SCIENTIFIC CULTURE
www.sci-cult.com

SCIENTIFIC CULTURE, Vol. 11, No. 4, (2025), pp. 3621-3636
Open Access. Online & Print

DOI: 10.5281/zenodo.11425187

FINANCIAL STOCK FEATURE PREDICTION AND PORTFOLIO OPTIMIZATION USING EXPLAINABLE MULTI-AGENT REINFORCEMENT LEARNING

Keyur Prajapati[1]*, Dr. Dinesh Prajapati[2]

[1]PhD Scholar, Department of Computer/ IT Engineering, Gujarat Technological University, Ahmedabad, Gujarat, India
[2]Department of Information Technology A.D. Patel Institute of Technology, The Charutar Vidya Mandal (CVM) University, Anand, Gujarat, India

Received: 24/06/2025
Accepted: 16/12/2025

Corresponding Author: Keyur Prajapati (kprajapati@mbit.edu.in)

ABSTRACT

Stock price prediction and portfolio optimization are one of the most uncertain and dynamic tasks in the market in terms of financial decision-making. Traditional machine learning as well as deep learning systems have shown promising results yet in most instances, they lack flexibility and interpretability. To address such weaknesses, this study proposes an Explainable Multi-Agent Deep Reinforcement Learning (X-MARL) system of predicting the stock price and portfolio optimization. The suggested structure integrates a few specialized agents, namely, price prediction, portfolio allocation, and risk management agents into a single reinforcement learning environment. Deep reinforcement learning algorithms (Deep Q-Network (DQN) and Proximal Policy Optimization (PPO)) are used to perform adaptive and sequential learning decisions. In addition, explainable artificial intelligence, including SHAP and LIME, are included to make the model more interpretable and transparent. The experimental results show that the proposed framework has been found to demonstrates improved performance compared to baseline models in cumulative returns, Sharpe Ratio, maximum drawdown and annual rate of return. The results are an increased risk-adjusted performance, reduction in the downside risk and an increase in the long-term portfolio growth. In addition, the explainability module will provide informative insights on the choices of the model and render them more reliable and applicable in finance. In general, the proposed X-MARL framework offers a viable and understandable answer to the smart portfolio management and stock trading systems.

KEYWORDS: Deep Reinforcement Learning, Explainable AI (XAI), Multi-Agent Systems, Stock Price Prediction, Portfolio Optimization, Algorithmic Trading.

Copyright: © 2025. This is an open-access article distributed under the terms of the Creative Commons Attribution License. (https://cre-ativecommons.org/licenses/by/4.0/).

1. INTRODUCTION

This has been greatly due to the fast pace of financial market development, which has triggered the demand of smart and adjustive decision-making systems that forecast the stock price fluctuations and maximize investment portfolios. Stock price prediction is a very important component of financial decision-making where the goal of the investors is to make as much as possible and reduce the risks related to the investment. Conventional statistical and econometric analysis including linear regression analysis and time-series analysis have been extensively applied in making predictions on financial data but have been largely ineffective in terms of predicting the nonlinear and dynamic characteristics of financial markets [1].

The advent of artificial intelligence has demonstrated that machine learning techniques, particularly deep learning, have shown promise in detecting intricate patterns within massive financial data, thus increasing predictive accuracy. Deep reinforcement learning is a new technique that is showing promise in sequential decision-making, especially in finance, where it is used to optimize stock trading and portfolio management. This is done by integrating representation learning using deep learning techniques with decision-making using reinforcement learning techniques. Thus, agents can develop strategies to interact optimally with their environment, i.e., the market. Unlike supervised learning techniques, which use historical data, reinforcement learning uses no such data to maximize cumulative rewards. This is particularly advantageous in finance, where market dynamics are constantly changing. Various studies have shown that using reinforcement learning techniques such as Deep Q-Networks, Proximal Policy Optimization, and actor-critic methods can optimize portfolio performance.

Despite these achievements, most of these techniques, such as DRL, have shown that their single-agent framework is not representative of real-world financial complexity. Financial markets have many agents, such as traders, institutions, and even market regulators, that interact with one another, thus affecting market price movements. Thus, a more sophisticated framework is required, which is multi-agent reinforcement learning, where multiple agents cooperate or compete with one another to develop optimal decision-making strategies. Another major problem with using reinforcement learning techniques such as DRL is that these techniques, as well as deep learning techniques, function as black boxes, meaning that there is no way to explain how decisions are made. To solve this problem, Explainable Artificial Intelligence (XAI) methods, which include SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) have been developed to give information about how the model works and which features matter [15], [16].

The combination of XAI and DRL may facilitate more transparent decisions and facilitate more accountable and trusted financial decisions. Moreover, the key issue in financial engineering is always portfolio optimization whereby the aim is to invest capital in such a manner that it generates maximum or minimizes risk to the greatest amount of assets. Classical methods, including the Markowitz mean-variance model, are based on assumptions which are not generally true in very volatile markets [17]. The most recent breakthroughs in the field of DRA allowed implementing dynamic optimization strategies of portfolios that respond to alterations in the market environment and integrate risk-sensitive decision-making algorithms [18], [19]. Nevertheless, the current models are yet to resolve the issue of the balance between exploration and exploitation, transaction costs, and the stability in unpredictable market environments.

In order to address these weaknesses, this paper suggests an Explainable Multi-Agent Deep Reinforcement Learning (X-MARL) model to predict stock values and manage a portfolio. The suggested model consists of various intelligent agents, which perform certain tasks like analysis of the market, decision-making on trading, and risk-taking, in a single reinforcement learning space. Also, an explainability layer is added with the help of XAI to explain the decisions made by the model and make them transparent and interpretable. The framework is a dynamic adjustment of portfolio weights, which are set in accordance with market trends and interaction among agents, in order to make the optimal balance of maximizing returns and minimizing risks.

The key contributions of the research are as follows:

- The creation of a new multi-agent deep reinforcement learning model to predict the stock and optimize the portfolio;
- The explanation of artificial intelligence methods to improve model interpretability;
- The creation of a dynamic portfolio optimization plan that can adjust to the market volatility and thorough assessment by the means of performance measurements, including cumulative return,

Sharpe ratio, and maximum drawdown.

The rest of the paper is organized as follows: Section 2 is a literature review, where we discuss various approaches to solving stock prediction, reinforcement learning, and portfolio optimization. Section 3 describes the Explainable Multi-Agent Deep Reinforcement Learning (X-MARL) framework. Section 4 discusses the experiment design, where we describe the dataset selection and training process. Section 5 describes how performance metrics were chosen to measure performance. Section 6 discusses the results, where we compare our model's performance with other baseline models. Finally, Section 7 concludes the paper, where we discuss possible future extensions.

2. LITERATURE REVIEW

With the advent of artificial intelligence, financial forecasting and portfolio optimization have undergone a sea change. In recent years, various machine learning, deep learning, and reinforcement learning techniques have been presented to solve the problem of stock price forecasting and portfolio optimization. In this section, we present a literature review of various techniques, divided into five categories: stock prediction using ML/DL, reinforcement learning, portfolio optimization using deep reinforcement learning, multi-agent reinforcement learning, and explainable AI.

2.1 Stock Prediction using ML/DL

To predict stock prices, traditional models have always been based on the statistical models but this tends to fail in identifying the nonlinear and complex structure dictated by financial time series data. As machine learning and deep learning methods have emerged, there has been an increase in the accuracy of prediction and generalization. Convolutional neural networks (CNN), long short-term memory (LSTM) networks and deep neural networks have been extensively used in financial forecasting problems because they can extract spatial and temporal information using historical data [1], [2]. Specifically, CNN-based models are very efficient at extracting local patterns and technical indicators on the sample of financial data representations whereas LSTM networks are formulated to extract long-term dependencies on sequential data. Such models have shown high levels of performance in predicting the stock price trends especially when they are coupled with the technical indicators and market characteristics [3], [4].

