LLM to Bridge Human Instructions with a Dynamic Symbolic Representation in Hierarchical Reinforcement Learning

Zihe Ji[1,2], **Mehdi Zadem**[3], **Sao Mai Nguyen**[1,4]
[1]Flowers Team, U2IS, ENSTA Paris, IP Paris & Inria
[2] Shanghai Jiaotong University
[3]LIX, École Polytechnique,Institut Polytechnique de Paris, France
[4]IMT Atlantique, Lab-STICC, UMR CNRS 6285
zihe.ji@ensta-paris.fr, zadem@lix.polytechnique.fr, nguyensmai@gmail.com

Abstract

Compositionality is addressed by Hierarchical Reinforcement Learning (HRL) by breaking down complex tasks into manageable subtasks, but faces challenges with efficiency and generalization in continual learning environments. Potential solutions to these limitations include a dimensional reduction of the high-level state space through a symbolic representation and region of interest identification through language input for imitation learning. In this work, we propose the integration of a dynamic symbolic representation and large language models (LLM) in the framework of HRL, leveraging LLM's natural language and reasoning capabilities to bridge the gap between human instructions and an emerging abstract representation. By acting as an interface for translating human demonstrations into actionable reinforcement learning signals, LLM can improve task abstraction and planning within HRL. Our approach builds upon the Spatial-Temporal Abstraction via Reachability (STAR) algorithm, using LLM to optimize the hierarchical planning process. We conduct experiments in ant robot environments, showing how LLM can translate abstract spatial states into symbol representations and assist with task planning. The results demonstrate the potential of LLM to enhance HRL in continual multi-task learning environments requiring spatial reasoning and hierarchical control.

1 Introduction

In continual learning, to solve long-horizon tasks such as compositional tasks, also referred to as sequential tasks, based on the ability to combine simple behaviors to create more complex behaviors, we examine a novel strategy for compositional learning based on the combination of imitation learning using language instructions and Reinforcement Learning (RL). While language excels at composing simpler tokens into complex ideas, it relies on pre-defined symbols. On the other hand, RL for continual learning environments need to tackle continuous high dimensional environments. We here propose a path to bridge the two worlds in the framework of Hierarchical Reinforcement Learning (HRL). Indeed HRL approaches allow agents to solve complex, long-horizon problems by decomposing them into easier, more manageable sub-problems. Unlike other HRL algorithms that use only continuous space or continuous abstract representation, the STAR [Zadem et al., 2024] algorithm automatically learns a discrete abstract goal space that preserves environment dynamics by focusing on reachability relations between sets of states. This abstraction acts as a discretisation of the state space, where every goal is a set of states that exhibit similar reachability properties in the task. This goal representation is acquired online. Despite these advances, the learned goals are not directly

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

interpretable since misaligned with human representations such as in natural language, making it a difficult for a human user to actively interact with the system. An important aspect of building systems that can serve as intelligent assistants to humans, is to ensure that we can communicate with them in a way that is intuitive and efficient. A learning agent should allow users to provide feedback on its behavior and to instruct it to perform specific tasks. Users should also be able to guide the learning process just as we teach and coach other humans. Inversely, robots should also be capable of asking questions to the user when they are uncertain about the task they are performing or to proactively seek guidance when they are stuck.

