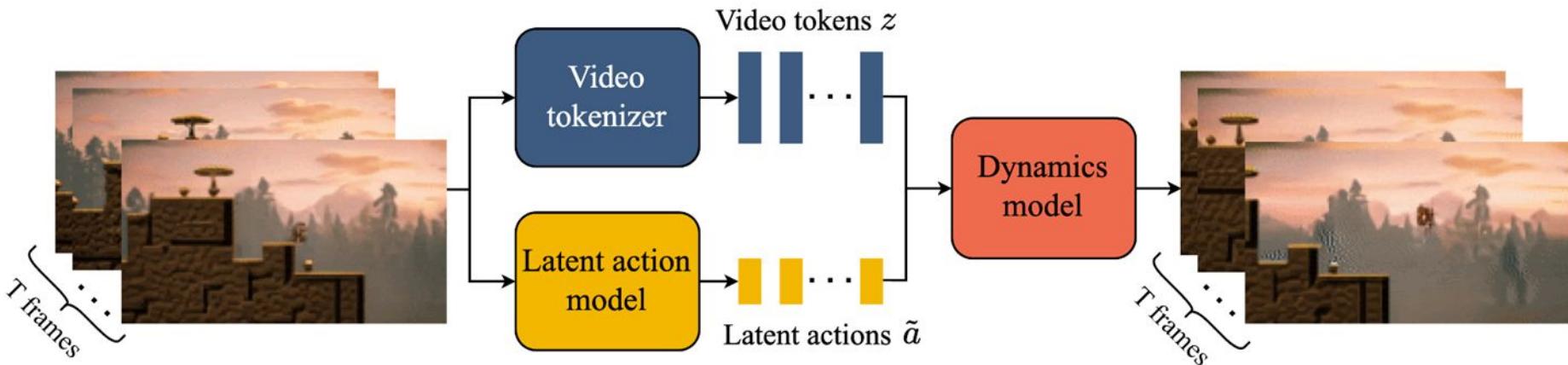


Motivation

- There remains a gulf between the level of **interactions and engagement of video generative models** and language tools such as ChatGPT.
- Given a large corpus of videos from the Internet, we could not only train models capable of generating novel images or videos, but **entire interactive experiences**.
- This goal is achievable with **latent actions**, which can be learned from unlabelled Internet videos in an **unsupervised** manner.

