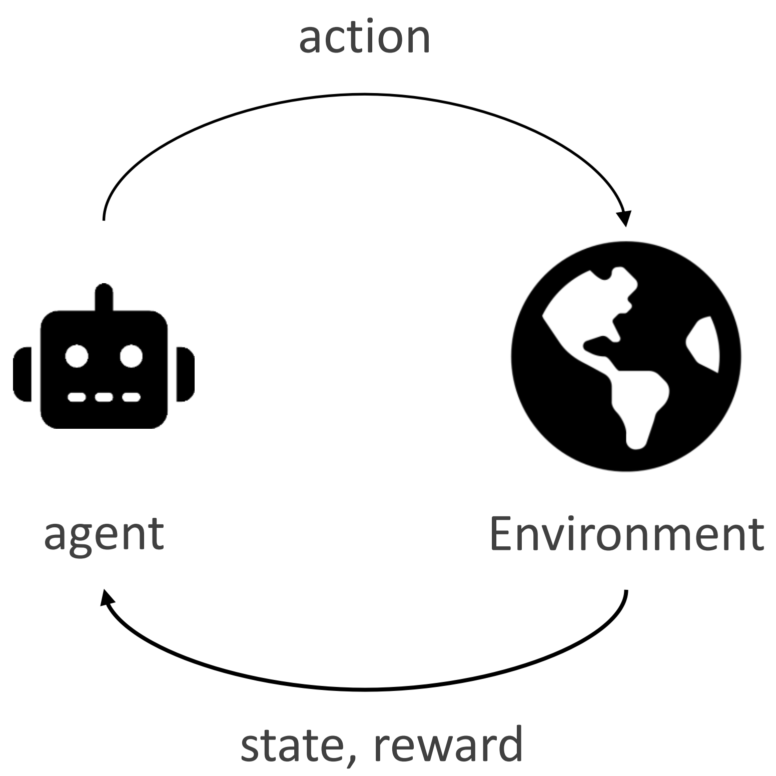
VICtoR: Learning Hierarchical Vision-Instruction Correlation Rewards for Long-horizon Manipulation

Reinforcement Learning, Reward Learning, Modality Alignment

Kuo-Han Hung* Pang-Chi Lo* Jia-Fong Yeh* Han-Yuan Hsu
Yi-Ting Chen Winston H. Hsu

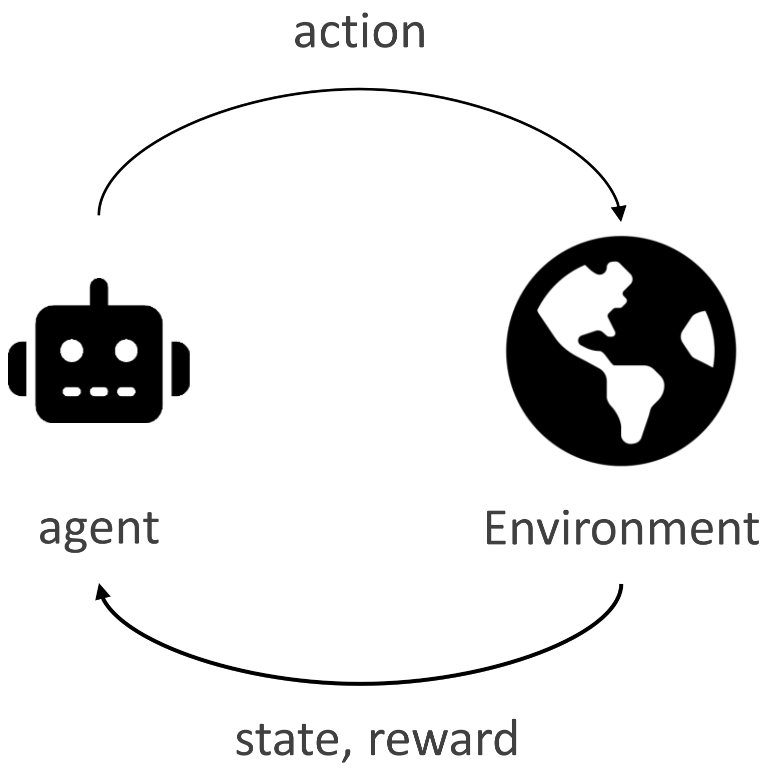
Reinforcement Learning (RL)

- **RL** learns policies by **interacting** with the environment and adjusting based on **feedback** (rewards)



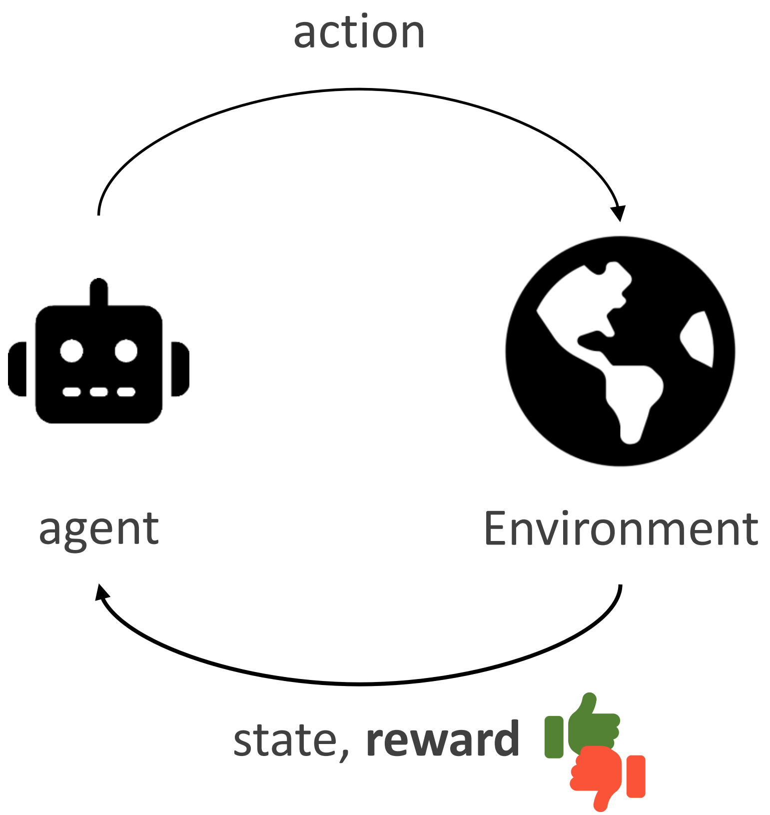
Reinforcement Learning (RL)

- **RL** learns policies by **interacting** with the environment and adjusting based on **feedback** (rewards)
- It has proven to be an **effective** framework for various **downstream tasks**, including *LLM post-training*, *gaming*, and *robotics*