On the other hand, language, as represented by Large Language Models (LLMs) show composition and reasoning capabilities that can be beneficial to abstract representations. Following the principles of a human-centered approach [Boy, 2017], the machine should ground its reasoning in a common language with humans. The Human In The Loop (HITL) [Wu et al., 2021, Retzlaff et al., 2024] Reinforcement Learning paradigm studies how to integrate humans in the different stages of an agent's life cycle. This includes how human demonstrations can be used to enhance the learning process of primitive [Nguyen and Oudeyer, 2012] or sequential [Duminy et al., 2019] tasks, and how humans can instruct RL agents via natural language [Colas et al., 2020]. In this vein, the integration of LLMs in synergy with RL agents has recently gained attention. The advances achieved in building LLMs (e.g OpenAI's GPT, Meta's LLAMA, Anthropic's Claude), have accelerated the creation of language based HITL approaches [Pternea et al., 2024]. First, RL can be used in service of training and improving LLMs in natural language tasks such as conversation and question answering. In the vein of approaches, Reinforcement Learning from Human Feedback (RLHF) [Ouyang et al., 2022] has demonstrated how human feedback can be captured by a RL agent and used to fine-tune large language models. Inversely, a LLM can benefit RL agents in improving sample efficiency and injecting a reasoning layer [Du et al., 2023] that would alleviate the need for extensive exploration, especially in the initial training phases. A popular example of such approaches rely on the LLM as a high-level planner, providing instructions to the RL agent [Wong et al., 2023, Ichter et al., 2022, Wu et al., 2023], which can then be used to guide the learning process. Under such architectures, the LLM has to communicate with the RL agent in a common language that allows to express goals. Establishing this common language is a challenging task, and often researchers resort to using predefined predicates reducing the generality of the approach.

We propose in this paper some perspectives on how the interpretability of the reachability-aware goal abstraction in STAR can allow for a LLM to reason about abstract goals and boost the planning capabilities of the approach. We argue that this approach on the one hand allows humans to instruct the algorithm in natural language, and on the other hand, to allow the algorithm to clearly communicate its behavior. Our main contributions in this work is to explore using LLMs as high-level instructor for the STAR algorithm, and whether it can interpret agent behaviour to humans.

2 Spatial-Temporal Abstraction via Reachability (STAR) Algorithm

We base our work on the STAR algorithm, which efficiently partitions the state space. The partitioning data from the STAR algorithm is collected and used to test the integration of language instructions in the hierarchical reinforcement learning framework.

2.1 Overview of the STAR Algorithm

We consider a goal-conditioned Markov Decision Process $(\mathcal{S}, \mathcal{A}, P, r_{\text{ext}})$, where $\mathcal{S} \subseteq \mathbb{R}^n$ is a continuous state space, $\mathcal{A}$ is an action space, $P\left(s_{t+1} \mid s_t, a_t\right)$ is the transition function, and $r_{\text{ext}} : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is the reward, defined as the negative distance to the goal $g^* \in \mathcal{S}$: $r_{\text{ext}}\left(s, g^*\right) = -\left\|g^* - s\right\|_2$. The objective in multi-task reinforcement learning is to learn a goal-conditioned policy π that maximizes the expected reward by sampling actions $a \sim \pi\left(s_t \mid g^*\right)$ at each timestep.

The goal abstraction is modeled by a function $\mathcal{N} : \mathcal{S} \to 2^{\mathcal{S}}$ that maps states to sets of states (i.e., $\forall s \in \mathcal{S}, \mathcal{N}(s) \subseteq \mathcal{S}$). We refer to the abstract goal space as $\mathcal{G}_{\mathcal{N}}$ or simply $\mathcal{G}$ when clear. The elements of $\mathcal{G}$ are denoted as G.

The function $\mathcal{N}$ varies depending on the abstraction method. For example, Mannor et al. [2004] use stochastic partitioning with linear subpolicies, while the STAR algorithm [Zadem et al., 2024] uses

k-step reachability: a state s can reach s' using policy $\pi(., G_j)$ in k steps. Thus, the abstract goal space $\mathcal{G}$ consists of sets of reachable states.

The STAR architecture comprises three hierarchical agents:

- **Navigator**: The high-level agent selects an abstract goal $G \in \mathcal{G}$ to guide the agent towards the task goal g^*: $G_{t+k} \sim \pi_{\text{Nav}}(s_t, g^*)$.
- **Manager**: The mid-level agent picks subgoals in the state space, conditioned on the Navigator's goal: $g_{t+l} \sim \pi_{\text{Man}}(s_t, G_{t+k})$.
- **Controller**: The low-level policy samples actions to reach the subgoal: $a \sim \pi_{\text{Cont}}(s_t, g_{t+l})$.