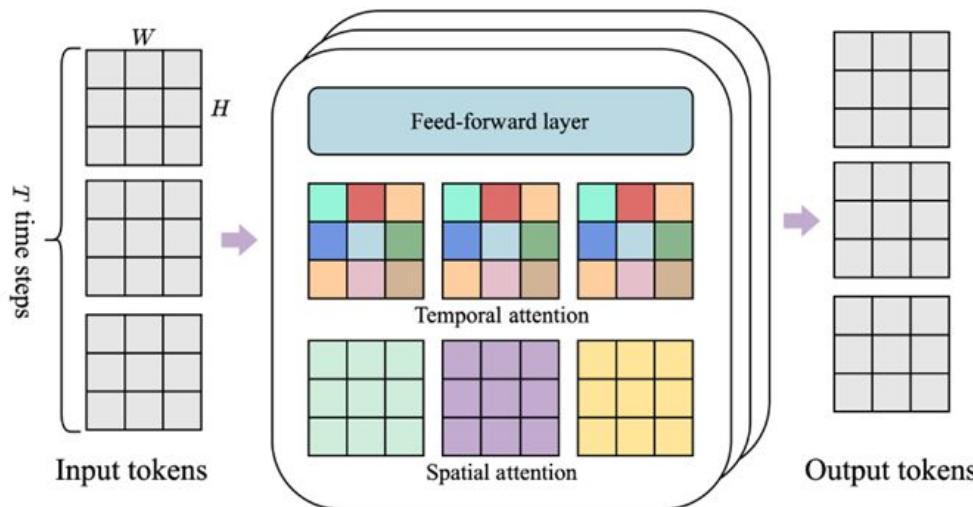
Methodology

- **Model Components.**



Methodology

- **ST-transformer architecture.**



Methodology

- **Video Tokenizer.**

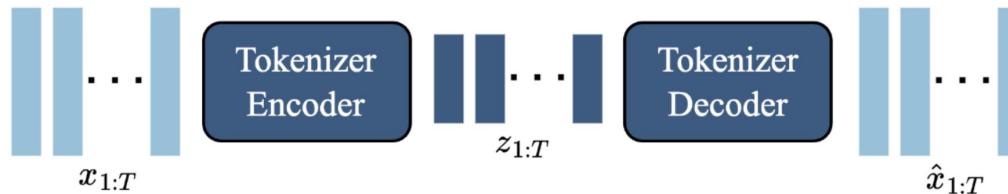


Figure 6 | **Video tokenizer:** a VQ-VAE with ST-transformer.

Methodology

- Latnet Action Model and Dynamics Model.

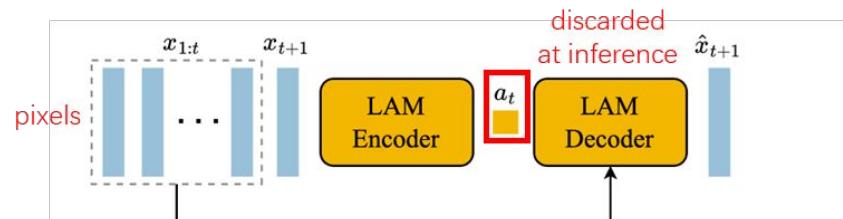


Figure 5 | Latent action model: learns actions a_t unsupervised from unlabelled video frames.

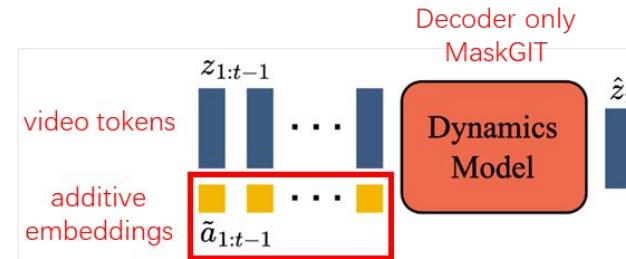


Figure 7 | Dynamics model: takes in video tokens and action embeddings, and predicts future masked video tokens.