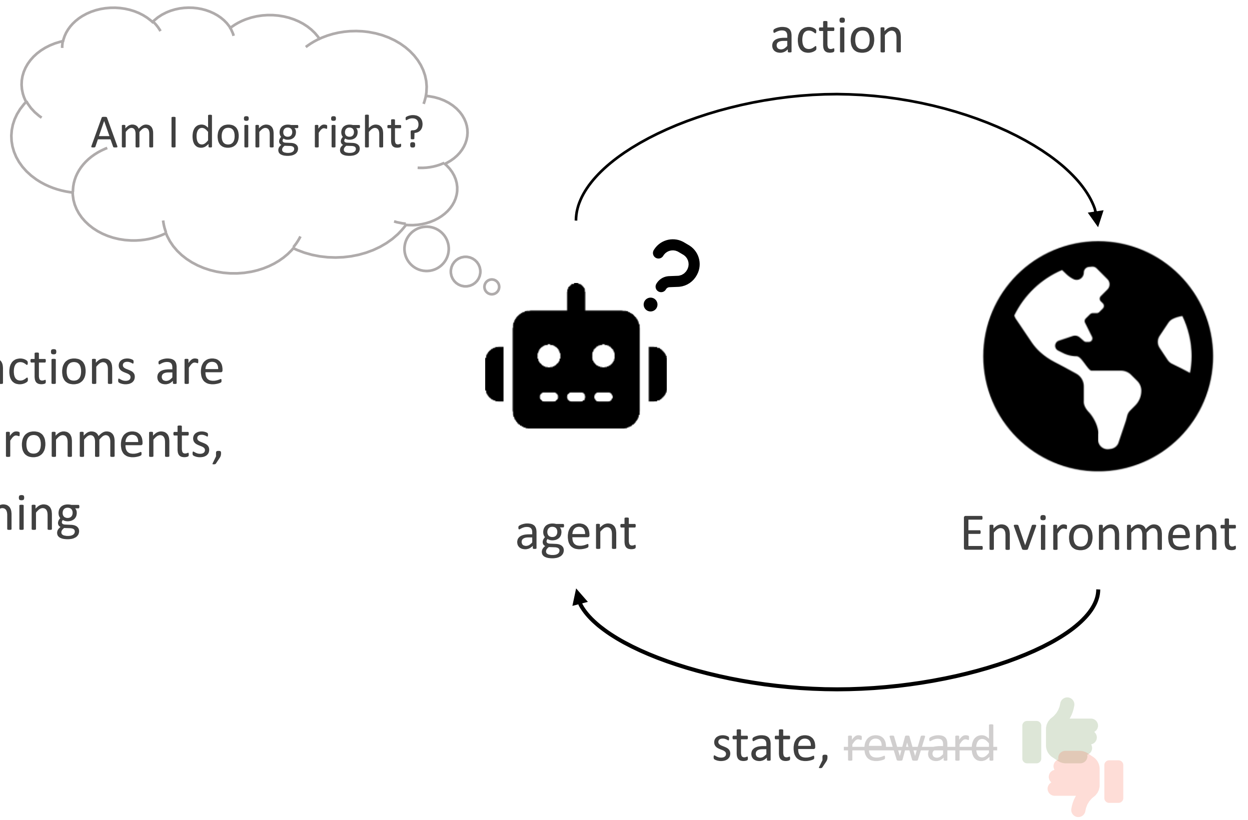
Rewards is Important to RL

- A **reward signal** should provide **timely** and **informative** feedback on whether the action taken in the current state **contributes** to task completion



Rewards in Real-world Environments

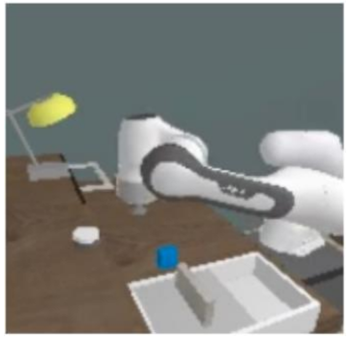
- Unlike in simulation, reward functions are often **lacking** in **real-world** environments, leading to **ineffective** policy learning



Vision-Instruction Correlation (VIC) Reward

- **VIC reward**: Learning reward signals by capturing the **correlation** between the current visual observation and task instructions
- A more **efficient** approach to reward learning in practical scenarios

"Move the block into the closed drawer"



Task Instruction **Visual Observation**

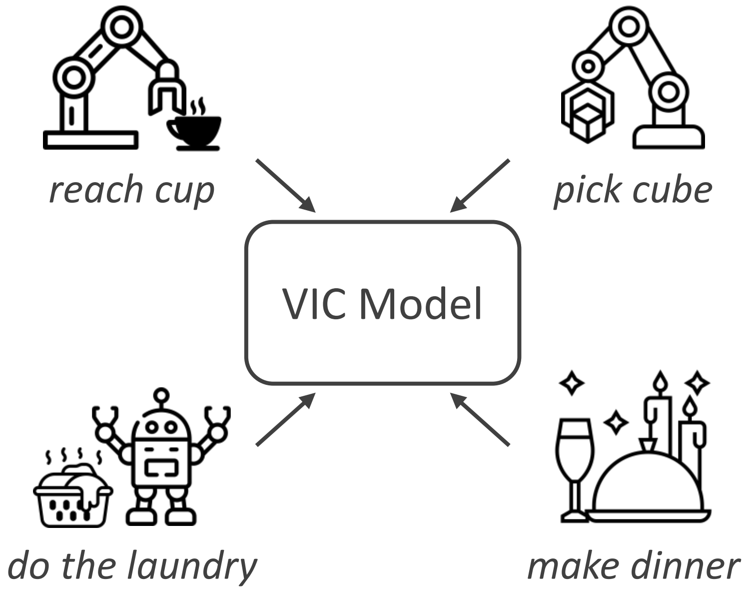
Correlated or **Uncorrelated**?