The Manager and Controller use TD3 [Fujimoto et al., 2018] for learning, while the Navigator employs Q-learning. Each agent operates at different timescales: the Navigator selects a goal every k steps, the Manager every l steps (with k a multiple of l), and the Controller at each step. Initially, the abstraction $\mathcal{G}$ is coarse, making direct goal-reaching challenging. The Manager's subgoals serve as intermediate targets, facilitating easier learning for the Controller. This structure allows STAR to guide the agent through large state abstractions while supporting low-level policy learning.

2.2 Integration of LLM

For tasks in real-world environments, humans intuitively understand and navigate them. For instance, navigating a maze, moving from the living room to the kitchen, can be easily communicated using language. To reason and compose symbols grounded in a continuous environment, we take advantage of the discrete representation output by STAR as an intermediary capable of extracting the abstract spatial states of the algorithm and human instructions, then converting them into a format the algorithm can understand, ultimately accelerating the learning process. To achieve this, we propose the conversion of abstract spatial states and goals into a textual representation using LLM.

As the top-level agent, the Navigator only selects the next abstract region $G_{t+k} \sim \pi_{\text{Nav}}(s_t, g^*)$, we propose a translation instruction experiment. In the first experiment, we test the ability of LLMs to perform full route planning based on human-provided instructions, $(G_{t+k}, \ldots, G_{t+nk}) \sim \pi_{\text{LLM}}(X, s_t, g^*)$. Simultaneously, from another perspective, to evaluate the interactivity and alignment of LLM with spatial reasoning, we propose a naming experiment. In the second experiment, we translate abstract regions, $G \in \mathcal{G}$, into natural language descriptions and test whether LLM can support the mapping between continuous spatial regions and symbolic representations.

2.3 Representation of States and Goals

The Ant, adapted from Duan et al. [2016] and Nachum et al. [2018], is a simulated quadrupedal robot with a 30-dimensional state space, including positions, orientations, velocities, and joint angles. The action space is continuous and 8-dimensional, corresponding to forces applied on the joints.

We evaluate two tasks in a 2D environment of size 25 for each dimension: **AntMaze**, where the Ant navigates a $\supset$-shaped maze to the exit, and **AntFall**, which involves crossing a chasm using a movable block as a bridge. These tasks are hierarchical, requiring both low-level movement and high-level navigation. The environment uses Mujoco physics simulator [Todorov et al., 2012]. A training episode lasts up to 500 timesteps. The reward is the negative Euclidean distance to the goal, scaled by 0.1, with success if the distance is smaller than 5.

We use the partitioning from the STAR algorithm's training to test integrating human demonstrations. Human instructions guide the agent in the AntMaze or AntFall environments. To represent partitioning data as prompts for the LLM, we use:

- **Maze layout**: Compressed textual form with marked obstacles and partition regions.
- **Coordination information**: Tracks the agent's current location and the goal.
- **Adjacency list**: Details neighboring relations for each region.

3 Experimental Evaluation

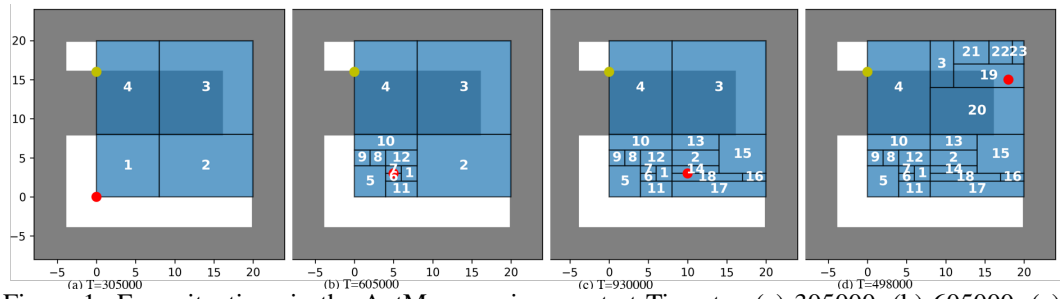

Figure 1: Four situations in the AntMaze environment at Timestep (a) 305000, (b) 605000, (c) 930000, and (d) 4980000. The red point is the agent's current location, the yellow point is the goal.

