

Reinforcement Learning for Algorithmic Trading: 6-Month Project Timeline

Phase 1: Foundations (Weeks 1-4)

Week 1-2: Finance Fundamentals

- Study basic financial concepts: stocks, markets, trading principles
- Learn about technical analysis and common trading strategies
- Resources: Online courses (e.g., Coursera, edX), finance textbooks

Week 3-4: Project Setup and Data Processing

- Set up development environment (Python, necessary libraries)
- Learn to access and process financial data (e.g., using yfinance or alpha_vantage API)
- Implement basic data visualization for stock prices

Phase 2: RL and Trading Basics (Weeks 5-8)

Week 5-6: Reinforcement Learning Fundamentals

- Study RL concepts: states, actions, rewards, policies
- Implement a simple Q-learning algorithm

Week 7-8: Basic Trading Environment

- Create a simple market simulation environment
- Implement a basic trading agent using Q-learning

Phase 3: Advanced RL and Trading Strategies (Weeks 9-16)

Week 9-10: Deep Q-Network (DQN)

- Implement DQN algorithm
- Apply DQN to the trading environment

Week 11-12: Trading Strategies

- Implement one or two basic trading strategies (e.g., moving average crossover)
- Integrate strategies with the RL agent

Week 13-14: Performance Metrics and Backtesting

- Implement key performance metrics (e.g., Sharpe ratio, max drawdown)
- Develop a basic backtesting framework

Week 15-16: Hyperparameter Optimization

- Implement a simple hyperparameter optimization technique (e.g., grid search)

- Optimize the RL agent's performance

Phase 4: UI Development and Integration (Weeks 17-20)

Week 17-18: Basic Dashboard

- Design and implement a simple web dashboard using React
- Display basic trading results and performance metrics

Week 19-20: Strategy Configuration Interface

- Create an interface for configuring trading parameters
- Integrate the UI with the trading system

Phase 5: Testing, Refinement, and Documentation (Weeks 21-24)

Week 21-22: System Testing

- Conduct thorough testing of all components
- Debug and refine the system

Week 23-24: Documentation and Final Touches

- Write comprehensive documentation
- Prepare a final project presentation

Feasibility Assessment

Given your strong technical background in computer science and AI, completing a basic version of this project in 6 months is challenging but potentially achievable. Here's why:

1. **Strong Programming Skills:** Your experience with Python, JavaScript, and full-stack development is a significant advantage.
2. **AI/ML Background:** Your coursework in AI and machine learning will help you grasp RL concepts more quickly.
3. **Project Experience:** Your previous projects demonstrate your ability to build complex systems, which is crucial for this project.
4. **Limited Finance Knowledge:** This is the main challenge. The timeline includes dedicated time for learning finance basics, but you may need to invest additional time to fully understand the trading aspects.

Recommendations:

1. Focus on implementing core functionality first. Start with a single asset (e.g., one stock) and a simple RL algorithm.
2. Simplify the trading strategies initially. You can add complexity as you learn more.
3. Leverage your strong web development skills to create a functional UI, but keep it simple at first.

4. Be prepared to adjust the timeline as you progress. Some areas may take longer than anticipated, especially those related to finance and trading.
5. Consider finding a mentor with finance experience to guide you through the domain-specific challenges.

This project is ambitious for a 6-month timeline, especially given your limited finance background. However, with dedicated effort and focus, you could create a basic working system within this timeframe. The key will be managing the scope carefully and being flexible with features based on your progress.