Challenges in Learning VIC Rewards

- Existing methods **fail to learn** VIC rewards for **long-horizon tasks** because they:

Challenges in Learning VIC Rewards

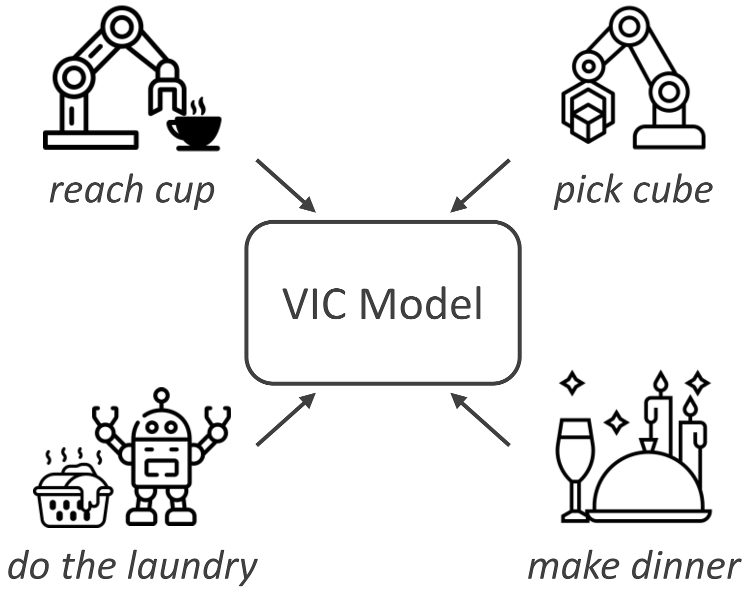
- Existing methods **fail to learn** VIC rewards for **long-horizon tasks** because they:



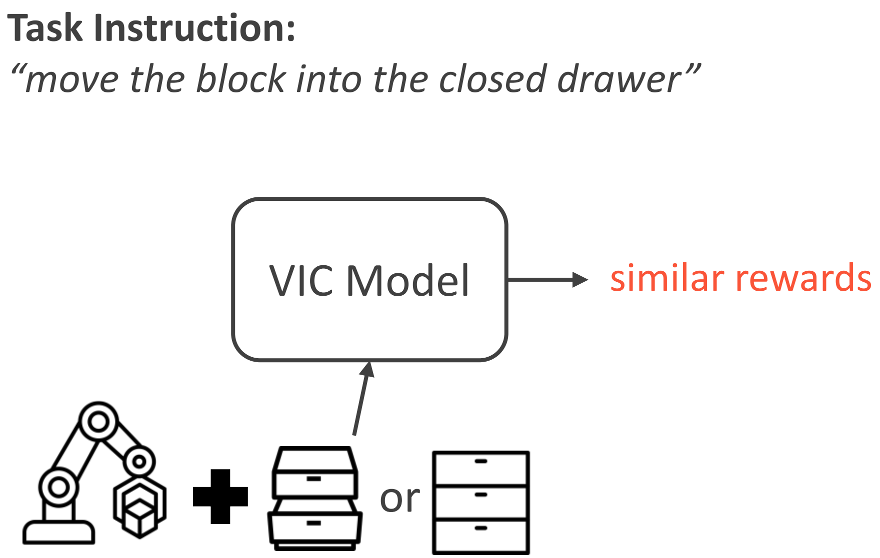
Use a single model to learn rewards for tasks
of **varying difficulty and horizon**

Challenges in Learning VIC Rewards

- Existing methods **fail to learn** VIC rewards for **long-horizon tasks** because they:



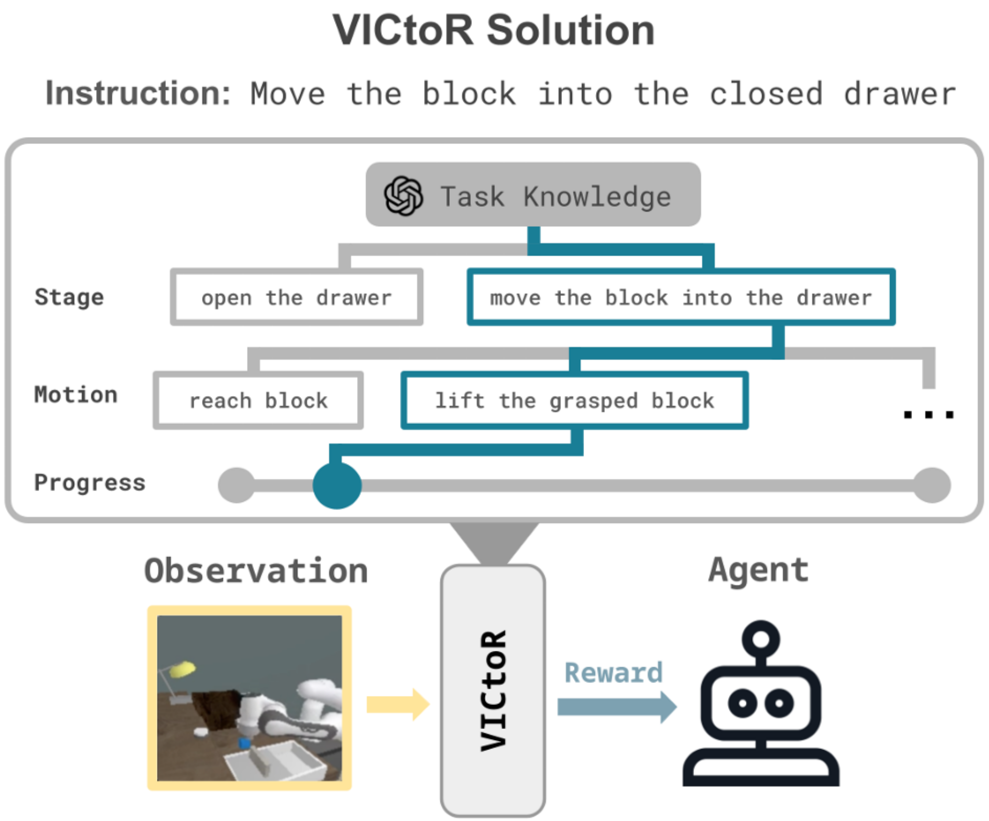
Use a single model to learn rewards for tasks of **varying difficulty and horizon**



Overlook object state details in the environment

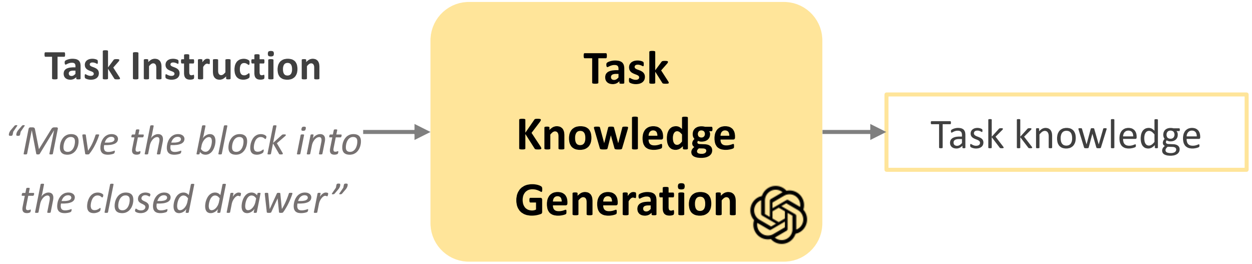
Introduce VICtoR

Motivation



- **Decomposing** long-horizon task into manageable segments
- Evaluating the task progress at three **different granularities**
- Tracking the **changes of object status** in the environment

VICtoR Components



Task Knowledge Generation

Long-horizon Task

"Move the block into the closed drawer"

Task

LLM (GPT-4)