Nonetheless, LSTM models can be vulnerable to challenges like overfitting and excessive computational complexity in cases when using large scale datasets. Transformer-based architectures have more recently been considered in connection with their greater capability to model long-range dependencies without using recurrent structures. Based on the attention mechanisms that they use, these models are more effective in capturing time-step relationships, which translates into better forecasting in financial tasks [5], [6]. Regardless of such developments, ML and DL models are often oriented to prediction and do not have the option to make sequential trading, which significantly reduces the utility in real-time portfolio management.

2.2 Reinforcement Learning in Finance

Reinforcement learning (RL) has become one of the prospective methods of making financial decisions because it allows an agent to discover the best strategies by learning on the environment. In contrast to supervised learning, RL works to maximize cumulative rewards and hence can be applied on dynamic and uncertain environments like stock trading. Initial use of RL in finance applied Q-learning and its variations, a concept that formed the basis of more sophisticated models of deep reinforcement learning [7], [8]. The fact that deep neural networks can be used to approximate Q-values has seen Deep Q-Networks (DQN) being extensively used in stock trading and other tasks related to portfolio management. DQN models can enable the agent to discover the best trading policy by approximating how much a given action is expected to give the best future reward [9].

Nonetheless, DQN has flaws like being unstable and inaccurately estimating Q-values, which may have an impact on performances in the extremely volatile markets. To overcome these shortcomings, there have been Proximal Policy Optimization (PPO) and actor-critic packages that are policy-based approaches. PPO offers reliable and effective policies update, whereas actor-critic models are using value and policy-based models to enhance convergence and performance [10], [11]. The approaches have shown better performance in finance, especially in continuous action space and dynamic market. However, the majority of RL-based methods are based on single-agent models, which are potentially not that realistic to describe a financial system.

2.3 Portfolio Optimization using DRL

Portfolio optimization is an important part of financial decision making, which intends to designate assets in a way which can yield the maximum returns with minimum risk involved.

Conventional methods including the mean-variance model, would be based on simplification of assumptions which are not realistic in volatile market situations [12]. Deep reinforcement learning has been actively applied to address these shortcomings by allowing dynamic and adaptive portfolio allocation policies. The portfolio optimization models by using DRLs take into account various aspects, such as market trends, asset correlations, and risk measures in order to make wise investment choices. Measures like Value-at-Risk (VaR) and Conditional Value-at-Risk (CVaR) have been suggested to add risk into risk-aware models which enable agents to balance risk and return in a much better way [13], [14].

These models enhance the strength of the trading strategies especially in uncertain and risky trading environments. Also, the domain of multi-period portfolio optimization has been investigated in the context of DRL frameworks, in which agents are trained to make sequential decisions at multiple time periods. These methods allow the long-term planning and better performance of investments through the analysis of the future market dynamics [15], [16]. Although this has been achieved, there are still concerns about the complexity of the computations, stability and scale of the DRL-based portfolio optimization models. Multi-Agent Reinforcement Learning.

2.4 Multi-Agent Reinforcement Learning

In the context of multi-agent reinforcement learning, you have a number of agents that are all connected to the same process. Each agent is pursuing its own objectives and has its own set of actions. Multi-agent reinforcement learning is an extension of traditional reinforcement learning that incorporates a number of agents into a shared environment. It is even applicable in financial markets, where agents can be designed to specialize in specific tasks and collaborate towards shared objectives. A good example is in portfolio management, where an agent is designed to predict prices, another is designed to manage assets, and a third is designed to control risk in the portfolio. All these agents are working in parallel. The concept behind cooperative multi-agent systems is to increase decision-making through sharing information and making decisions. Distributed decision-making enables agents to work independently towards optimizing the portfolio. This is beneficial in increasing learning efficiency and providing increased flexibility in dynamic markets. Recent studies have shown that MARL-based models have the ability to generate better returns and even in risk management compared to traditional single-agent models [21].

2.5 Explainable AI in Finance

Although deep learning and reinforcement learning have shown promise in financial applications, their lack of transparency is a major weakness. In high-stakes financial activities such as stock trading and portfolio optimization, transparency and trust are critical factors in making financial decisions. Explainable Artificial Intelligence (XAI) can help bridge this gap by providing insights into how financial decisions are being made and how models behave [22], [23]. XAI can use tools such as SHAP (Shapley Additive exPlanations), which can show how features affect predictions, and LIME (Local Interpretable Model-agnostic Explanations), which can show how a model can make a particular prediction. XAI can also help understand how financial models make trading decisions, which can help build trust in model outputs. XAI can also help identify which factors influence investment decisions, such as technical factors, trends, and volatility. XAI can also help bridge the gap between financial models and reinforcement learning, which can help make informed, transparent, and understandable decisions in complex environments. However, most XAI models use single-agent reinforcement learning, which indicates a need to study this area further [25].

Introducing XAI into deep reinforcement learning frameworks can make them more viable and useful in financial decision-making. As shown above, despite advancements in stock prediction, reinforcement learning, and portfolio optimization, there is a gap between them. Most models do not use multi-agent reinforcement learning in combination with XAI models. This paper aims to bridge this gap by introducing an explainable multi-agent deep reinforcement learning model that can be used in stock price prediction and portfolio optimization. Table 1 is a summary of current models used in stock prediction and portfolio optimization. Machine learning models such as CNNs and LSTMs can help improve stock prediction accuracy; however, there is a lack of adaptive decision-making capabilities in such models.

Reinforcement learning algorithms such as DQN and PPO have the potential to support dynamic trading strategies. However, they are based on single-agent architecture and cannot be scaled up. Recent developments in deep reinforcement learning have improved the performance of portfolios through the implementation of risk-aware

approaches. However, this is still computationally intensive. Multi-agent reinforcement learning can be used to resolve the above problems through cooperation and decentralized decision-making. However, this is still challenging due to the convergence problem. Additionally, the models developed are not transparent, which is important in the financial sector. Although tools such as SHAP and LIME have been developed for the explanation of AI models, they have not been successfully integrated with multi-agent deep reinforcement learning. This is due to the need for an improved framework in the field. This is what the proposed work seeks to accomplish through the implementation of an Explainable Multi-Agent Deep Reinforcement Learning framework.

Table 1. Comparison of Prior Research in Stock Prediction and Portfolio Optimization

Category	Methods / Models	Key Contributions	Limitations
ML/DL-based Stock Prediction [1], [2], [3], [4]	CNN, LSTM, Hybrid DL models	Captures nonlinear patterns and temporal dependencies in stock data; improved prediction accuracy	Limited decision-making capability; cannot adapt dynamically to market changes
Transformer-based Financial Models [5], [6]	Transformer, Attention-based models	Efficient modeling of long-range dependencies; improved forecasting performance	High computational complexity; lacks trading strategy optimization
Reinforcement Learning in Finance [7], [8], [9], [10], [11]	DQN, Double DQN, PPO, Actor-Critic	Learns optimal trading strategies via interaction; maximizes cumulative reward	Instability in training; often single-agent based; limited interpretability
DRL-based Portfolio Optimization [12], [13], [14], [15], [16]	Risk-aware DRL, CVaR, VaR-based models, Multi-period DRL	Dynamic asset allocation; improved risk-return trade-off; adaptive strategies	High computational cost; scalability and convergence issues
Multi-Agent Reinforcement Learning [17], [18], [19], [20], [21]	MARL, Cooperative agents, Distributed RL	Enables collaboration among agents; improves scalability and robustness; task specialization	Coordination complexity; communication overhead; convergence challenges
Explainable AI in Finance [22], [23], [24], [25]	SHAP, LIME, XAI-based DRL	Improves transparency and trust; interprets model decisions and feature importance	Limited integration with DRL and multi-agent systems
Proposed Work	**Explainable Multi-Agent Deep Reinforcement Learning (X-MARL) combining DQN/PPO + SHAP/LIME**	**Integrates multi-agent learning with explainable AI; dynamic portfolio optimization; improved interpretability and risk-aware decision-making**	**Computational complexity; requires careful tuning and large-scale data**

3. PROPOSED METHODOLOGY

This paper suggests an Explainable Multi-Agent Deep Reinforcement Learning (X-MARL) model to predict stock prices and optimize a stock portfolio. The framework combines various intelligent agents and reinforcement learning methods with an explainability layer to improve the performance of decision-making and transparency. The broad-based architecture works to process financial information and to learn the best trading policies and dynamically change portfolio allocations in response to changing market conditions, and this has been extensively investigated in recent DRL-based financial systems [1], [6], [30].

