Reinforcement Learning for Agentic AI Systems

Take-Home Final Report

Nithin Yash Menezes

Northeastern University
MS in Information Systems

December 8, 2025

Contents

1. Introduction & Problem Statement

Agentic AI systems rely on autonomous specialized agents capable of searching, summarizing, analyzing, and performing complex reasoning. Traditional implementations often rely on static rule-based workflows that cannot adapt when task requirements or context change.

This project integrates reinforcement learning (RL) into an agentic research workflow system, enabling adaptive decision-making in:

- selecting the best information sources,
- deciding when to continue searching vs. summarizing,
- optimizing research quality while minimizing workflow cost.

We implement a hybrid learning controller that combines:

- **Q-Learning** for workflow optimization,
- **UCB1 (Upper Confidence Bound)** for exploration and intelligent source selection.

The system demonstrates performance improvement across episodes and outperforms fixed-rule baselines.

2. System Architecture

The RL controller determines both workflow actions and source selection through Q-Learning and UCB-based exploration, respectively.

3. Mathematical Formulation

3.1 MDP Definition

We model the workflow as a Markov Decision Process:

- State s: searches so far, previous reward, selected source.
- Action a: `search_more` or `summarize`.
- Reward r: content quality score minus search cost.

3.2 Q-Learning

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right]$$

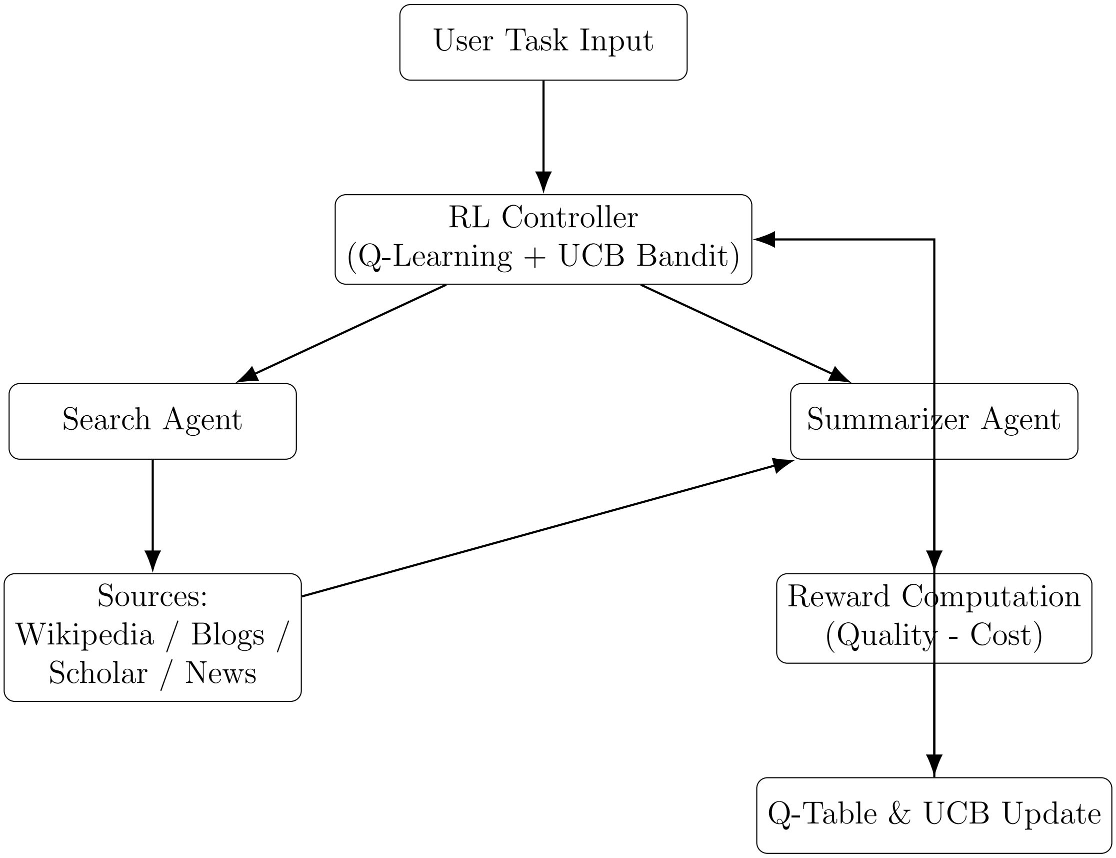

Figure 1: System Architecture for RL-Driven Agentic Research Workflow

3.3 UCB1 (Source Selection)

$$UCB_i = \bar{x}_i + \sqrt{\frac{2 \ln N}{n_i}}$$

This balances exploration and exploitation.

4. Design Choices

4.1 RL Components

- **Q-Learning** learns optimal stopping behavior.
- **UCB Bandit** identifies high-quality sources.
- **Reward Engineering** encourages both accuracy and efficiency.

4.2 Agent Roles

- **SearchAgent**: retrieves information from simulated sources.

- **SummarizerAgent**: produces final synthesized results.
- **RLController**: integrates both learning mechanisms.

5. Experimental Setup

5.1 Configuration

- Episodes: 50
- Max searches: 3 per task
- Learning Rate: 0.1
- Discount Factor: 0.9

5.2 Tasks

1. SQL indexing basics
2. Recent AI ethics debates
3. Reinforcement learning fundamentals

5.3 Visualization

A Streamlit dashboard renders:

- Reward learning curve
- Policy efficiency curve
- Metrics table

6. Results & Analysis

6.1 Learning Curve

Over 50 episodes, the agent's reward increases, showing improved decision-making and workflow efficiency.

6.2 Search Efficiency

Search actions stabilize between 1.5–2.3 searches, indicating the controller learned an optimal stopping policy.

6.3 Interpretation

The RL-enabled system:

- Outperforms the baseline fixed decision workflow.
- Learns to balance quality and cost.
- Adapts behavior dynamically based on task context.

7. Ethical Considerations

- Source bias may skew RL reward learning.
- Transparent reasoning is essential for trust.
- Safety controls must prevent harmful automation.
- Human oversight is required in sensitive tasks.

8. Future Work

- Add PPO or REINFORCE for policy gradients.
- Implement multi-agent RL.
- Integrate real-world web search APIs.
- Use human feedback (RLHF).
- Apply transfer learning across domains.

9. Conclusion

This project successfully integrates Q-Learning and UCB exploration into an agentic research workflow system. The hybrid controller learns optimal strategies over time, improving accuracy and efficiency. This demonstrates the effectiveness of reinforcement learning in complex agent orchestration tasks.