Table 1: Unique Name given by LLM

Timestep - Region	LLAMA3.1-8B-Instruct	GPT4o
305000 - 2	Rightward Passage	Eastern Pathway
605000 - 1	**Western** Entrance	**Southern** Junction
605000 - 5	Leftward Passage	Western Approach
605000 - 6	**Rightward** Passage	**Northern** Link
605000 - 12	**Southern** Expansion	**Eastern** Border
4980000 - 3	Northern Passage	Northern Access
4980000 - 20	Southern Corridor	Southern Corridor
4980000 - 21	**Eastern** Extension	Northeastern Outlet

Naming Experiment for Spatial Regions To evaluate the LLMs' ability to generate human-readable descriptions of abstract goals, we utilized four scenarios with Timestep 930000 as a one-shot prompt (see Fig.1 and annex A). Llama3-8b-instruct and GPT-4o were tested; the former runs on a GPU with more than 16GB of RAM. Completing the STAR program for 5 million timesteps takes about 15 hours, with LLM inference taking 0.6 seconds each. Table 1 shows the names given by the LLMs, when tasked with naming neighboring regions. The results indicate that the LLMs can generate clear and concise names for each region.

In Table 1, bold text denotes incorrect region descriptions. The LLMs struggled with directional accuracy, particularly in densely packed situations (e.g., Timestep 605000, with 25% accuracy). However, when focusing on regions adjacent to the agent's location, directional accuracy exceeded 75%, suggesting that representing continuous regions as symbolic names using LLMs is feasible.

Instruction Translation Experiment In Fig. 1.b, a LLM might compose instructions into a complex planning : "Go east, then north past the wall, and finally west to the goal." With region segmentation, this means *moving through regions* ($G_n = (1, 2, 3, 4)$. We tested the LLM's ability to infer this sequence from such instructions (see annex C for the promts). We report the accuracy, defined as IoU $= \frac{|G_{\text{LLM}} \cap G_n|}{|G_{\text{LLM}} \cup G_n|}$, where G_{LLM} is the LLM-predicted sequence and G_n is the true sequence. Table 2 reports the IoU for his ChatGPT-4o and Claude 3.5 Sonnet, with detailed outputs reported in annex D. GPT-4o's errors stemmed from omitting intermediate regions, while Claude added extra ones. Both models achieved over 80% IoU, with 100% accuracy in predicting the next region, indicating effective translation of instructions into abstract regions.

Table 2: IoU Comparison Between GPT-4o and Claude 3.5 Sonnet

Environment	GPT-4o	Claude 3.5 Sonnet
AntMaze	82.1%	90.3%
AntFall	75.0%	100.0%
Total	81.25%	91.43%

4 Discussion

The experiments show that LLMs can enhance HRL tasks, particularly in sequential planning, despite its dynamic abstract representation. This is owing to its emergent symbolic representation capable to handle long-horizon tasks in continual learning. LLMs effectively bridge human instructions and

HRL, aiding task abstraction owing to its reasoning capability. Challenges remain in densely packed environments where directional errors occur. Our work opens the door to compositional reasoning for representation learning in reinforcement learning. Future work should refine LLM spatial reasoning and apply this approach to dynamic, real-world tasks. Overall, integrating LLMs into HRL can improve complex task performance in hierarchical control and spatial reasoning contexts.

Acknowledgments and Disclosure of Funding

This work was partially supported by Hi! Paris.

Acknowledgements

The authors would like to thank Sergio Mover for his support.

References

G. A. Boy. *The handbook of human-machine interaction: a human-centered design approach*. CRC Press, 2017.

C. Colas, A. Akakzia, P. Oudeyer, M. Chetouani, and O. Sigaud. Language-conditioned goal generation: a new approach to language grounding for RL. *CoRR*, abs/2006.07043, 2020.