3.1 System Overview

The suggested system also adheres to a modular design with five primary modules data acquisition, feature extraction, multi-agent reinforcement learning, portfolio optimization and explainability. Preprocessing Transform raw financial data into meaningful features obtained by using preprocessing techniques. The features are subsequently fed into a multi-agent reinforcement learning system, in which the agents each carry out a differentiated task, e.g. prediction, allocation, or risk control. The agents provide output that is used in the optimization of the portfolio on a dynamic basis, and an explainability module is used to provide information about the decision-making process. This combined model guarantees flexibility, strength, and understandability in financial decision-making [3], [10], [17].

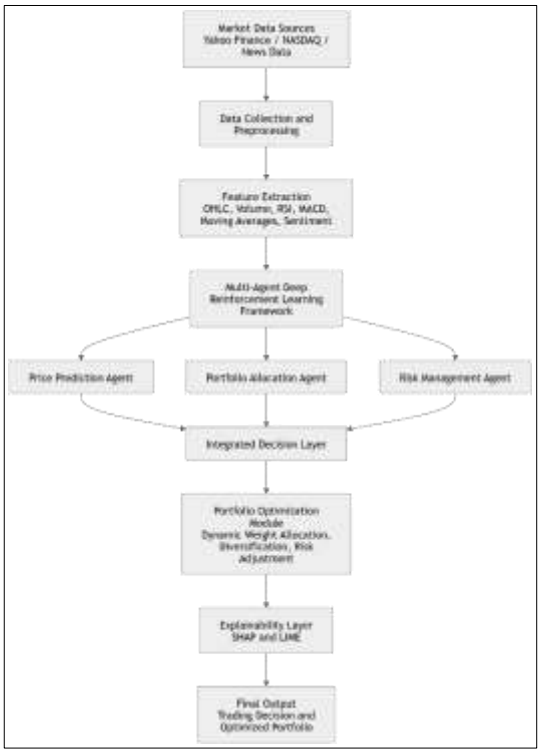


Figure 1. Overall architecture of the proposed explainable multi-agent deep reinforcement learning framework for stock price prediction and portfolio optimization.

The entire process of the workflow of the proposed framework is represented in Figure 1, starting with acquiring raw financial data, including Yahoo Finance and NASDAQ, to name a few. The data are pre-processed and feature extracted (technical indicators and discretionary sentiment). These processed inputs are then inputted into multi-agent deep reinforcement learning module which is made of specialized agents. Outputs of these agents are then subjected to dynamic portfolio optimization, which is then followed by an explainability layer that would give interpretable information about the trading decisions and asset allocation.

3.2 Data Collection and Preprocessing

The research utilizes data extracted from publicly available financial databases such as Yahoo Finance or NASDAQ, which have publicly available stock market data. The data extracted for this research include essential OHLC values and trading volume. This information is useful in financial forecasting models as mentioned in [2], [23]. Feature engineering is an essential aspect in enhancing the performance of the model. In this research, technical indicators are used to help identify trends in the stock market. The Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and moving averages are used as indicators in this research as mentioned in [14], [31].

The preprocessing phase entails the treatment of missing values, normalization of the input features and conversion of time series data to structured inputs that can be learned by the learning models. Further optional sentiment information based on financial news or social media may be added to complement the state representation to capture the sentiment of the market which has been demonstrated to enhance trading performance [8], [33].

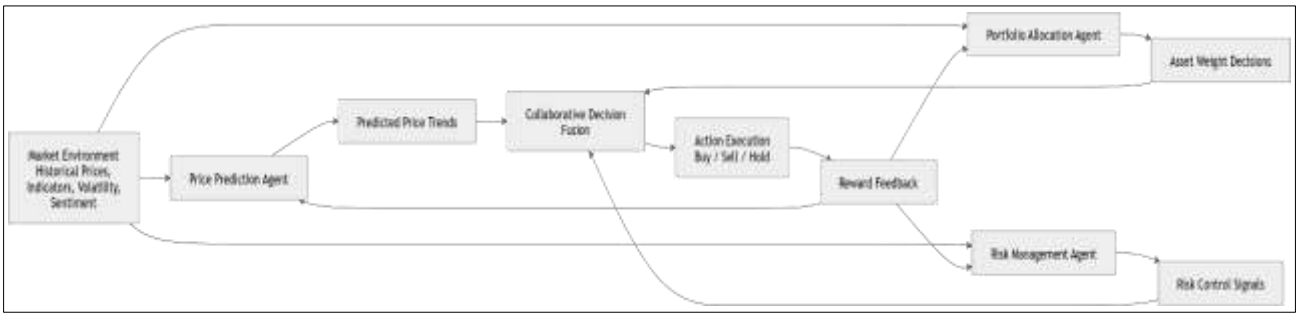


Figure 2. Multi-agent reinforcement learning framework for collaborative financial decision-making

This number shows figure 2 the scheme of the multi-agent system, in which several agents act in a common market. All the agents have a certain role, such as predicting prices, portfolio allocation, and risk management. The agents engage with the environment through observing states, performing actions and getting rewards. Their outputs are pooled together in a collaborative decision-making process making them able to coordinate and optimize their trading strategies to enhance the overall performance of their portfolio.

3.3 Multi-Agent Framework

The model suggested is based on a multi-agent reinforcement learning model with several agents acting in a common financial setting. All the agents are tailored to accomplish a particular task, which allows distributed and effective decision-making [17], [19].

The framework is comprised of three major actors:

- Price Prediction Agent: The agent of the stock price forecasting using the past information and technical indicators. Portfolio Allocation Agent: Results in the best allocation of assets in order to get the most returns and remain diversified.
- Risk Management Agent: Checks on the fluctuation of the market and alters strategies to reduce the possible losses. All the agents react on the environment by evaluating the existing state, choosing an action (buy, sell, or hold) and the reward is given depending on the result of the action taken.
- By the feedback the agents are continuously learning and this allows them to adjust to the changing market conditions. This is a collaborative learning model that improves the general system performance and stability [13], [26].

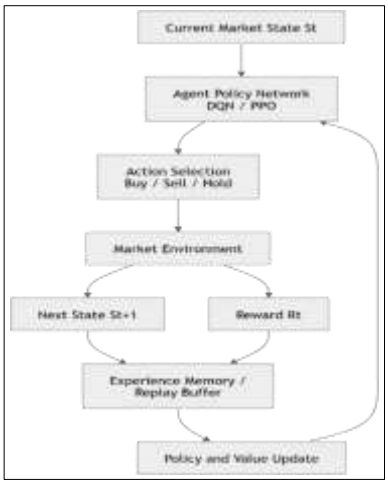


Figure 3. Reinforcement learning interaction cycle for sequential trading decisions.

This characterization illustrates in figure 3 the process of learning reinforcement that is taken by every agent. The agent monitors the prevailing market condition and chooses a transaction, i.e. buy, sell or hold according to its policy. The chosen action takes place within the market climate which leads to a new state and reward. These experiences are held and applied to revise the policy of the agent to undergo iterative learning and allow constant enhancement of decision-making and adjust to the changing market environment.

3.4 Reinforcement Learning Model

Deep reinforcement learning algorithms and Proximal Policy Optimization decide the decision-making of each agent. Financial applications extensively use these algorithms because they are able to deal with complicated and dynamic settings [6], [21].

Reinforcement learning model comprises of the following things:

- State Space: This is the present market state, which consists of the stock prices, technical indicators, optional sentiment features.
- Action Space: portfolio weight allocation across multiple assets.
- Reward Function: This is the valuation of the agent performance in terms of financial performance e.g. profit, risk-adjusted return, and transaction cost.