Stage

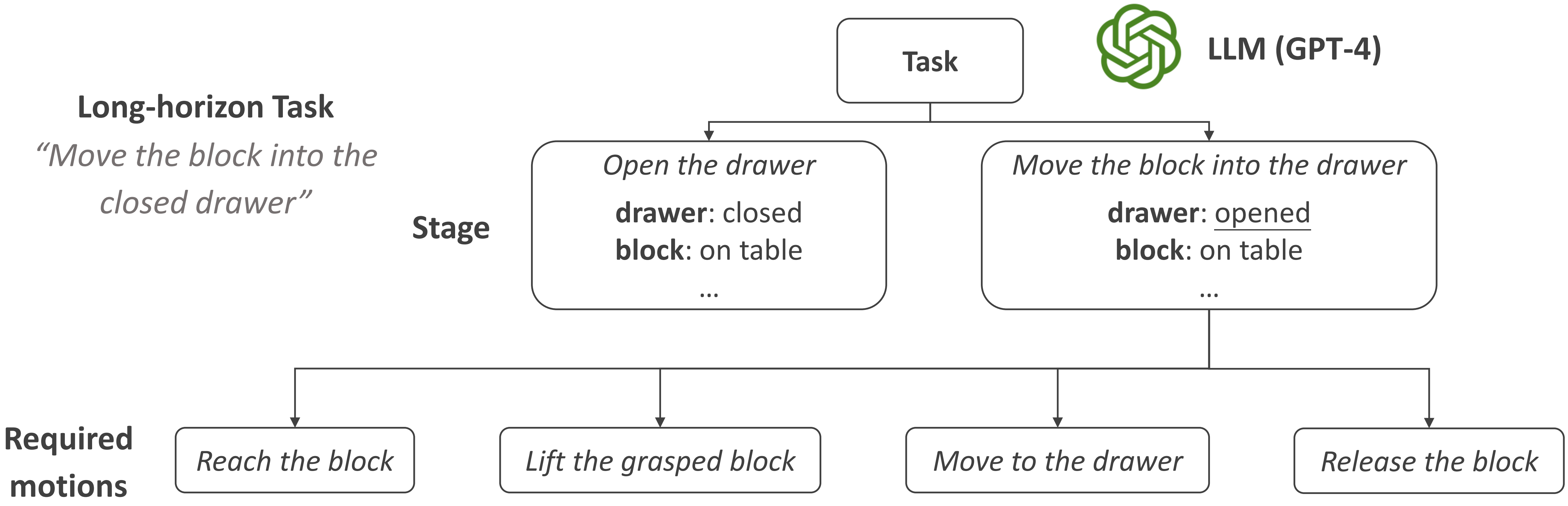
Open the drawer

drawer: closed
block: on table
…

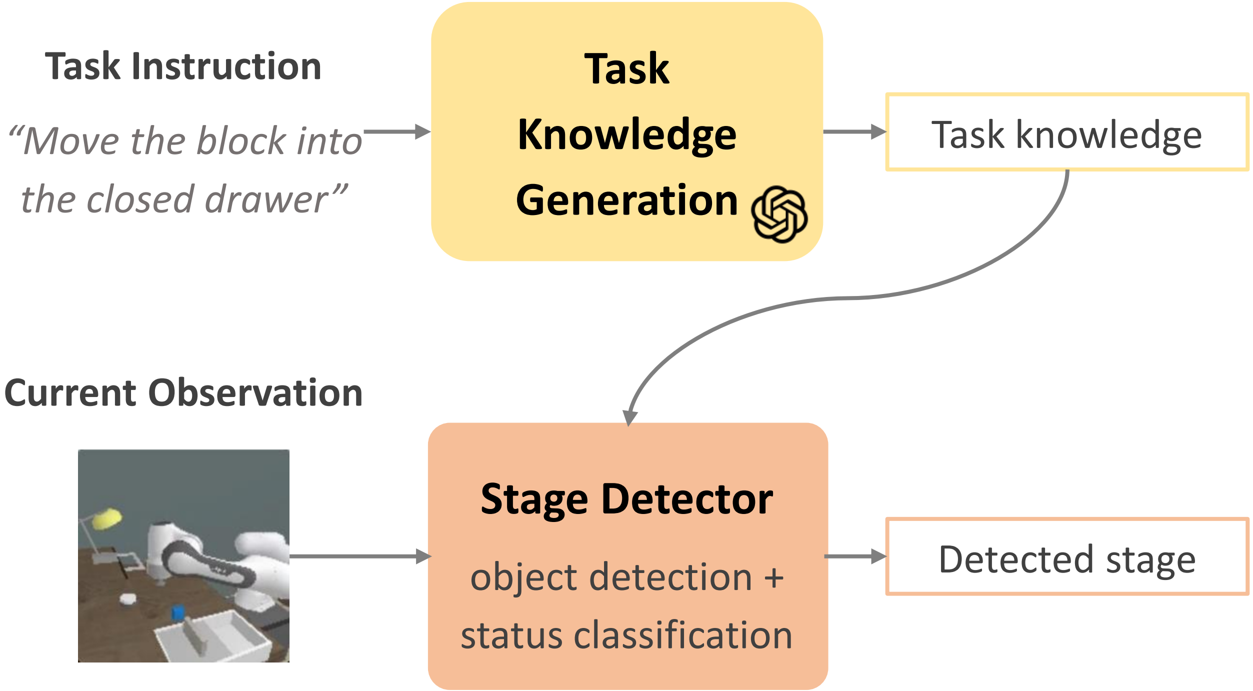
Move the block into the drawer

drawer: opened
block: on table
…

Task Knowledge Generation