Y. Du, O. Watkins, Z. Wang, C. Colas, T. Darrell, P. Abbeel, A. Gupta, and J. Andreas. Guiding pretraining in reinforcement learning with large language models. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 8657–8677. PMLR, 2023.

Y. Duan, X. Chen, R. Houthooft, J. Schulman, and P. Abbeel. Benchmarking deep reinforcement learning for continuous control. In *International conference on machine learning*, pages 1329–1338. PMLR, 2016.

N. Duminy, S. M. Nguyen, and D. Duhaut. Learning a set of interrelated tasks by using a succession of motor policies for a socially guided intrinsically motivated learner. *Frontiers in neurorobotics*, 12:87, 2019.

S. Fujimoto, H. Hoof, and D. Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR, 2018.

B. Ichter, A. Brohan, Y. Chebotar, C. Finn, K. Hausman, A. Herzog, D. Ho, J. Ibarz, A. Irpan, E. Jang, R. Julian, D. Kalashnikov, S. Levine, Y. Lu, C. Parada, K. Rao, P. Sermanet, A. Toshev, V. Vanhoucke, F. Xia, T. Xiao, P. Xu, M. Yan, N. Brown, M. Ahn, O. Cortes, N. Sievers, C. Tan, S. Xu, D. Reyes, J. Rettinghouse, J. Quiambao, P. Pastor, L. Luu, K. Lee, Y. Kuang, S. Jesmonth, N. J. Joshi, K. Jeffrey, R. J. Ruano, J. Hsu, K. Gopalakrishnan, B. David, A. Zeng, and C. K. Fu. Do as I can, not as I say: Grounding language in robotic affordances. In *CoRL*, volume 205 of *Proceedings of Machine Learning Research*, pages 287–318. PMLR, 2022.

S. Mannor, I. Menache, A. Hoze, and U. Klein. Dynamic abstraction in reinforcement learning via clustering. In *Proceedings of the twenty-first international conference on Machine learning*, page 71, 2004.

O. Nachum, S. S. Gu, H. Lee, and S. Levine. Data-efficient hierarchical reinforcement learning. *Advances in neural information processing systems*, 31, 2018.

S. M. Nguyen and P.-Y. Oudeyer. Active choice of teachers, learning strategies and goals for a socially guided intrinsic motivation learner. *Paladyn*, 3(3):136–146, 2012.

L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Christiano, J. Leike, and R. Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc., 2022.

M. Pternea, P. Singh, A. Chakraborty, Y. D. Oruganti, M. Milletarí, S. Bapat, and K. Jiang. The RL/LLM taxonomy tree: Reviewing synergies between reinforcement learning and large language models. *CoRR*, abs/2402.01874, 2024.

C. O. Retzlaff, S. Das, C. Wayllace, P. Mousavi, M. Afshari, T. Yang, A. Saranti, A. Angerschmid, M. E. Taylor, and A. Holzinger. Human-in-the-loop reinforcement learning: A survey and position on requirements, challenges, and opportunities. *J. Artif. Intell. Res.*, 79:359–415, 2024.

E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

L. Wong, J. Mao, P. Sharma, Z. S. Siegel, J. Feng, N. Korneev, J. B. Tenenbaum, and J. Andreas. Learning adaptive planning representations with natural language guidance. *CoRR*, abs/2312.08566, 2023.

J. Wu, Z. Huang, C. Huang, Z. Hu, P. Hang, Y. Xing, and C. Lv. Human-in-the-loop deep reinforcement learning with application to autonomous driving. *CoRR*, abs/2104.07246, 2021.

Y. Wu, S. Y. Min, Y. Bisk, R. Salakhutdinov, A. Azaria, Y. Li, T. M. Mitchell, and S. Prabhumoye. Plan, eliminate, and track - language models are good teachers for embodied agents. *CoRR*, abs/2305.02412, 2023.