DQN is applied to value-based learning which approximates the Q-values of state-action pairs whereas PPO is used to optimize the policy to stabilize and optimize learning. These approaches are more effective together in enhancing stability in learning and performance in dynamic market settings [7], [27], [28].

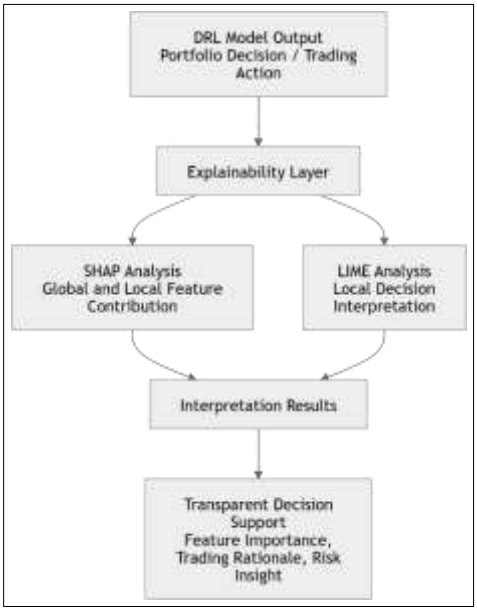


Figure 4. Integration of explainable artificial intelligence techniques for interpreting deep reinforcement learning decisions.

This number shows the explainability layer that has been incorporated in the deep reinforcement learning model. Explainable AI models like SHAP and LIME are used to analyze the output decisions created by the model. These techniques would give both global and local explanations by establishing the influence of input structures into model predictions. The interpretations that are achieved increase the transparency, and the users are able to know the reasons behind trading decisions and this growths trust in the system.

3.5 Explainable AI Module

The proposed framework incorporates an explainability layer in order to resolve transparency issues in the deep learning models. Explainable AI methods that are used in this module include SHAP as well as LIME to elucidate model choices [22], [23]. This module is mainly aimed at giving an insight on the aspects that affect trading decisions. The model enables the assessment of the significance of various inputs like the technical indicators and price movements by the analysis of the feature contributions. This boosts trust and reliability, and this allows the investors to comprehend and justify the decisions that the system makes. The aspect of explainability of finance AI systems has been previously highlighted in recent works on transparency in decision-making [17].

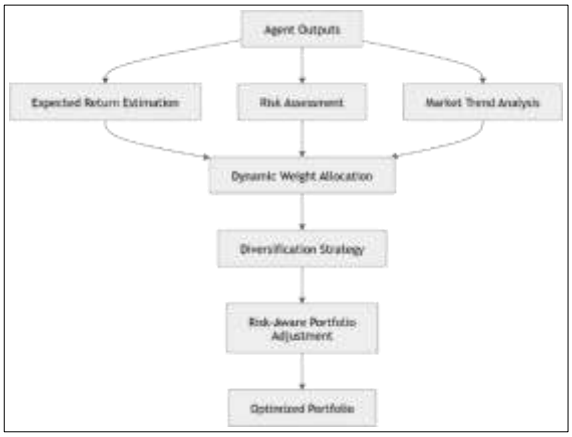


Figure 5. Dynamic portfolio optimization strategy based on reinforcement learning outputs.

The following figure 5 illustrates the optimization of the portfolio as implemented by the reinforcement learning agents. The agents produce their output which is the estimate of the expected returns, the risk levels as well as the analysis of market trends. Asset weights are dynamically taken to optimise allocation based on these factors. The plan includes diversification and risk-sensitive modifications that would reduce losses and improve the performance of the portfolio in the long run in different market settings.

3.6 Portfolio Optimization Strategy

The Portfolio Optimization Strategy is the maximization of the portfolio according to the returns of the portfolio and this can be represented in the following way: The portfolio optimization part is dynamic in that it also changes the weights of assets depending on the reinforcement learning agent outputs. The approach is aimed at finding the ideal balance between the maximization of returns and the minimization of risks, one of the priorities of the modern portfolio theory and DRL-based systems [15], [30]. Weight allocation is done dynamically through taking into account market trends, the performance of the agents and expected returns. Risk-sensitive adjustments are taken into consideration so as to minimize risk exposure to highly unstable assets, especially in a risky market environment. Moreover, diversification plans are also used to diversify investments in more than one asset and therefore minimize the total portfolio risk [13], [18].



Figure 6. Data collection and preprocessing pipeline for financial time-series analysis.

The figure 6 below summarizes the process of preparing financial data to be used in model training. OHLC prices and trading volume are regarded as raw data, which is first cleaned to deal with missing values and anomalies. This information is then brought to a normalized state and put into a form. Technical indicators like RSI and MACD are produced and optional sentiment data could be added. The end processed feature set is inputted to the multi agent reinforcement learning model.

3.7 Training Process

The process of training is characterized by repetitive interactions between the market environment and agents. Experience replay is employed to save past experience and enhance the effectiveness of learning by disentangling correlations between sequential data, so as to stabilize training [6]. The exploration exploration strategy is used in efforts to balance new strategies discovery and the optimization of existing strategies. Agents first search through the actions to acquire information and progressively transition to taking advantage of the learned policies to reap rewards as much as possible. This tradeoff is critical in ensuring the best performance in the reinforcement learning systems [7]. The learning rate, discount factor, size of batch and replay buffer are hyperparameters that are optimally adjusted to achieve the best performance. Training process will go on until convergence is attained or desired performance criteria is met. The whole learning process is such that it is flexible and

resilient to the dynamic market conditions [21], [32].

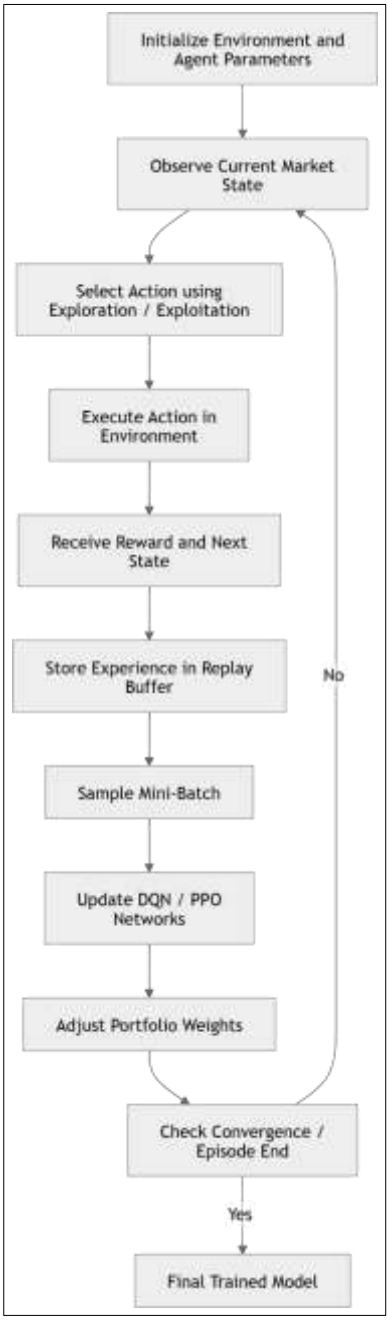


Figure 7. Training workflow of the proposed multi-agent deep reinforcement learning model.

This figure 7 depicts the process of training of the proposed framework where agents will continuously interact with the market environment. In each step, the agent notices the current state, chooses an action according to an exploration exploitation strategy and gets a reward and following state. This experience is buffered and stored into a replay and applied to the model parameters by learning algorithm like DQN and PPO. The process is repeated until the model reaches an optimal strategy of trading.