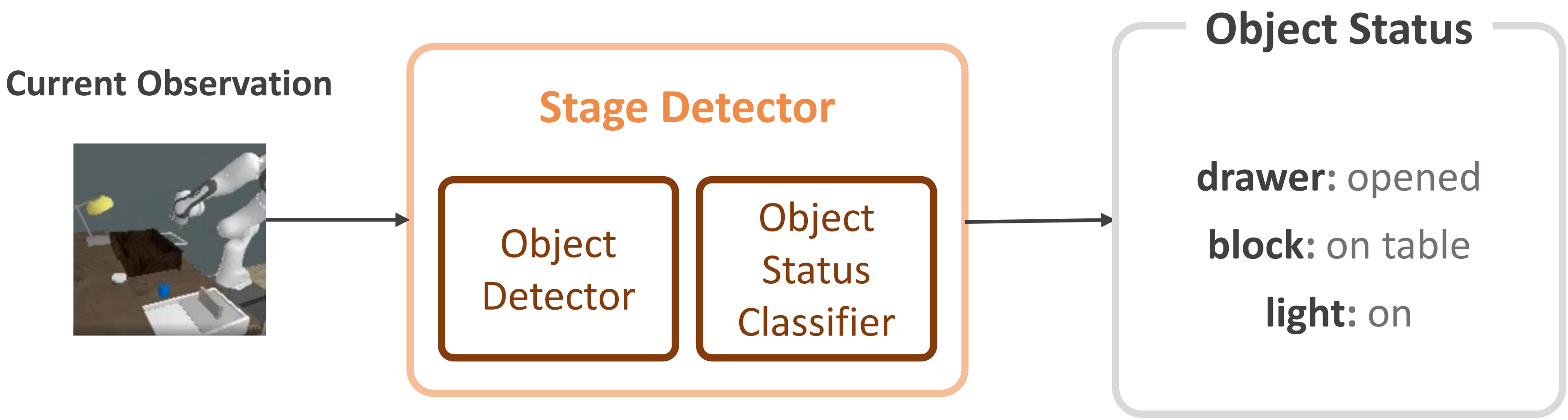


VICtoR Components



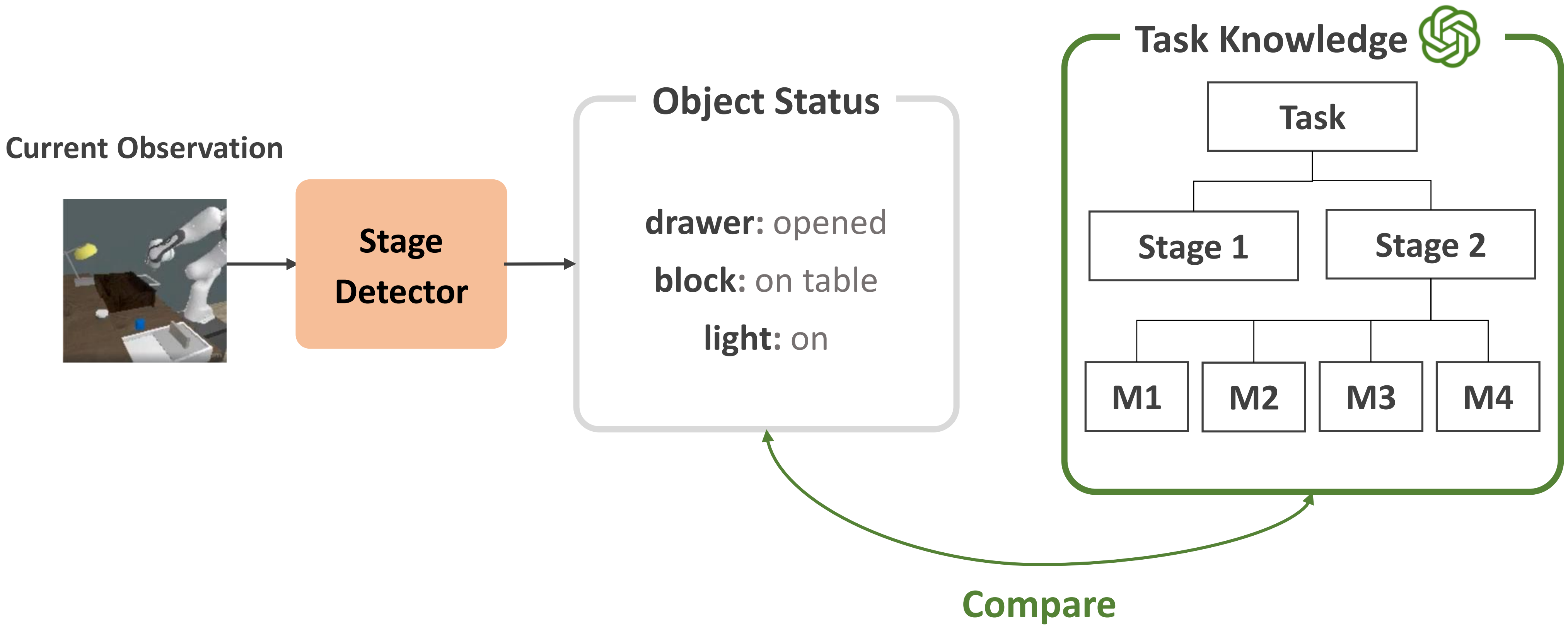
Stage Determination

- Estimating objects status by an open-vocabulary object detector and an object status classifier