M. Zadem, S. Mover, and S. M. Nguyen. Reconciling spatial and temporal abstractions for goal representation. *arXiv preprint arXiv:2401.09870*, 2024.

A Prompt for Naming Experiment

The system prompt used in the experiment *Naming Experiment for Spatial Regions:*

In this task, You are a navigation assistant, helping agent to reach the goal. Based on the data, name the adjacent regions with a unique name understandable by the user to help him follow your instructions.

The following is the few-shot prompt used in the experiment:

Role: User
Content:
Data:
State: [10, 3], Region 14
Goal: [0, 15], Region 4
Adjacency list:
Region 1: [5, 11, 18]
Region 2: [13]
Region 3: [4]
Region 5: [1, 6, 11]
Region 6: [1, 11]
Region 11: [1, 6, 17]
Region 13: [3, 15]
Region 14: [1, 2, 15, 18]
Region 15: [2, 3, 14]
Region 17: [18]
Region 18: [1, 14]
The top-down view of the maze is shown below, W represents walls, A represents the agent's current position,G represents the goal. The number represents the region number:
4 4 4 4 4 3 3 3 3 3 3
G 4 4 4 4 3 3 3 3 3 3
W W W W W W W W W 3 3
10 10 10 10 10 13 13 13 15 15 15
9 9 8 12 12 2 2 2 15 15 15
5 5 5 7 1 14 14 14 15 15 15
5 5 5 7 1 14 A 14 15 15 15
5 5 5 6 1 18 18 18 18 18 16
5 5 5 11 11 17 17 17 17 17 17

Thinking Process:
1.Determine which regions are directly adjacent to the agent's current location.
2.Assign Names Based on Relative Position: Use directional names (north, south, east, west) or (left, middle, right) based on the region's position relative to the agent or between the adjacent regions.
3.Special Features: If a region has unique features (like being entirely walled off or a sole connection to another important area), this should be reflected in the name.

Role: Assistant
Content:
Based on the maze layout and the adjacency list provided, the agent is currently in Region 14. The regions adjacent to Region 14 are Regions 1, 2, 15, and 18.

1. Region 1

Location: West of Region 14.
Unique Name: Western Gateway
Rationale: It's in the west direction relative to the agent's current position and acts as a connecting hub to other regions (5, 11, 18).

Table 3: Region status at different timesteps

Region	Timestep 305000	Timestep 605000	Timestep 930000	Timestep 4980000
Region 1	Top Left	Top-Left	Top Left Corner	One
Region 2	Top Middle	Top-Middle	Not Available	Two
Region 3	Top Right	Top-Right	Not Available	Three
Region 4	Bottom	Goal	Goal Region	Goal
Region 5	-	Bottom-Left	Top Middle	Five
Region 6	-	Bottom-Middle	Top Right	Six
Region 7	-	Bottom-Right	Not Available	Seven
Region 8	-	Not Found	Not Available	Eight
Region 9	-	Not Found	Not Available	Nine
Region 10	-	Not Found	State Region	Ten
Region 11	-	Bottom-Right Corner	Upper Middle	Eleven
Region 12	-	Not Found	Not Available	Twelve
Region 13	-	-	Down Left	Thirteen
Region 14	-	-	Current Region	Fourteen
Region 15	-	-	Down Middle	Fifteen
Region 16	-	-	Not Available	Sixteen
Region 17	-	-	Down Right	Seventeen
Region 18	-	-	Bottom Right	Current
Region 19	-	-	-	Nineteen
Region 20	-	-	-	Twenty
Region 21	-	-	-	Twenty-One
Region 22	-	-	-	Twenty-Two
Region 23	-	-	-	Twenty-Three

2. Region 2:

Location: Directly above Region 14 in the maze.
Unique Name: Northern Corridor
Rationale: It is located to the north of the agent's current position and appears to be a narrow passage leading to Region 13.