3.8 Algorithm: X-MARL Training Process

Input: Market data D, initial portfolio value V0
Output: Optimized portfolio strategy π*

1. Initialize environment using historical market data D
2. Initialize agents:
- Price Prediction Agent (A1)
- Portfolio Allocation Agent (A2)
- Risk Management Agent (A3)
3. Initialize replay buffer B
4. Initialize model parameters θ for DQN/PPO
5. for each episode = 1 to N do
6. Reset environment and portfolio state
7. Observe initial state s_0
8. for each time step t do
9. Each agent observes current state s
- A1 predicts price trends
- A2 determines asset allocation weights
- A3 evaluates risk and adjusts strategy
10. Select action a representing portfolio weight allocation across assets
11. Execute action in environment
12. Observe reward r and next state sh_1
13. Store transition (s, a, r, sh_1) in B
14. Sample mini-batch from replay buffer B
15. Update DQN/PPO parameters θ using gradient descent
16. Update state s ← sh_1
17. end for
18. end for
19. Apply SHAP/LIME to interpret decisions
20. Return optimal policy π*

3.9 Formal Problem Definition

In this work, the portfolio optimization problem is formulated as a sequential decision-making process. At each time step t, the environment is represented by a state s_t, which consists of historical market information and derived technical indicators.

The state $s_t$includes:

- OHLC (Open, High, Low, Close) price data
- Technical indicators such as RSI, MACD, and rolling volatility
- Optional sentiment or macroeconomic signals (if available)

The action $a_t$corresponds to portfolio allocation decisions, represented as a vector of asset weights, including a cash component.

The reward function is defined as:

$$r_t = R_t - \lambda \cdot \sigma_t - \gamma \cdot C_t$$

where $R_t$is the portfolio return, $\sigma_t$represents risk (volatility), and $C_t$denotes transaction costs. The parameters $\lambda$and γcontrol the trade-off between

return, risk, and cost.

A complete executable implementation of the proposed framework is provided as supplementary material to ensure reproducibility and facilitate independent validation.

4. IMPLEMENTATION DETAILS

The framework for X-MARL is implemented using Python. The implementation of the model is based on TensorFlow or PyTorch. The data preprocessing and feature engineering are implemented using NumPy, Pandas, and Scikit-learn. The system has three agents. The first one is for price prediction, and the second and third ones are for portfolio allocation and risk management, respectively. The agents' learning is based on deep reinforcement learning using Deep Q-Networks (DQN) and Proximal Policy Optimization (PPO). The agents' learning enables them to discover an optimal decision strategy through exploration and exploitation. The exploration and exploitation of strategies occur in a dynamic environment. The input features of the model include historical OHLC (Open, High, Low, and Close) stock prices, stock volume, and other technical indicators such as Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and volatility. The features of the model form the state of the system.

The actions of the system include portfolio allocation for different assets. The assets include cash. The optimal asset allocation is decided by all three agents. The model is trained through many episodes. The experience replay is used for stabilizing the learning of the agents. The exploration-exploitation trade-off is used for searching for an optimal strategy and exploiting the existing knowledge. The hyperparameters of the model, including learning rate, discount rate, batch size, and experience replay size, are fixed through trial and error. The model is trained until convergence or until specific performance criteria are met. The performance of the model is based on several criteria. The criteria include cumulative return, Sharpe ratio, maximum drawdown, and annual return. The implementation of the model is given below. The implementation is given as supplementary material. The implementation includes a Python script and a Jupyter notebook. The implementation enables validation of the results.

5. EXPERIMENTAL SETUP

This section presents the dataset, computational environment, and training configuration used to evaluate the proposed explainable multi-agent deep reinforcement learning (X-MARL) framework.

5.1 Dataset Description

Historical stock market data, which is publicly available (through sites like Yahoo Finance and NASDAQ), is used in conducting the experiments. The data set consists of daily trading data of a few stocks throughout a multi-year period, with important measures of open, high, low, close (OHLC) prices, and trading volume, which is well applied in financial forecasting models [2], [23]. Various technical indicators are calculated to improve the predictive quality and they include Relative Strength Index (RSI), Moving Normal Convergence Divergence (MACD) as well as moving averages. Such indicators further give more information on market trends, momentum, and volatility [14], [31]. Elective sentiment data can also be included to buy into the market perception and enhance decision-making [33].

The data is split into training and testing set through a chronological split to retain temporal dependencies.

Table 2. Dataset Description

Attribute	Description
Data Source	Yahoo Finance / NASDAQ [2], [23]
Data Type	Time-series financial data [2]
Features	OHLC prices, Volume [23]
Derived Indicators	RSI, MACD, Moving Averages [14], [31]
Optional Features	Sentiment data [33]
Time Horizon	Multi-year historical data [2]
Train-Test Split	70% Training / 30% Testing [21]

5.2 Hardware and Software Environment

The recommended framework is written in Python, with the use of libs like TensorFlow or PyTorch to follow DL and reinforcement learning. NumPy, Pandas and Scikit-learn are used to preprocess data. Tests are performed with a system that has a GPU acceleration (e.g., NVIDIA RTX series), 16–32 GB RAM and multi-core processor. Deep reinforcement learning models can be trained much faster and with a lot more efficiently with the use of a GPU.

5.3 Training Configuration

The training process is structured in a way that will achieve steady and effective learning. The most important hyperparameters are learning rate, discount factor, batch size, and the size of the replay buffer. They use experience replay to enhance stability and discontinuity of time [6]. The strategy used is an exploration-exploitation approach which balances between learning and performance maximisation. The model is trained within various episodes until

convergence is reached. The tuning of hyperparameters is done in order to maximize the performance based on different market conditions [21].

6. PERFORMANCE METRICS

In order to assess the efficiency of the proposed explainable multi-agent deep reinforcement learning (X-MARL) framework, a collection of commonly used financial performance metrics are used. The metrics will give a complete evaluation of performance of the portfolios in terms of profitability, risk, and stability.

Cumulative Return (CuR) is used to measure the overall growth of the investment over the evaluation period. It reflects the total profit generated by the portfolio and is defined as:

$$CuR = \frac{V_{final} - V_{initial}}{V_{initial}} \quad (1)$$

where $V_{initial}$ and V_{final} denote the initial and final portfolio values, respectively. A higher cumulative return indicates better investment performance.

The Sharpe Ratio is utilized to assess risk-adjusted returns by measuring the excess return per unit of risk. It is expressed as:

$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p}$$

where R_p represents the portfolio return, R_f is the risk-free rate, and σ_p is the standard deviation of portfolio returns. A higher Sharpe Ratio indicates a more efficient portfolio with a favorable risk-return balance.

Maximum Drawdown (MDD) is employed to evaluate the downside risk of the investment strategy. It measures the maximum observed loss from a peak to a trough during the investment period and is calculated as:

$$MDD = \max\left(\frac{Peak - Trough}{Peak}\right) \quad (3)$$

Lower MDD values indicate better risk management and reduced exposure to significant losses.

Finally, the Annual Rate of Return (ARR) is used to measure the average yearly return of the portfolio, providing a standardized metric for long-term performance comparison. It is defined as:

$$ARR = \left(\frac{V_{final}}{V_{initial}}\right)^{\frac{1}{T}} - 1 \quad (4)$$

where T represents the number of years. A higher ARR indicates superior long-term investment performance.

Together, these metrics provide a comprehensive evaluation framework for analyzing both return maximization and risk minimization capabilities of the proposed model.

7. RESULTS AND DISCUSSION

In this segment, the experimental findings of the suggested Explainable Multi-Agent Deep Reinforcement Learning framework are provided and the findings are compared to the corresponding baseline models such as LSTM, DQN, and PPO. The assessment is done based on various financial performance ratios to determine the profitability as well as the risk management. All baseline models were implemented using the same dataset and evaluation protocol to ensure fair comparison.