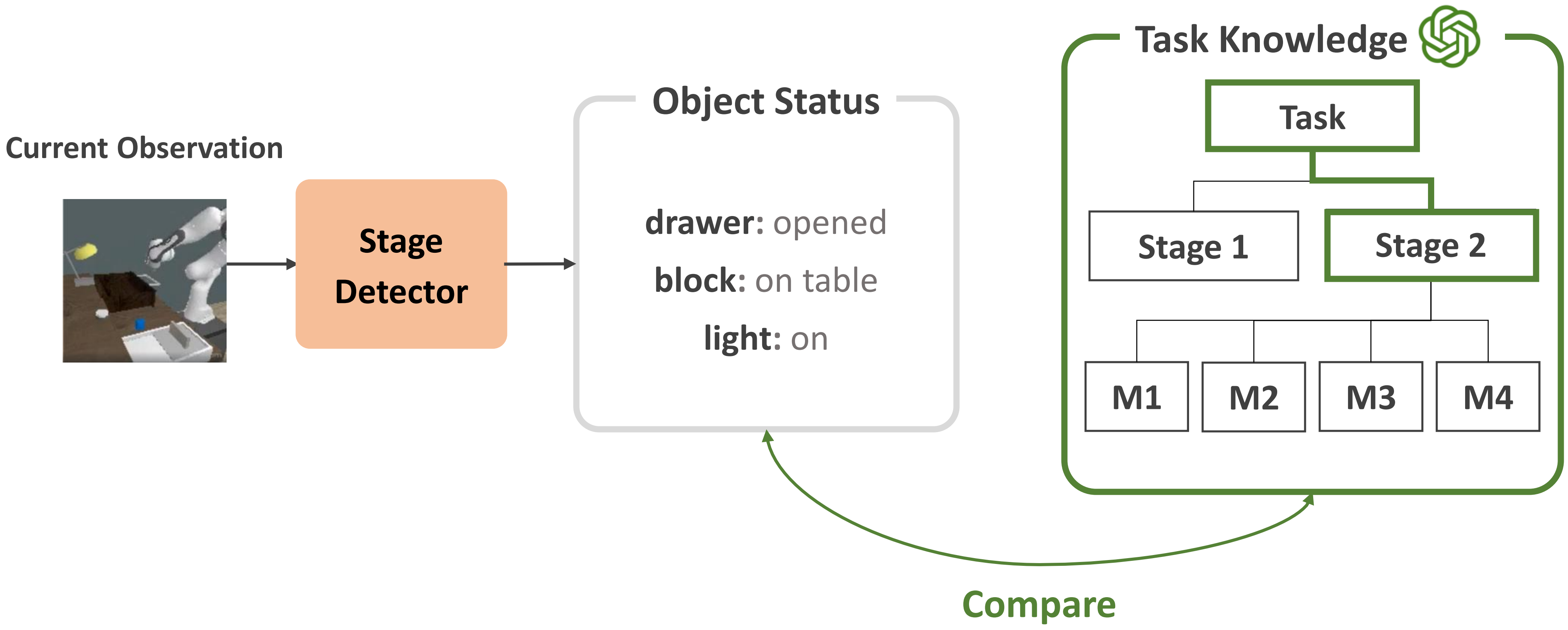
Stage Determination

- Compare the estimated status with the ideal ones in the task knowledge

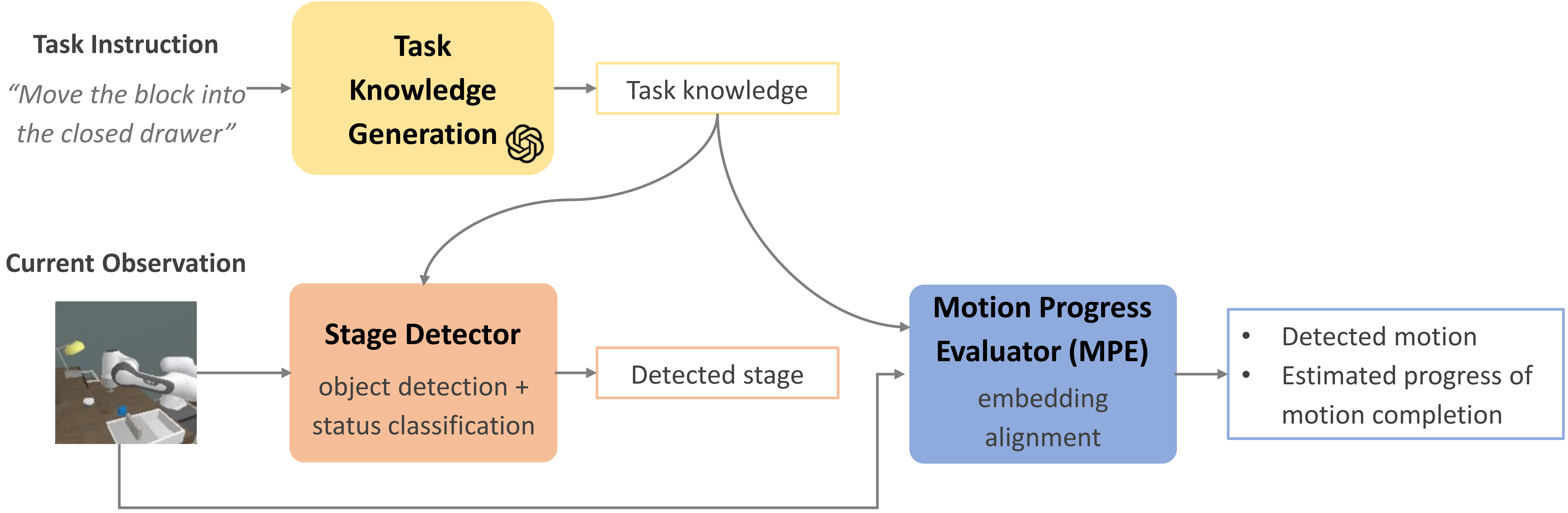


Stage Determination

- Compare the estimated status with the ideal ones in the task knowledge



VICtoR Components



Motion Determination & Progress Estimation

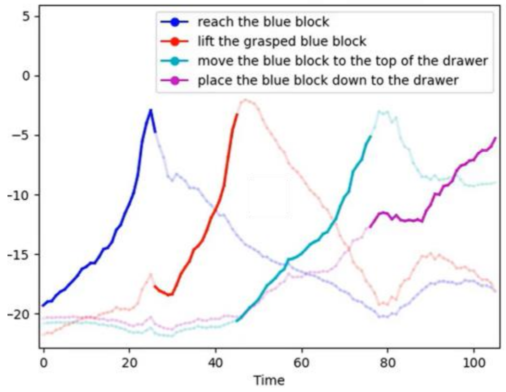
Motion Instructions

M1 M2 M3 M4

Current Observation

Motion Progress Evaluator (MPE)

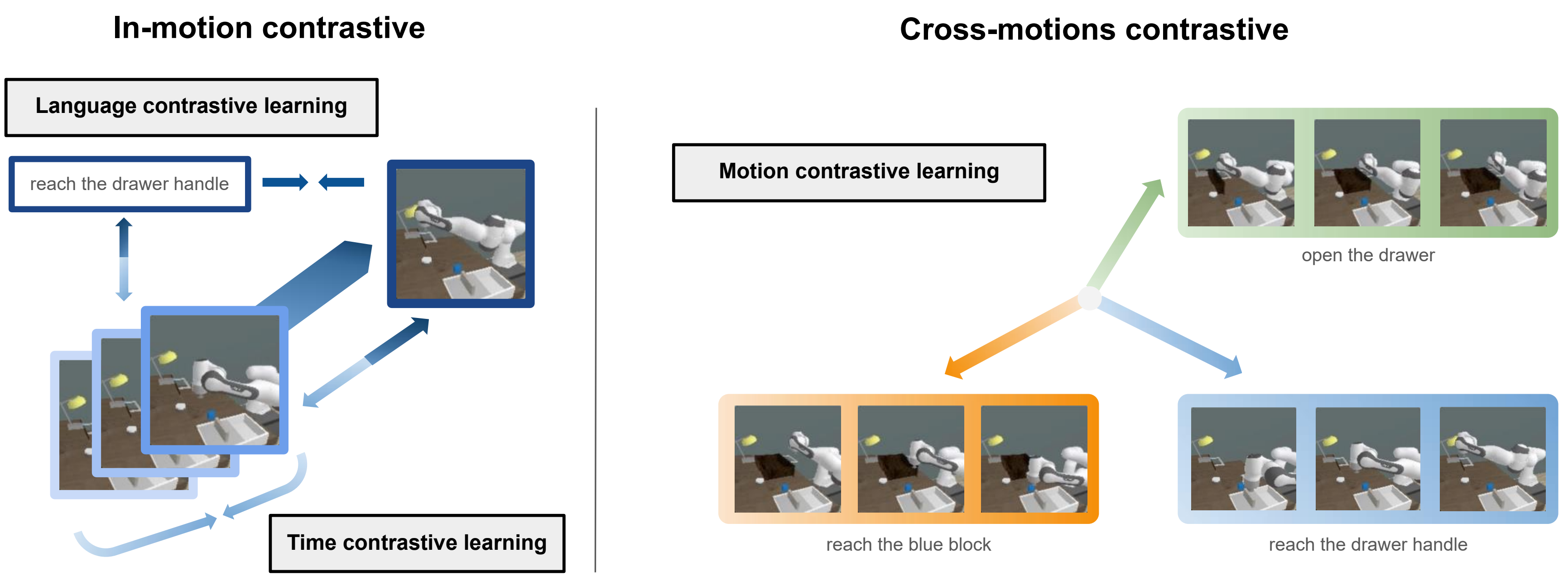
- CLIP-variant model
- Embedding distance



