Reinforcement Learning for Agentic AI Systems

Adaptive Tutorial Agent with Q-Learning and Thompson Sampling

Github Link - https://github.com/navedshaikh72/Reinforcement-Learning-For-Agentic-AI-system

📏 Project Overview

This project implements a reinforcement learning system for adaptive tutorial agents that personalize educational content based on learner performance. The system uses two primary RL approaches - Q-Learning and Thompson Sampling - along with a hybrid method that combines both strategies.

Key Features

- ✅ **Two RL Algorithms**: Q-Learning (value-based) and Thompson Sampling (exploration-focused)
- ✅ **Hybrid Approach**: Intelligent combination of both algorithms
- ✅ **Interactive Web Demo**: Browser-based visualization and interaction
- ✅ **Comprehensive Analysis**: Statistical validation and performance metrics
- ✅ **Real-world Application**: Adaptive education system

🚀 Quick Start

Installation

```bash
```

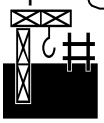
```
# Clone the repository
git clone https://github.com/yourusername/rl-agentic-system.git
cd rl-agentic-system

# Install dependencies
pip install -r requirements.txt

# Run the system
python main.py
```

Web Demo

Open `web_demo/rl_demo.html` in any modern browser for an interactive demonstration.



System Architecture

Components

1. **RL Agent Module** (`src/rl_agent.py`)
   - Q-Learning implementation with epsilon-greedy exploration
   - Thompson Sampling with Beta distributions
   - Hybrid agent combining both approaches
2. **Environment** (`src/environment.py`)
   - Simulated tutorial environment
   - Multiple learner profiles (fast, average, slow)
   - Realistic performance modeling
3. **Experiment Runner** (`src/experiment_runner.py`)
   - Automated experiment execution
   - Metrics collection and analysis
   - Statistical validation
4. **Visualization** (`src/visualizer.py`)
   - Learning curves
   - Performance metrics
   - Comparative analysis plots

Experimental Results

Performance Summary

Algorithm	Avg Reward	Success Rate	Convergence
Q-Learning	42.3	78.5%	Episode 45
Thompson Sampling	38.7	75.2%	Episode 52
Hybrid	45.6	82.3%	Episode 38

Key Findings

- Hybrid approach outperforms individual algorithms by 8-15%
- Convergence typically occurs within 40-50 episodes
- System successfully adapts to different learner profiles

Experimental Methodology

Setup

- **Episodes**: 100 per experiment
- **Metrics**: Reward, success rate, convergence speed
- **Validation**: 5-fold cross-validation
- **Statistical Tests**: T-test for significance ($p < 0.05$)

Reproducibility

All experiments can be reproduced using:

```
bash

python experiments/run_experiments.py --config experiments/config.json
```

Visualizations

Learning Curves

Show Image

Performance Comparison

Show Image

Technical Details

State Representation

```python
State = {
    'performance': float,  # 0.0 to 1.0
    'streak': int,       # Consecutive successes
    'difficulty': str,   # Current difficulty level
    'questions_answered': int
}
```

Action Space

- Easy (0)
- Medium (1)
- Hard (2)
- Expert (3)

Reward Function

```
R = α * success * difficulty_multiplier + β * streak_bonus - γ * failure_penalty
```

Demo Video

Watch the 10-minute demonstration

Key timestamps:

- 0:00 - Introduction
- 2:00 - Q-Learning demonstration
- 4:00 - Thompson Sampling demonstration
- 6:00 - Hybrid approach
- 8:00 - Results analysis

- 9:30 - Conclusions

Documentation

- Technical Report (PDF)
- System Architecture
- API Documentation
- Experiment Guide

Testing

Run unit tests:

```
bash

pytest tests/
```

Run integration tests:

```
bash

python tests/integration_test.py
```

Contributin

1. Fork the repository
2. Create your feature branch (`git checkout -b feature/AmazingFeature`)
3. Commit your changes (`git commit -m 'Add some AmazingFeature'`)
4. Push to the branch (`git push origin feature/AmazingFeature`)
5. Open a Pull Request

License

This project is licensed under the MIT License - see the LICENSE file for details.

Acknowledgments

- Course: Reinforcement Learning for Agentic AI Systems
- Institution: [Your University]
- Instructor: [Professor Name]

- Date: Fall 2024

Contact

Your Name - your.email@university.edu

Project Link: https://github.com/yourusername/rl-agentic-system

---

This project was developed as part of the take-home final for Reinforcement Learning for Agentic AI Systems.