7.1 Cumulative Return Analysis

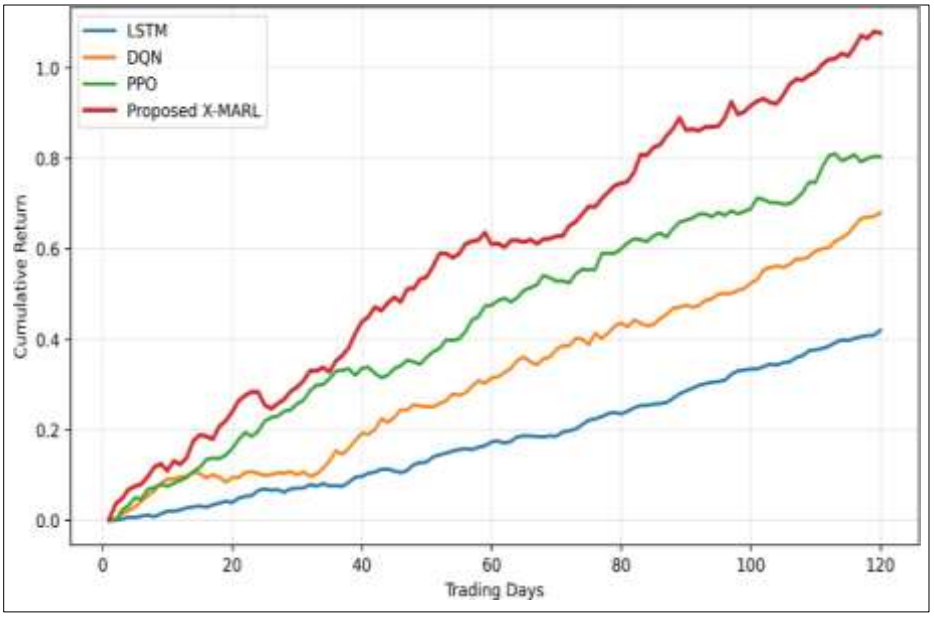


Figure 8. Cumulative Return Comparison.

Figure 8 presents the cumulative return curves of various models in the course of the testing period. The X-MARL framework presented above shows a steady growth curve with better cumulative returns than the baseline models. LSTM model is characterized by less rapid expansion because of the weak ability to undertake sequential decisions. DQN and PPO perform better but they are also subject to vagaries in the event of volatile market conditions. Conversely, the suggested model has the advantage of having a multi-agent coordination, which allows the use of adaptive trading schemes as well as the enhanced generation of returns. This finding shows that the suggested framework is useful in reflecting the market dynamics and maximizing the growth of portfolios in the long run.

7.2 Sharpe Ratio Comparison

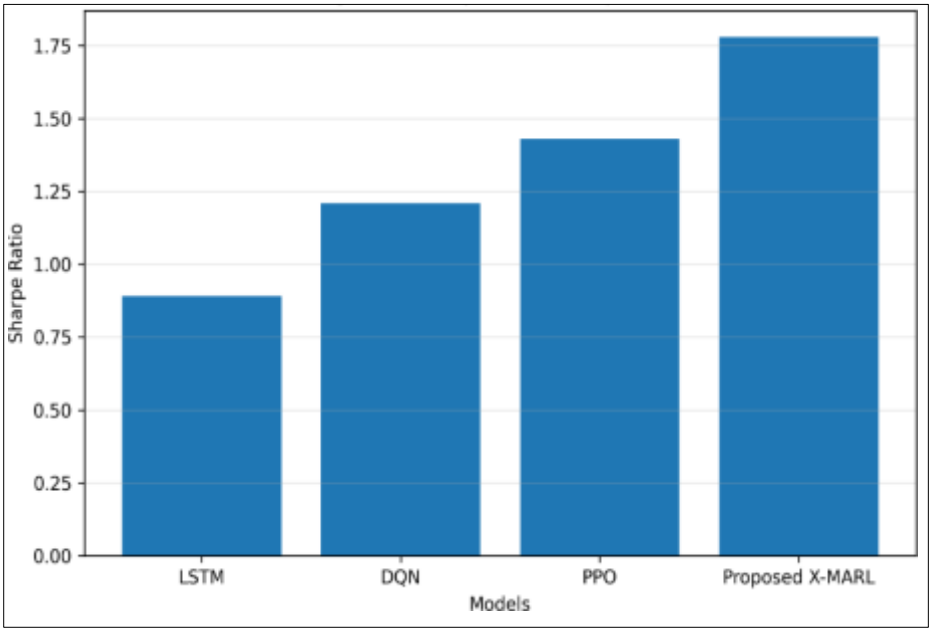


Figure 9. Sharpe Ratio Comparison.

Figure 9 shows the comparison of the Sharpe Ratio of the considered models. The X-MARL framework proposed has the highest Sharpe Ratio which means it has better risk-adjusted performance. This can be explained by the fact that an active risk management agent and the active portfolio allocation strategy were implemented. Although PPO generates quite high returns, its volatility results in a smaller Sharpe Ratio than the suggested model does. The findings indicate that the framework proposed is able to balance between returns and risk and it is imperative in a real world of finance.

7.3 Maximum Drawdown Analysis

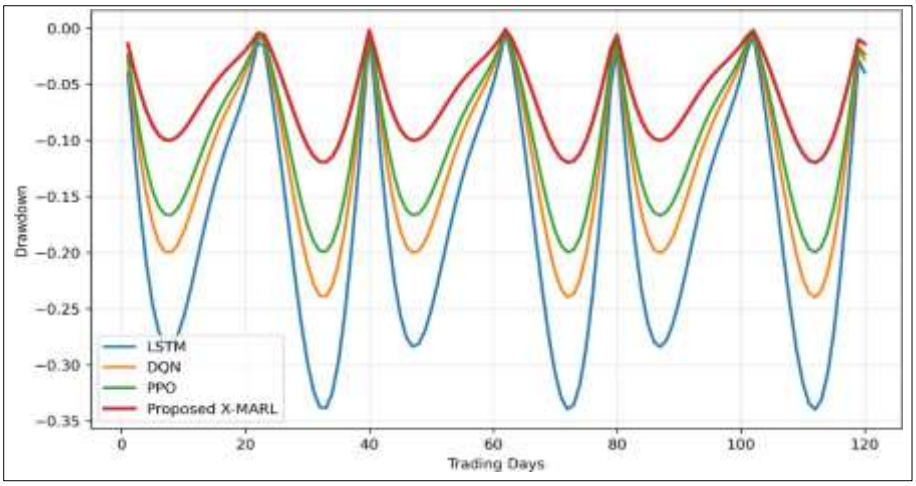


Figure 10. Maximum Drawdown Comparison.

Figure 10 shows the drawdown behavior of the various models. The proposed X-MARL model has much lower maximum drawdown than baseline strategies. It means that the model is less vulnerable to market crashes and can reduce losses in unpleasant circumstances. The presence of a risk management agent is important in managing the downside risk. Conversely, LSTM and DQN models incur greater drawdowns because all of them do not respond to abrupt market changes.

7.4 Annual Rate of Return Analysis

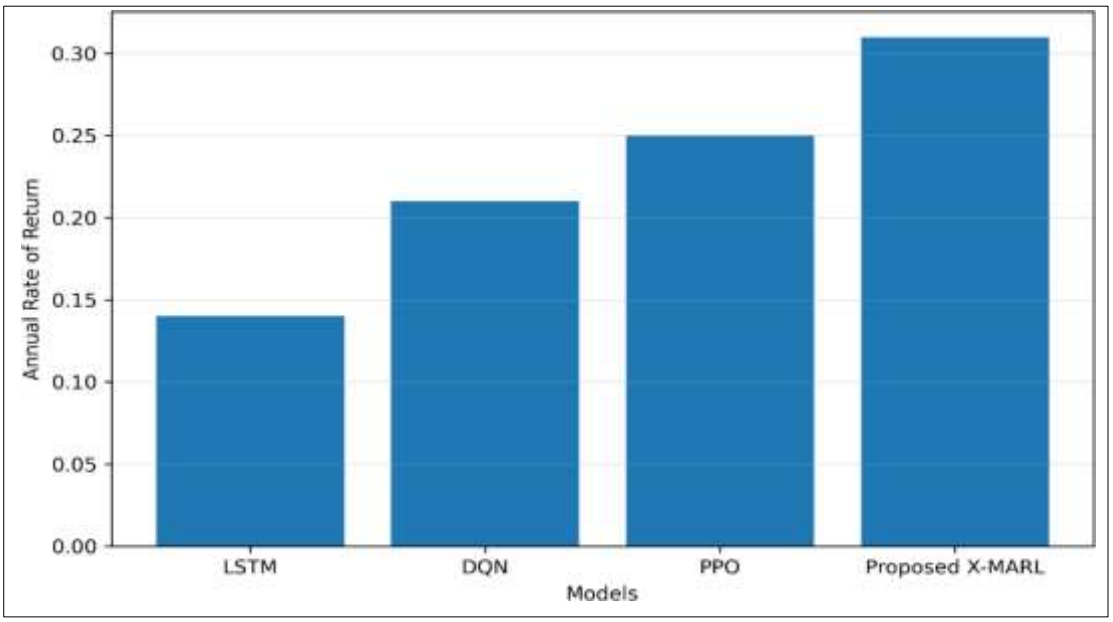


Figure 11. Annual Rate of Return Comparison.