3. Region 15:

Location: Right next to Region 14 on the east side.
Unique Name: Eastern Gateway
Rationale: This region connects to multiple other regions (Regions 2, 3, and 14) and serves as a gateway towards the eastern part of the maze.

4. Region 18:

Location: Directly below Region 14 in the maze.
Unique Name: Southern Expansion

Rationale: It extends to the south and also connects to Region 1 and 17, offering a larger area or expansion from the agent's current position.

B Zero-shot Naming Experiment

The table 3 shows the result of the zero-shot naming experiment. In this experiment, no shots were given to the LLM, and the LLM was asked to name all the regions, instead of concentrating on the regions adjacent to the agent's current location. The results show that the LLM give up when facing too complex tasks, such as naming all the regions in the maze. Therefore, we asked the LLM to focus on naming the regions adjacent to the agent's current location in the few-shot naming experiment.

Table 4: Region Information Across Timesteps

Timestep - Region	Location Description	Unique Name
	LLAMA3.1-8B-Instruct	
305000 - 2	Directly right of Region 1 in the maze	Rightward Passage
605000 - 1	West of Region 7	Western Entrance
605000 - 5	Directly left of Region 7	Leftward Passage
605000 - 6	Directly right of Region 7	Rightward Passage
605000 - 12	Below Region 7 in the maze	Southern Expansion
4980000 - 3	Directly above Region 19 in the maze	Northern Passage
4980000 - 20	Directly below Region 19 in the maze	Southern Corridor
4980000 - 21	Right next to Region 19 on the east side	Eastern Extension
	GPT4o	
305000 - 2	Directly east of Region 1 in the maze	Eastern Pathway
605000 - 1	South of Region 7	Southern Junction
605000 - 5	West of Region 7	Western Approach
605000 - 6	North of Region 7	Northern Link
605000 - 12	East of Region 7	Eastern Border
4980000 - 3	North of Region 19 in the maze	Northern Access
4980000 - 20	Below Region 19 in the maze	Southern Corridor
4980000 - 21	Northeast of Region 19	Northeastern Outlet

C Prompts for Instruction Translation Experiment

The system prompt used in this experiment: In this task, You are a navigation assistant, helping agent to reach the goal. Based on the data and the instruction given, tell the route to the agent to reach the goal.

The following is the few-shot prompt used in the experiment:

User:
Data:
State: [10, 3], Region 14
Goal: [0, 15], Region 4
Adjacency list:
Region 1: [5, 11, 18]
Region 2: [13]
Region 3: [4]
Region 5: [1, 6, 11]
Region 6: [1, 11]
Region 11: [1, 6, 17]
Region 13: [3, 15]
Region 14: [1, 2, 15, 18]
Region 15: [2, 3, 14]
Region 17: [18]
Region 18: [1, 14]
The top-down view of the maze is shown below, W represents walls, A represents the agent's current position,G represents the goal. The number represents the region number:
4 4 4 4 4 3 3 3 3 3 3
G 4 4 4 4 3 3 3 3 3 3
W W W W W W W W W 3 3
10 10 10 10 10 13 13 13 15 15 15
9 9 8 12 12 2 2 2 15 15 15
5 5 5 7 1 14 14 14 15 15 15
5 5 5 7 1 14 A 14 15 15 15

5 5 5 6 1 18 18 18 18 18 16
5 5 5 11 11 17 17 17 17 17 17

Instruction:
Go straight to east till the end, then go north till pass the wall. Finally go west and reach the goal.

Thinking Process:
1. Identify the agent's current region and the goal region.
2. Interpret the Instruction: Understand the directional commands provided in the instruction and translate them into movements between regions.
3. Plan the Route: Based on the adjacency list and the maze layout, determine the sequence of regions the agent should traverse to follow the given instructions and reach the goal.

Assistant:

Current State: The agent is located in Region 14, at coordinates [10, 3].
Goal: The target is in Region 4, at coordinates [0, 15].