Joint training

Methodology

- **Genie Inference.**

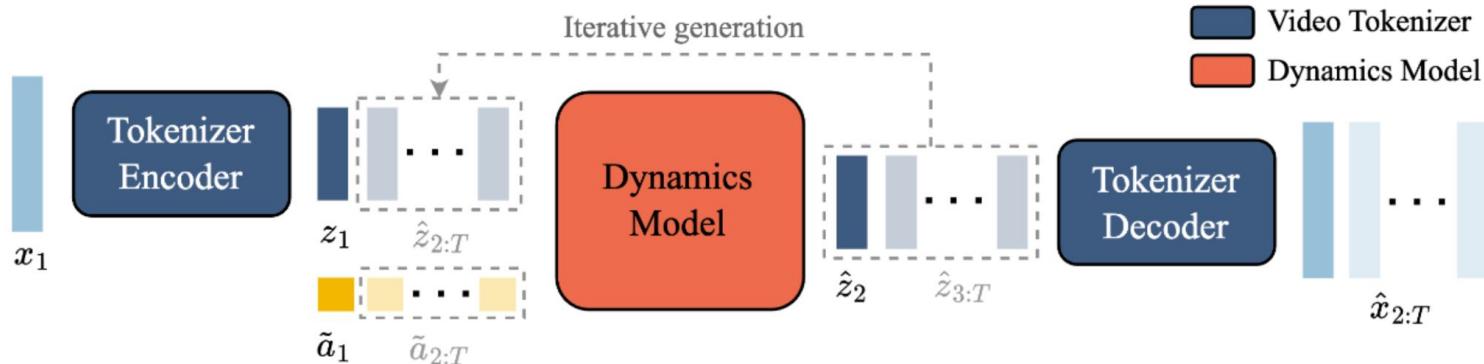


Figure 8 | **Genie Inference:** the prompt frame is tokenized, combined with the latent action taken by the user, and passed to the dynamics model for iterative generation. The predicted frame tokens are then decoded back to image space via the tokenizer’s decoder.

Training Datasets

- “**Platforms**” Dataset: We construct the Platformers dataset by filtering publicly available videos for keywords relating to platformers, yielding 55M 16s video clips at 10FPS, with *160x90 resolution*. The final dataset contains *6.8M 16s video clips (30k hours)*, within an order of magnitude of other popular Internet video datasets.
- “**Robotics**” Dataset: We also consider the robotics datasets used to train RTI (Brohan et al., 2023), combining their dataset of *~130k robot demonstrations* with a *separate dataset of simulation data* and the *209k episodes of real robot data* from prior work (Kalashnikov et al., 2018). Note that we do not use actions from any of these datasets, and simply treat them as videos.

Experiments

- Question 1: The advantages and future applications of Latent Actions?

Experiments

- Qualitative Results of Platformers-trained model



Figure 10 | Playing from Image Prompts: We can prompt Genie with images generated by text-to-image models, hand-drawn sketches or real-world photos. In each case we show the prompt frame and a second frame after taking one of the latent actions four consecutive times. In each case we see clear character movement, despite some of the images being visually distinct from the dataset.

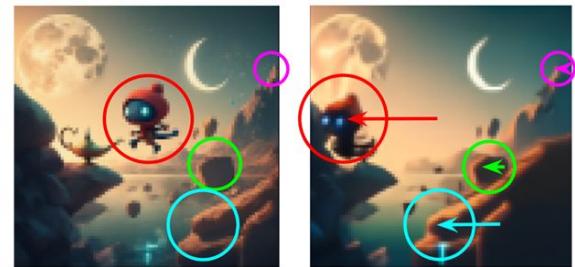


Figure 12 | Emulating parallax, a common feature in platformer games. From this initial text-generated image, the **foreground** moves more than the **near** and **far** middle ground, while the **background** moves only slightly.

Experiments

- Qualitative Results of Robotics-trained model

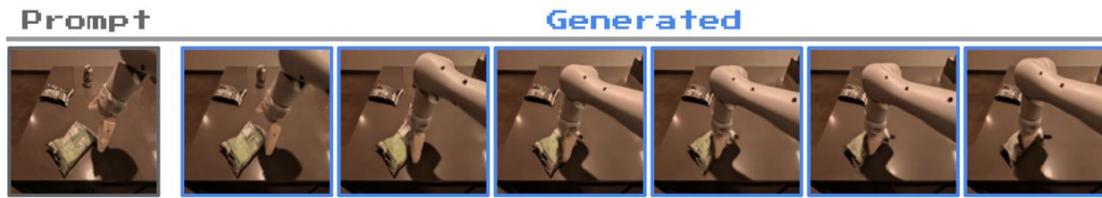


Figure 11 | Learning to simulate deformable objects: we show frames from a ten step trajectory in the model, taking the same action. Genie is capable of learning the physical properties of objects such as bags of chips.