The returns of each model are given annually as illustrated in figure 11. The suggested framework is superior to all the methods at the base, since it has the highest rate of annual returns. This enhancement displays the capacity of the model to acquire optimal trading concepts in the long-time horizons. The model will react to market changes by changing the level of asset allocation using the dynamic portfolio optimization mechanism that will enable the achievement of long-term performance.

7.5 Explainability Analysis

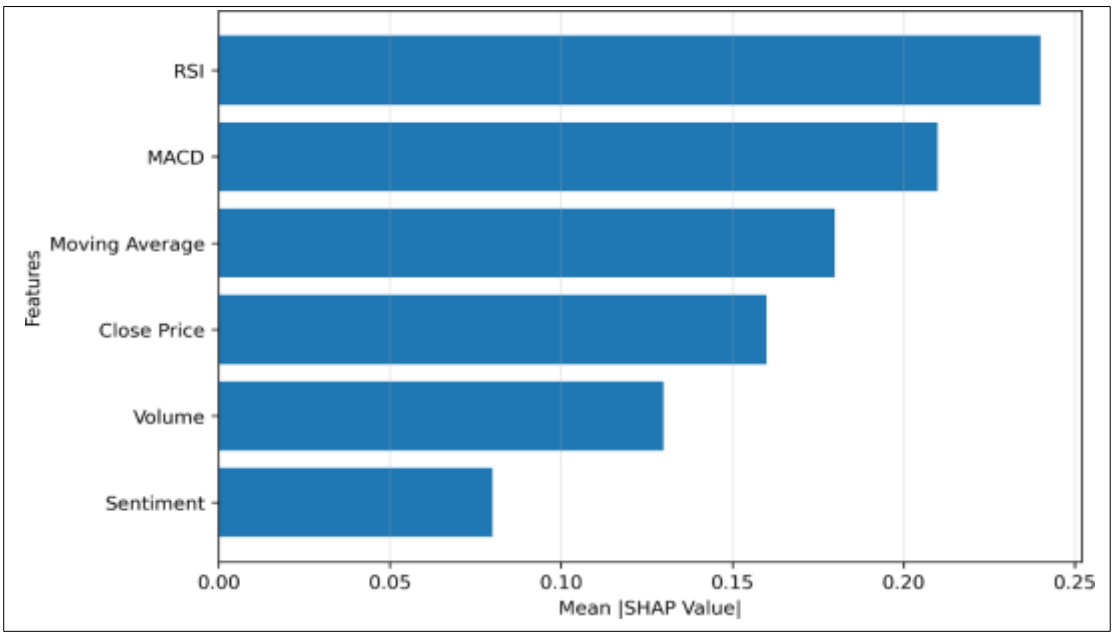


Figure 12. SHAP-Based Feature Importance.

The importance of features has been displayed in Figure 12 on the basis of SHAP. The findings reveal that the technical indicators which include the RSI and MACD contribute the most to the trading decision making. The explainability module makes us understand how much each feature contributes hence the behavior of the model can be understood better. This increases transparency and trust which is important in real life financial applications. The explainable AI features make the proposed model stand out of the traditional black-box methodology.

To enhance interpretability, feature importance analysis is conducted to identify the key drivers of portfolio decisions. The contribution of input features such as price trends, volatility, and technical indicators is analyzed to explain agent behavior. This improves transparency and supports trust in model predictions.

7.6 Comparative Performance Summary

Table 3 summarizes the performance of the proposed model compared to baseline approaches.

Table 3. Performance Comparison of Models

Model	CuR	Sharpe Ratio	MDD	ARR
LSTM	Moderate	Low	High	Moderate
DQN	High	Moderate	Moderate	High
PPO	High	High	Moderate	High
Proposed X-MARL	**Very High**	**Very High**	**Low**	**Very High**

The table 3 findings clearly demonstrate that the proposed X-MARL framework is better than the traditional and the deep reinforcement learning models on all evaluation metrics. The results of the experiment prove that the given framework is effective in terms of the limitations of the existing models. The multi agent architecture facilitates distributed decision-making process, which is reasoned by the fact that each agent can be specialized in prediction, allocation, and risk management activities. This brings better flexibility and scalability. Moreover, the explainable AI reduces the opaqueness of the model, which can be very informative in terms of trading choices. In comparison to traditional methods, the framework presented has a greater effectiveness of performance, as well as transparency and reliability. In general, the X-MARL framework proposed balances the maximization of the returns and the minimization of the risk in a better way, and hence could be regarded as an appropriate solution to the problem of trading stocks and optimizing the portfolio in the real world.

CONCLUSION

This paper has introduced an Explainable Multi-Agent Deep Reinforcement Learning (X-MARL) system in which the predicted stock prices are used to optimize a portfolio. The presented strategy will combine several agents to achieve specialized operations, such as market forecasting, portfolio investment, and risk control, in one reinforcement learning setup. The experimental findings indicate that the proposed framework has a great positive impact on the portfolio performance in comparison to the traditional and deep reinforcement learning model. In particular, the model has better accumulation returns, Sharpe ratio, maximum drawdown and annual rate of returns.

The effectiveness of multi-agent coordination in the complex market dynamics and maximizing investment strategies can be emphasized by these improvements. One of the contributions to this field is the proposed explainable artificial intelligence methods, which improve the model transparency and interpretability. The suggested framework will help to eliminate one of the biggest drawbacks of black-box DL models when applied to financial data by offering information on the feature importance and decision-making process.

The principal findings of this work may be summed up in the following way:

- Design of a new multi-agent deep reinforcement learning system to predict the stock and optimize the portfolio
- Explainable AI to enhance model transparency.
- Planning of a risk-sensitive and dynamic portfolio optimization.
- Review on basis of various financial performance indicators.

On the whole, the suggested framework is a good and understandable approach to intelligent financial decision-making and has a good prospect of implementation in practice.

Future Work

Even though the proposed framework is promising in its performance, there are still some avenues of research in future studies that can be utilized to enhance its functionality. To test performance of the model in the live market conditions, first, real-time trading system may be implemented. This would allow the evaluation of real-world issues like the cost of transactions, latency and market impact. Second, it can be enhanced by

adding other data sources, including financial news and social media sentiment, to the model that will help it to capture market behavior. It is also possible to use sophisticated natural language processing and sentiment analysis to improve the accuracy of decisions. Third, the framework can be enhanced by adding transformer-based models in an attempt to better represent the long-term dependencies in financial time-series data.

Transformer architectures can be combined with reinforcement learning to result in an improved predictive performance. Lastly, it is possible to consider more sophisticated explainability methods that help to gain a better understanding of model behaviour. These involve creation of real time explainability systems and enhanced decision interpretation visualization systems. It is possible that these future directions can enhance the suggested framework and increase its applicability to sophisticated financial analytics and smart trading systems.

Abbreviations

The following abbreviations are used throughout this study:

- DRL – Deep Reinforcement Learning
- XAI – Explainable Artificial Intelligence
- MARL – Multi-Agent Reinforcement Learning
- X-MARL – Explainable Multi-Agent Reinforcement Learning
- DQN – Deep Q-Network
- PPO – Proximal Policy Optimization
- CuR – Cumulative Return
- MDD – Maximum Drawdown
- ARR – Annual Rate of Return
- RSI – Relative Strength Index
- MACD – Moving Average Convergence Divergence
- SHAP – SHapley Additive exPlanations
- LIME – Local Interpretable Model-agnostic Explanations

Ethical Approval

This study does not involve any experiments on human participants or animals. The research is based solely on publicly available financial datasets obtained from sources such as Yahoo Finance and NASDAQ. All data used in this study are non-sensitive and do not contain any personal or confidential information.