Detected Motion
M1 - Reach the block

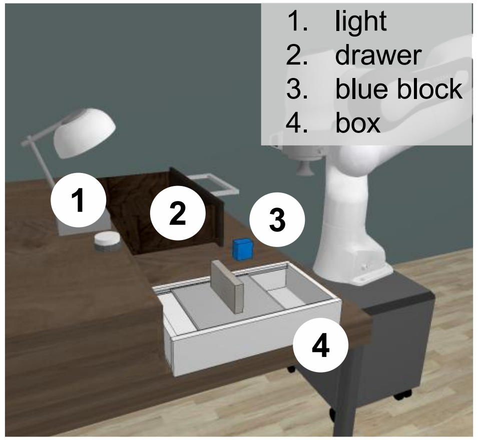
Estimated Progress
10%

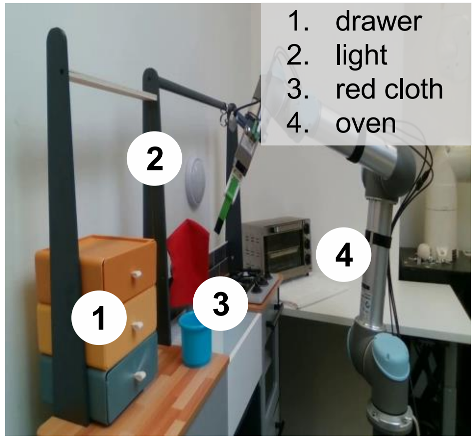
Training Objectives for MPE



- **Time Contrastive:** Frame embeddings with shorter temporal distances should be closer
- **Language (Progress) Contrastive:** Frame embeddings near the end of a motion should be closer to the motion instruction embedding
- **Motion Contrastive:** Frame embeddings should be closer to the embedding of their corresponding motion instruction

Evaluations – Environments and Baselines

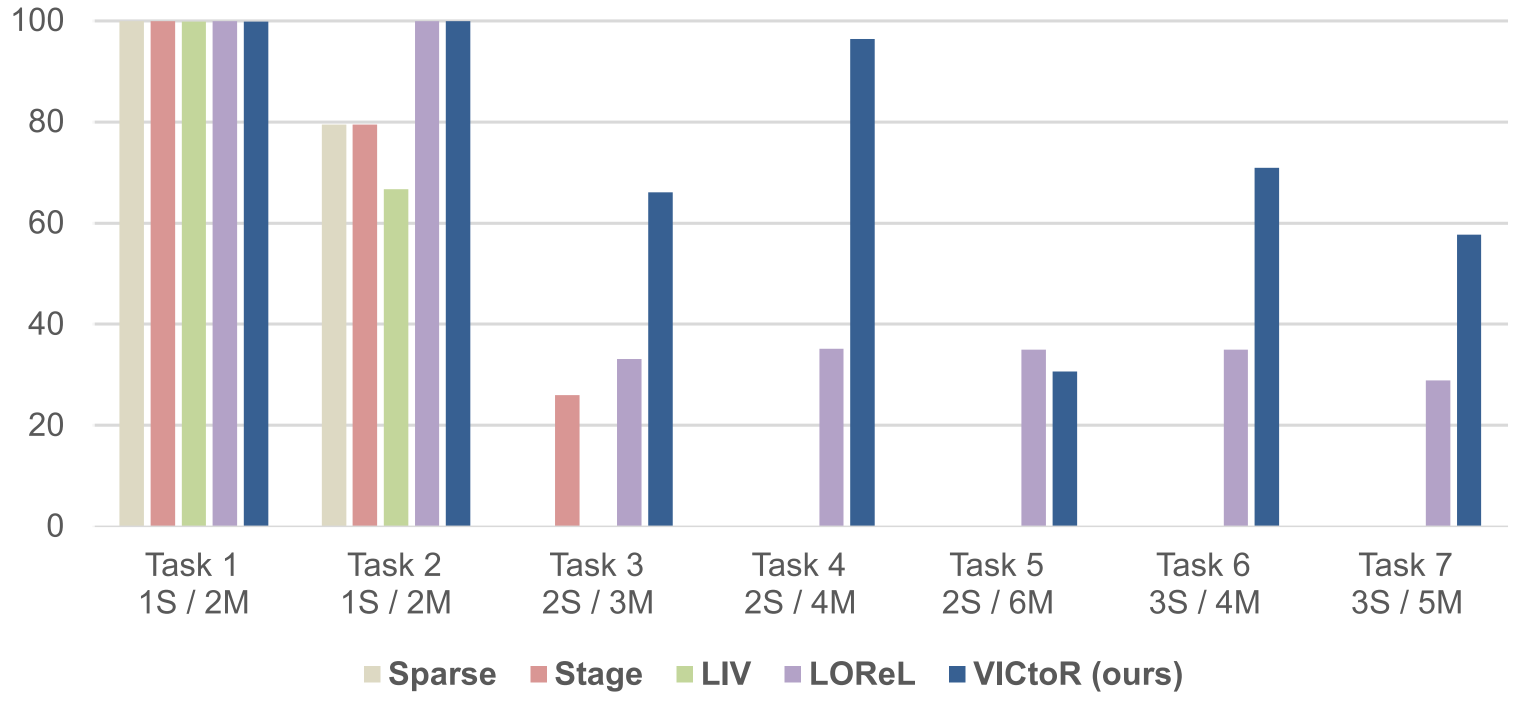




Environment	Tasks	Total Demos	Dataset Type
Simulated	9	2300	Machine Demos
Real World (XSkill)	8	360	Machine Demos