Interpretation of Instructions:
Go straight to the right till the end: Starting from Region 14, moving east would involve transitioning through adjacent regions towards the eastern edge of the maze.
Then go north till pass the wall: After reaching the furthest right point accessible, the agent should then move northwards until they pass the wall.
Finally go west and reach the goal: The agent needs to move left from the northernmost point they reached, continuing until they arrive at the goal in Region 4.

Route Plan Based on Regions and Their Adjacencies:
Starting at Region 14: Move eastward within the maze. Adjacent regions from 14 are [1, 2, 15, 18]. The furthest east the agent can go from here within the maze layout while still adhering to the instruction to move to the right and then up includes moving to Region 15.

Move to Region 15: After reaching Region 15, the next step is to go north. From Region 15, the agent can continue to Region 3 via Region 13 (15 → 13 → 3). Region 13 doesn't pass the wall, so the agent should continue to Region 3.
Move to Region 3: On reaching Region 3, the agent would then move to the left, as instructed.
Region 3 directly connects to Region 4, which is the goal region.

Final Route Summary:
Region 14 → Region 15 → Region 13 → Region 3 → Region 4 (Goal)

D Detailed Results of Instruction Translation Experiment

The table 5 shows the detailed instructions given and the routes planned by ChatGPT-4o and Claude 3.5 Sonnet. The results show that both models succeed in choosing the first region in the sequence by interpreting the instruction correctly.

Table 5: Route Planning Results Comparison between ChatGPT-4o and Claude 3.5

Case	Instruction	ChatGPT-4o Route	Claude 3.5 Route
1	Go east until pass the wall, then go north until passing the wall, turn west and reach the goal.	7 → 1 → 2 → 3 → 4 (goal)	7 → 1 → 2 → 3 → 4 (goal)
2	Go north a little bit until pass the wall, then go west till the end, and the goal is on the left.	19 → 3 → 4 (goal)	19 → 21 → 3 → 4 (goal)
3	Go east until pass the wall, then go north till the end, the goal is on the left.	9 → 6 → 10 → 2 → 16 (goal)	9 → 8 → 11 → 2 → 3 → 15 → 14 → 16 (goal)
4	Go straight to the west, the goal is at the north.	20 → 12 → 11 (goal)	20 → 12 → 5 → 1 → 11 (goal)
5	Go east til pass the wall, then go south until the end, finally go west and reach the goal.	4 → 3 → 19 → 15 → 14 → 1 → 5 (goal)	4 → 3 → 19 → 20 → 15 → 2 → 12 → 8 → 5 (goal)
6	Go straight to east till the end, then go north to push the movable block.	10 → 2 → 15 (block)	10 → 5 → 2 → 15

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 paper presents a novel approach to integrating large language models (LLMs) into hierarchical reinforcement learning (HRL) tasks, demonstrating the potential of LLMs in enhancing task abstraction and planning.

   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 discusses the limitations of the LLMs in naming regions, particularly in densely packed environments, and the challenges faced in scaling this approach to real-world tasks.

   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 paper provides detailed information on the experimental setting, including the maze layout, agent's current location, goal region, and adjacency list, to reproduce the main experimental results.

   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 code will be made available upon acceptance, and the paper provides detailed instructions to reproduce the main experimental results.

   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 paper provides detailed information on the experimental setting, including the maze layout, agent's current location, goal region, and adjacency list, to understand the results. The data used in the experiments will also be made available upon acceptance.

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: [Yes]

Justification: The paper provides information about the statistical significance of the experiments.

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: Llama3-8b-instruct requires a GPU with more than 16GB of VRAM for deployment. The other experiments can be run online.

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: The research conducted in the paper conforms with the NeurIPS Code of Ethics.

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: [NA]

Justification: The paper does not address societal impact.

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 release data or models that have a high risk for misuse.

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: The paper properly credits the creators of the code and data used in the experiments and provides the license and terms of use.

    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: [NA]

    Justification: The paper does not introduce new assets.

    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: The paper does not involve crowdsourcing nor 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: The paper does not involve crowdsourcing nor 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.*