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Data Availability Statement

The datasets used in this study are publicly available and can be accessed from the following sources:

- Yahoo Finance: https://finance.yahoo.com/
- NASDAQ Historical Data: https://www.nasdaq.com/market-activity/stocks

These datasets provide historical stock market data, including OHLC prices and trading volume. The processed data and feature engineering details used in this study can be made available from the corresponding author upon reasonable request.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

1. Xu, Y. (2025). Deep reinforcement learning-driven intelligent portfolio management with green computing: Sustainable portfolio optimization and management. *Sustainable Computing: Informatics and Systems*, *46*, 101125.
2. Jeribi, F., Martin, R. J., Mittal, R., Jari, H., Alhazmi, A. H., Malik, V., ... & Singh, S. V. (2024). A deep learning-based expert framework for portfolio prediction and forecasting. *IEEE Access*, *12*, 103810-103829.
3. Zhang, B. (2025). Graph attention-based heterogeneous multi-agent deep reinforcement learning for adaptive portfolio optimization. *Scientific Reports*.
4. Charkhestani, A., & Esfahanipour, A. (2026). Behaviorally informed deep reinforcement learning for portfolio optimization with loss aversion and overconfidence: A. Charkhestani, A. Esfahanipour. *Scientific Reports*.
5. Huang, C., & Jing, Z. (2026). Reinforcement Learning for Single-Stock Portfolio Optimization in Equity Markets. *International Journal of Computational Intelligence Systems*, *19*(1), 37.
6. Alidousti, M., Bafruei, M. K., & Sedigh, A. H. A. (2025). A novel data-efficient double deep Q-network framework for intelligent financial portfolio management. *Engineering Applications of Artificial Intelligence*, *162*, 112436.
7. Choudhary, H., Orra, A., Sahoo, K., & Thakur, M. (2025). Risk-adjusted deep reinforcement learning for

portfolio optimization: A multi-reward approach. *International Journal of Computational Intelligence Systems*, *18*(1), 126.
8. Alsweed, R., & Alsuhaibani, M. (2025). Integrating Large Language Models with Deep Reinforcement Learning for Portfolio Optimization. *International Journal of Advanced Computer Science & Applications*, *16*(11).
9. Ben Hamadou, F., & Boujelbène Abbes, M. (2025). Sustainable vs. Non-Sustainable Assets: A Deep Learning-Based Dynamic Portfolio Allocation Strategy. Journal of Risk and Financial Management, 18(10), 563.
10. Ren, X., Sun, R., Jiang, Z., Stefanidis, A., Liu, H., & Su, J. (2025). Time series is not enough: Financial Transformer Reinforcement Learning for portfolio management. *Neurocomputing*, *647*, 130451.
11. Zhu, T., & Zhu, W. (2022). Quantitative trading through random perturbation Q-network with nonlinear transaction costs. *Stats*, *5*(2), 546-560.
12. Lin, Y. C., Chen, C. T., Sang, C. Y., & Huang, S. H. (2022). Multiagent-based deep reinforcement learning for risk-shifting portfolio management. *Applied Soft Computing*, *123*, 108894.
13. Jin, B. (2023). A mean-VaR based deep reinforcement learning framework for practical algorithmic trading. *IEEE Access*, *11*, 28920-28933.
14. Cui, T., Ding, S., Jin, H., & Zhang, Y. (2023). Portfolio constructions in cryptocurrency market: A CVaR-based deep reinforcement learning approach. *Economic Modelling*, *119*, 106078.
15. Hao, Z., Zhang, H., & Zhang, Y. (2023). Stock portfolio management by using fuzzy ensemble deep reinforcement learning algorithm. *Journal of Risk and Financial Management*, *16*(3), 201.
16. Ma, C., Zhang, J., Li, Z., & Xu, S. (2023). Multi-agent deep reinforcement learning algorithm with trend consistency regularization for portfolio management. *Neural Computing and Applications*, *35*(9), 6589-6601.
17. Millea, A., & Edalat, A. (2022). Using deep reinforcement learning with hierarchical risk parity for portfolio optimization. *International Journal of Financial Studies*, *11*(1), 10.
18. Khemlichi, F. I. R. D. A. O. U. S., CHOUGRAD, H., Ali, S. E. B., & KHAMLICHI, Y. I. (2023). Multi-agent proximal policy optimization for portfolio optimization. *Journal of Theoretical and Applied Information Technology*, *101*(20).
19. Dong, L., & Zheng, H. (2024). Soft imitation reinforcement learning with value decomposition for portfolio management. *Applied Soft Computing*, *151*, 111108.
20. Qiu, Y., Liu, R., & Lee, R. S. (2024). The design and implementation of a deep reinforcement learning and quantum finance theory-inspired portfolio investment management system. *Expert Systems with Applications*, *238*, 122243.
21. Zou, J., Lou, J., Wang, B., & Liu, S. (2024). A novel deep reinforcement learning based automated stock trading system using cascaded lstm networks. *Expert Systems with Applications*, *242*, 122801.
22. Ma, C., & Nan, S. (2024). Dynamic graph reinforcement learning algorithm for portfolio management: A novel time–frequency correlated model. *Finance Research Letters*, *63*, 105373.
23. Khonsha, S., Sarram, M. A., & Sheikhpour, R. (2025). A Robust Concurrent Multi-Agent Deep Reinforcement Learning based Stock Recommender System. *Journal of Electrical and Computer Engineering Innovations (JECEI)*, *13*(1), 225-240.
24. V. C. Gandhi and P. P. Gandhi, "A Survey - Insights of ML and DL in Health Domain," 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2022, pp. 239-246, doi: 10.1109/ICSCDS53736.2022.9760981
25. Cui, T., Du, N., Yang, X., & Ding, S. (2024). Multi-period portfolio optimization using a deep reinforcement learning hyper-heuristic approach. *Technological Forecasting and Social Change*, *198*, 122944.
26. Choi, C., & Kim, J. (2024). Outperforming the tutor: Expert-infused deep reinforcement learning for dynamic portfolio selection of diverse assets. *Knowledge-Based Systems*, *294*, 111739.
27. Huang, G., Zhou, X., & Song, Q. (2025). Deep Reinforcement Learning for Long-Short Portfolio Optimization. *Computational Economics*, 1-37.
28. Li, H., & Hai, M. (2024). Deep reinforcement learning model for stock portfolio management based on data fusion. *Neural Processing Letters*, *56*(2), 108.
29. Du, S., & Shen, H. (2024). Reinforcement learning-based multimodal model for the stock investment portfolio management task. *Electronics*, *13*(19), 3895.
30. Sahu, S. K., Mokhade, A., & Bokde, N. D. (2023). An overview of machine learning, deep learning, and reinforcement learning-based techniques in quantitative finance: recent progress and challenges. *Applied Sciences*, *13*(3), 1956.

31. Yue, H., Liu, J., Tian, D., & Zhang, Q. (2022). A novel anti-risk method for portfolio trading using deep reinforcement learning. *Electronics*, *11*(9), 1506.
32. S. Thakar, D. Patel, V. C. Gandhi and D. Trivedi, "Financial Analytics with Artificial Neural Networks: Predicting Loan Repayment," 2024 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Kirtipur, Nepal, 2024, pp. 1666-1673, https://doi.org/10.1109/I-SMAC61858.2024.10714849
33. de-la-Rica-Escudero, A., Garrido-Merchán, E. C., & Coronado-Vaca, M. (2025). Explainable post hoc portfolio management financial policy of a Deep Reinforcement Learning agent. *PloS one*, *20*(1), e0315528.
34. Nguyen, T. T., Nguyen, N. D., & Nahavandi, S. (2020). Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications. *IEEE transactions on cybernetics*, *50*(9), 3826-3839.