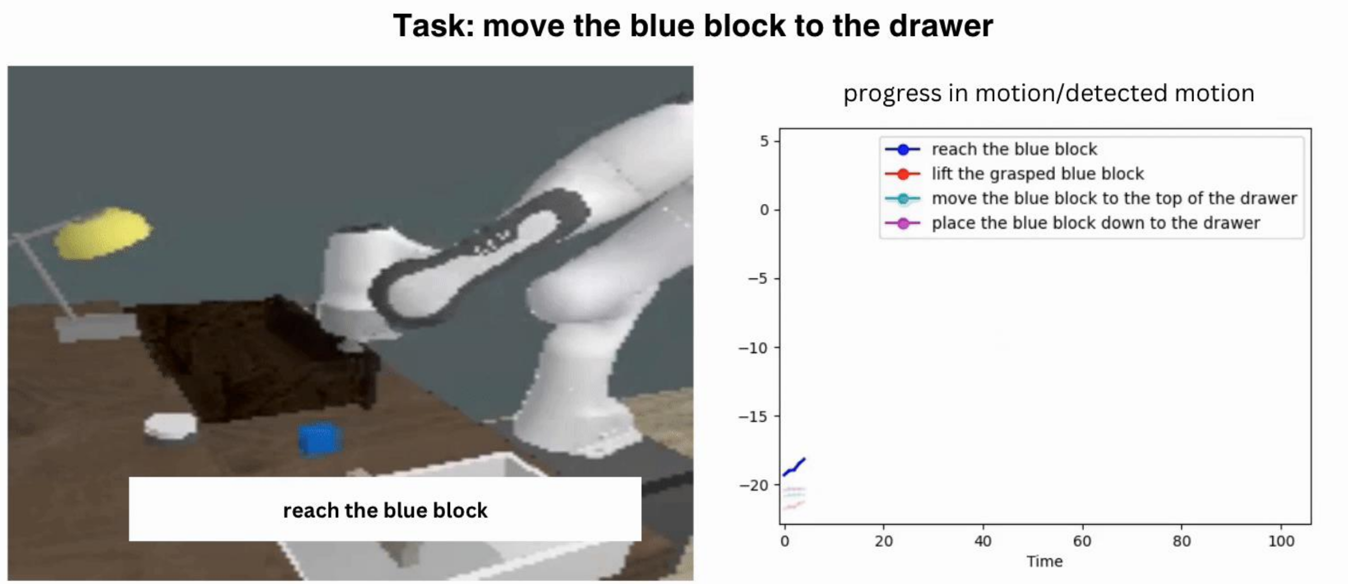
- Evaluations are conducted on both simulated and real-world experiments:
  - **Simulated:** reward learning + policy training
  - **Real-world (XSkill):** reward learning only
- **Baselines**
  - Sparse reward
  - Stage reward
  - LOReL (VIC-based)
  - LIV (VIC-based)
  - VICtoR (task-level)

Performance Comparison: PPO learned with Different Reward Models



- S and M indicate the number of stages and motions to complete the task
- The **same RL algorithm**, PPO, **trained with VICtoR** can learn **more complicated tasks**

Motion Determination Visualization

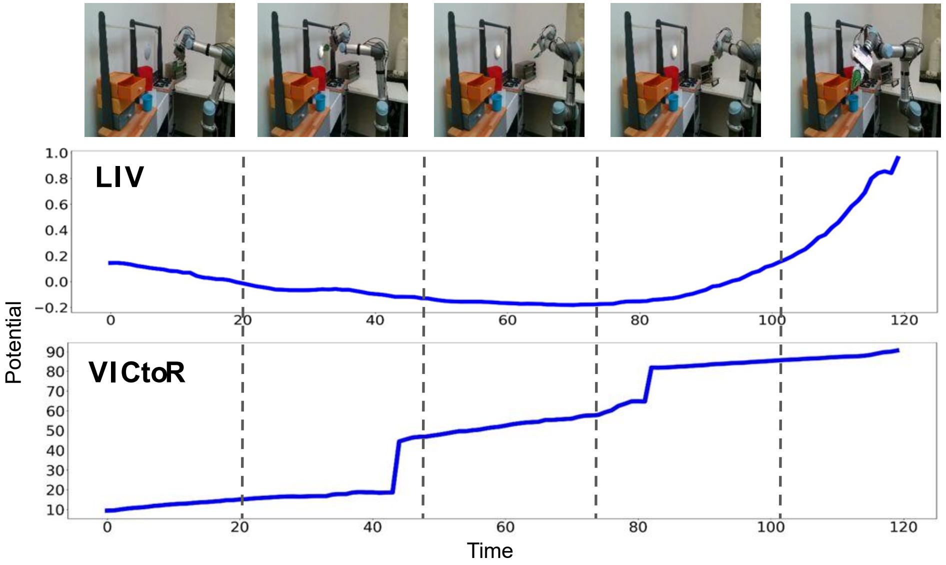


VICtoR can detect and assess progress in complex tasks and assign appropriate learning signals to the robot.

- Completing the task requires **four** distinct motions
- VICtoR **precisely identifies** the robot's current motion

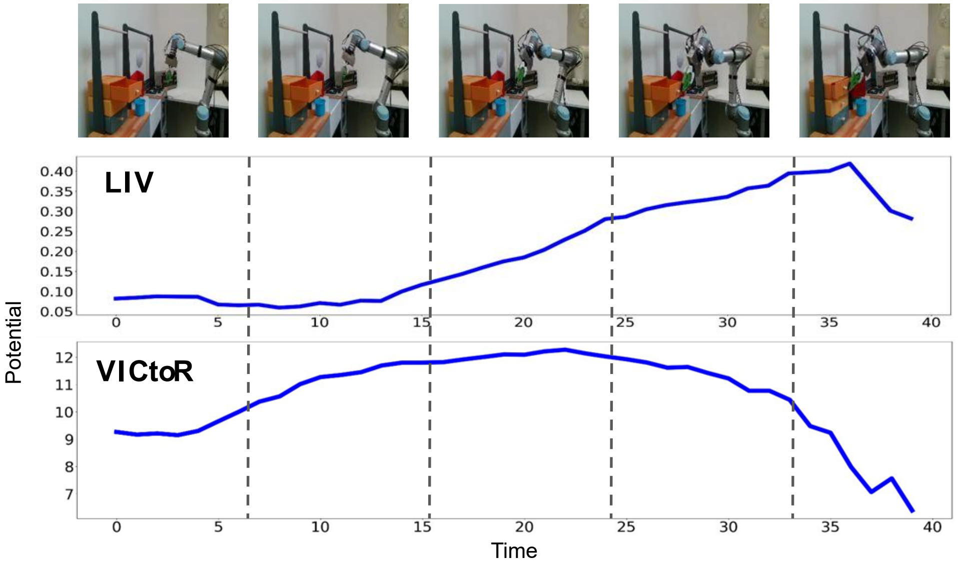
Rewards Generated for Different Cases

With the right actions



tested video: open light then open oven door then close the drawer
instruction: open light then open oven door then close the drawer

With the wrong actions



tested video: close the drawer
instruction: open light then open oven door then close the drawer

- In the **correlated case**, VICtoR generates meaningful rewards for **task progress**
- In the **uncorrelated case**, it recognizes **mismatches** and adjusts its rewards

Summary & Takeaways

- This work is the **first** to explore VIC reward models for **long-horizon tasks**
- By evaluating task progress at three **different granularities**, VICtoR generates **nuanced** and **informative** rewards
- Experimental results show that VICtoR enables the same RL algorithm to tackle more **complex**, long-horizon tasks, supported by extensive **visualization results**