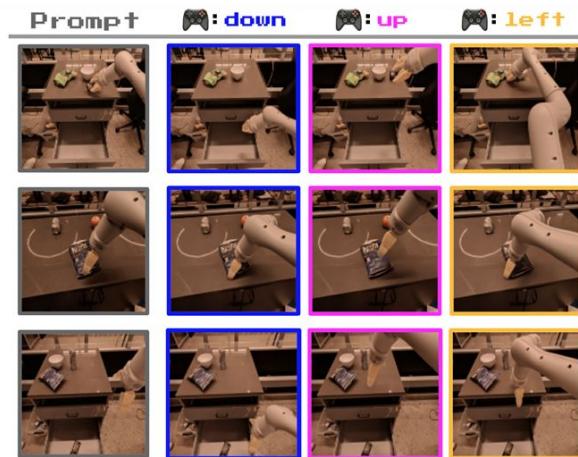


Figure 13 | Controllable, consistent latent actions in Robotics: trajectories beginning from three different starting frames from our Robotics dataset. Each column shows the resulting frame from taking the same latent action five times. Despite training without action labels, the same actions are consistent across varied prompt frames and have semantic meaning: *down*, *up* and *left*.

Experiments

- **Training Agents**

- **Objective**
 - If latent actions learnt from Internet videos can be used for **imitating behaviors from unseen videos?**
- **Details**
 - Use a frozen LAM to **label** expert videos with discrete latent actions.
 - Train a policy that **predicts** the likelihood of a latent action given an observation.
 - Use a small dataset with ground-truth actions to fill in a **mapping dictionary**.



Figure 14 | Playing from RL environments: Genie can generate diverse trajectories given an image of an unseen RL environment.

Experiments

- **Training Agents**
 - **Upper bound:**
 - an oracle behavioral cloning model that has access to expert actions.
 - **Lower bound:**
 - a random agent.
- **Question 2: In addition to gaming agents, how can we use latent actions to train a generalist robot policy?**

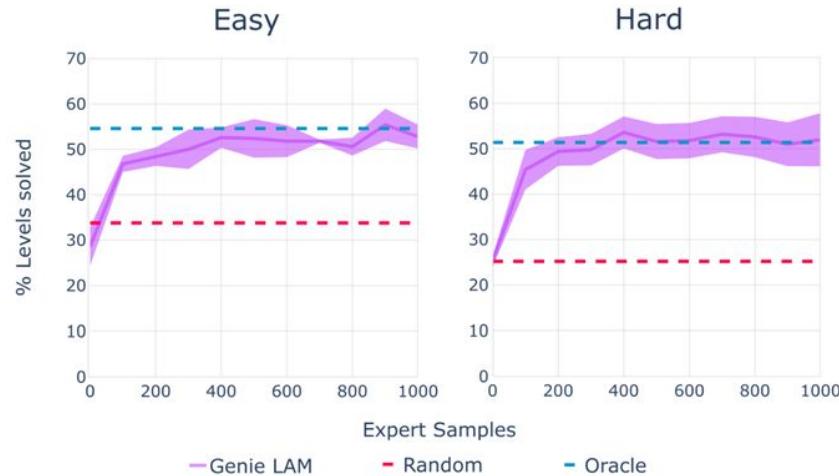


Figure 15 | BC results. Mean percentage of levels solved out of 100 samples, averaged over 5 seeds with 95% confidence intervals.

Future Direction

- Genie could be trained from *an even larger proportion of Internet videos* to simulate diverse, realistic, and imagined environments.
- Given that the lack of rich and diverse environments is one of the key limitations in RL, Genie could unlock new paths to *creating more generally capable agents*.

Discussions

- Question 1: Advantages and future applications of Latent Actions?
 - **Simulator**
 - Unified control signal to interact with the simulator.
 - Leverage large-scale unlabelled data for self-supervised training.
 - **Policy**
 - Enable training of a generalizable policy model and possible cross-embodiment transfer.
 - Reduce spatial-temporal redundancy -> faster generation speed, longer generation sequence.
 - **Reward Model**
 - Measure the rationality of a long trajectory video with a compact sequence.

Discussions

- Question 2: In addition to gaming agents, how can we use latent actions to train a generalist robot policy?
 - Robot action space has a higher degree of freedom.
 - Mapping between latent actions and real robot actions is not trivial.