Evaluating Action Tracking in ReAct Agents for Text-Based Game Navigation

CodeScientist

February 26, 2025

Abstract

This paper presents an experimental evaluation of action tracking mechanisms in ReAct (Reasoning and Acting) agents within text-based game environments. We compare three agent variants: a random baseline, a standard ReAct implementation, and a ReAct agent augmented with action history tracking. Using TextWorldExpress's CookingWorld environment, we evaluate these agents across 50 episodes, measuring task completion rates and average scores. Our results show that while both ReAct variants significantly outperform the random baseline, the addition of action tracking does not provide statistically significant improvements over the standard ReAct implementation.

1 Introduction

Text-based games present unique challenges for reinforcement learning agents, requiring both natural language understanding and strategic planning. The ReAct framework has shown promise in these environments by combining reasoning with action selection. This study investigates whether augmenting ReAct agents with explicit action history tracking can improve performance in cooking task scenarios.

2 Methodology

2.1 Experimental Setup

We utilized TextWorldExpress's CookingWorld environment with simplified parameters: single location, three ingredients, no distractor items, and no doors. The experiment comprised 50 episodes with a maximum of 30 steps per episode, using seeds 1-50 for reproducibility.

CodeScientist

2.2 Agent Implementations

Three agent variants were tested:

- Random Agent: Baseline implementation using uniform random action selection
- ReAct Baseline: Standard ReAct implementation using GPT-4 for reasoning
- ReAct+ActionTracking: Extended ReAct implementation with JSON-based action history tracking

The action tracking mechanism maintained a history of actions, their contexts, and outcomes, which was incorporated into the prompt for action selection.

3 Results

Table 1: Agent Performance Metrics

Agent	Average Score	Success Rate
Random	0.156	0.0%
ReAct Baseline	0.522	14.0%
ReAct+ActionTracking	0.507	14.0%

3.1 Performance Analysis

Both ReAct variants significantly outperformed the random baseline, with average scores more than three times higher. The standard ReAct implementation achieved an average score of 0.522, while the action-tracking variant achieved 0.507. Both ReAct variants achieved identical success rates of 14%.

3.2 Statistical Analysis

Bootstrap resampling analysis revealed:

- ReAct+ActionTracking vs. Random: Statistically significant improvement (p ¡ 0.001)
- ReAct+ActionTracking vs. ReAct Baseline: No significant difference (p = 0.678)

CodeScientist

4 Discussion

4.1 Key Findings

The experimental results do not support the hypothesis that action tracking improves ReAct agent performance in this environment. While both ReAct variants significantly outperform random action selection, the addition of action history tracking did not lead to measurable improvements in either average score or success rate.

4.2 Limitations

Several limitations should be considered:

- The simplified environment (single location, three ingredients) may not fully exercise the potential benefits of action tracking
- The implementation used a basic similarity metric for action history matching
- The experiment focused on cooking tasks, which may not generalize to other domains
- The maximum episode length of 30 steps may have limited the potential for learning from action history

5 Conclusion

While this study demonstrates the effectiveness of ReAct agents compared to random baselines in text-based game environments, it suggests that simple action tracking mechanisms may not provide additional benefits. Future work should explore more sophisticated action history utilization methods and test the approach in more complex environments.

6 Future Work

Future research directions include:

- Implementing more sophisticated similarity metrics for action history matching
- Testing in environments with increased complexity

CodeScientist

- Exploring different methods of incorporating action history into the decision-making process
- Investigating the impact of longer episode lengths on learning from history

CodeScientist