CS 584 Project Proposal

Project Title:

Finance Trading using Single-Agent Reinforcement Learning with Double Deep Q-Learning: An Exploration of Bullish and Bearish Engulfing Candlestick Patterns

Team Members:

Karteek varma Pericharla (A20543258)
Debasrita Sarkar (A20544597)
Harneet Kaur Dehiya (A20548613)

Description of the Problem:

The project focuses on using single-agent reinforcement learning to develop a trading strategy in the finance domain, specifically targeting stock trading. The goal is to create an intelligent agent that learns to make trading decisions based on historical market data and predefined rules, incorporating technical indicators such as weighted moving averages and candlestick patterns.

The agent will be trained using Double Deep Q-Learning (DDQN), a variant of deep reinforcement learning suitable for sequential decision-making tasks with large action spaces. The training data will consist of features such as open, close, high, low prices, volume traded, value of traded volume, and indicators like Weighted Moving Average (WMA).

In particular, the agent will be trained to recognize and act upon bullish and bearish engulfing candlestick patterns, which are widely recognized as strong reversal signals in technical analysis. The agent will consider additional conditions such as trend direction and the magnitude of the WMA compared to historical averages to validate the significance of these patterns.

Our project will contribute to the field of algorithmic trading by demonstrating the effectiveness of reinforcement learning in capturing complex market dynamics and exploiting profitable trading opportunities.

Brief Survey of Existing Work:

Traditional approaches to algorithmic trading have primarily relied on rule-based strategies, technical indicators, and simple machine learning algorithms. Rule-based strategies often involve predefined conditions for entering and exiting trades, such as moving average crossovers or support/resistance levels. While these strategies can be effective in certain market conditions, they often lack adaptability and struggle to capture complex patterns in the data.

In recent years, there has been a growing interest in applying machine learning techniques, particularly reinforcement learning, to algorithmic trading. Reinforcement learning offers the advantage of learning directly from data and dynamically adapting to changing market conditions.

Q-Learning, a foundational algorithm in reinforcement learning, has been applied to various domains, including finance, to learn optimal decision-making policies.

Deep Q-Learning (DQL), an extension of Q-Learning using deep neural networks, has shown remarkable success in mastering complex tasks, as demonstrated by the famous achievement of AlphaGo in playing the game of Go [1]. DQL algorithms, such as the Deep Q-Network (DQN), leverage deep neural networks to approximate the Q-function, enabling the agent to learn high-dimensional state-action value functions.

However, traditional DQN algorithms can suffer from overestimation bias, leading to suboptimal performance. To address this issue, Double Deep Q-Learning (DDQN) was proposed, which employs two separate neural networks to estimate the Q-values, mitigating the overestimation bias and improving stability and performance.

Despite the success of DQL algorithms in various domains, their application to financial trading presents unique challenges. Financial markets exhibit non-stationary and noisy behavior, making it difficult for traditional machine learning models to generalize effectively. Moreover, the effectiveness of technical indicators and trading patterns may vary across different market conditions and asset classes.

Frensi Zejnullahu, et al [2] explore the use of a DDQN algorithm for single-asset trading in the financial market. The research highlights the agent's ability to adapt to different conditions, such as incorporating transaction costs and adjusting behavior during market crises. Comparing the DDQN agent's performance to a "buy-and-hold" strategy also provides a valuable baseline for evaluation. While the paper focuses on the E-mini S&P 500 futures contract, exploring the generalizability of the DDQN approach across different asset classes and market conditions would be valuable. Comparing the DDQN agent's performance against other reinforcement learning algorithms or established trading strategies could also offer wider insights.

Proposed Approach:

In our proposed approach, we aim to leverage the power of reinforcement learning, specifically Double Deep Q-Learning, to develop a robust trading strategy capable of identifying and exploiting profitable opportunities in financial markets. Our focus will be on incorporating domain-specific knowledge, such as candlestick patterns and trend analysis, into the reinforcement learning framework to enhance the agent's decision-making capabilities.

We will design a custom reward function that incentivizes the agent to make profitable trades while penalizing losses, thereby encouraging risk management and prudent trading behavior. Additionally, we will integrate rules based on technical analysis principles, such as identifying bullish and bearish engulfing candlestick patterns, to guide the agent's actions in different market scenarios.

Preliminary Plan (Milestones):

1. Data Collection: Obtain historical market data including price and volume information, as well as calculated indicators like Weighted Moving Average.
2. Preprocessing: Normalize and preprocess the data for input into the reinforcement learning model.
3. Model Architecture: Implement a Double Deep Q-Learning (DDQN) agent to learn the optimal trading strategy.
4. Training: Train the DDQN agent on historical market data, emphasizing the identification and exploitation of bullish and bearish engulfing candlestick patterns.
5. Evaluation: Evaluate the trained agent's performance on historical and unseen data, assessing its ability to generate profitable trading decisions while managing risk.
6. Fine-Tuning: Fine-tune the agent's parameters and reward/punishment mechanisms to optimize its performance and adapt to changing market conditions.
7. Analysis and Reporting: Analyze the results and document the effectiveness of the proposed approach in comparison to baseline strategies and traditional trading methods.

References:

[1] Silver, D., Huang, A., Maddison, C. J., et al. (2016). Mastering the game of Go with deep neural networks and tree search. Nature, 529(7587), 484–489.

[2] Zejnullahu, F., Moser, M., & Osterrieder, J. (2022). Applications of reinforcement learning in finance -- Trading with a double deep Q-network [Working paper]. arXiv preprint arXiv:2206.14267.

[3] Watkins, C. J. C. H., & Dayan, P. (1992). Q-Learning. Machine Learning, 8(3-4), 279–292.

[4] Mnih, V., Kavukcuoglu, K., Silver, D., et al. (2015). Human-level control through deep reinforcement learning. Nature, 518(7540), 529–533.

[5] Van Hasselt, H., Guez, A., & Silver, D. (2016). Deep Reinforcement Learning with Double Q-learning. AAAI, 16, 2094–2100.

[6] Silver, D., Hubert, T., Schrittwieser, J., et al. (2018). A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. Science, 362(6419), 1140–1144.