

A STUDY ON PORTFOLIO OPTIMIZATION STRATEGIES USING HYBRID DEEP LEARNING AND REINFORCEMENT LEARNING APPROACHES

Dr. Dinesh Yadav¹

¹ Associate Professor, CSE Department, St. Andrews Institute of Technology & Management, Gurugram, Haryana, India
dinesh.yadav@saitm.ac.in

Abstract: A critical part of investing is to optimise the portfolio management, enabling the maximisation of profits whilst reducing risk. This study optimizes a portfolio using a hybrid deep learning model that incorporates Gated Recurrent Units (GRU) and Bidirectional Long Short-Term Memory (BiLSTM) networks, since the complexity of financial markets may outstrip the applicability of classical methods based on statistical models and historical data. The historical S&P 500 stock prices from October 2016 to 2023 were gathered and preprocessed using operations for missing value imputation, redundancy elimination, data denoising, normalisation, and one-hot encoding. The most significant predictors identified by feature importance analysis were momentum, liquidity, and volatility dynamics. The data were partitioned into training and test datasets, and the proposed hybrid model was evaluated against LSTM, XGBoost, and Linear Regression models. The hybrid GRU-BiLSTM model outperforms conventional approaches across assessment metrics such as R2 (95.80), RMSE (14.523), MAE (10.379), and MAPE (0.005). With improved accuracy and strength in stock price forecasting and portfolio optimization, the results validate the hybrid model's usefulness in both short-term and long-term time dependencies.

Keywords: Portfolio Optimization, Transaction Costs, Risk Management, Deep Learning, Reinforcement Learning.

1 INTRODUCTION

The optimization of a portfolio is an important aspect of financial engineering and risk management in the contemporary financial market. Its main focus is to make the most out of the returns and bring the risks under control with the rational distribution of different assets. The classical portfolio optimization models, These models are however, prone to various limitations when applied in practice such as assumptions of market efficiency, correct distribution of returns and insensitivity to changes in the dynamic market conditions [1][2][3]. The portfolio management process may be seen as both an information and an execution process. Investors want to capture the value that exists between the information's value and the cost of execution [4][5][6]. This aims at dynamic resource allocation so as to maximize portfolio returns, balance is a naturally difficult undertaking by investors.

Financial analysts and investment managers have long been interested in the most efficient use of portfolios as a tool for estimating the most profitable asset investments with the least amount of risk [7]. The mathematical foundation for the building of efficient portfolios that were on the efficient frontier was supplied by classical models such as Markowitz's Mean-Variance Optimization (MVO) and the Capital Asset Pricing Model (CAPM) [8][9]. However, these models oversimplify things by assuming things like regularly distributed returns and stable correlation, neither of which adequately account for the nonlinear, dynamic, and complex nature of modern financial markets.

Local optimization of investment portfolios always be part of fund management, as it leads investors to the best possible performance at minimum risk [10][11]. The classical methods of quantitative methods, especially the Mean-Variance model, developed by Markowitz, deal with the idea of adjusting between returns and risk using the diversification principle [12][13]. However, these conventional models are severely limited in their practical utility due to their inflexibility in dealing with the elements of rapid market changes and the continual correlations that characterize financial markets, both of which are non-linear and dynamic [14][15].

Machine learning (ML) provides analytical frameworks as a foundation, while deep learning (DL) enhances learning potential with its complex neural networks; both methods are finding applications in intelligent portfolio selection thanks to the advent of big data and AI [16][17]. Deep learning and reinforcement learning are setting a new standard for smart portfolio management. In such models, the DL components are used to extract features and predict the market and the RL agents are trained to adopt the best trading policies by means of rewards-based interactions with market environment [18][19]. DQN, PPO, and the Actor-Critic algorithms are algorithms that update the weights of their portfolios dynamically in order to maximize the long-term cumulative returns [20][21]. This predictive-based modeling and adaptive decision-making is an important development of the stationary optimization to self-learning autonomous financial systems.

1.1 Motivation and Contribution

Portfolio optimization is a highly important issue in the financial markets; proper forecasting of stock prices may increase investment choices and risk control to a large extent. Conventional models have difficulty in capturing intricate time-smoking patterns, and nonlinear connections of financial time-series information. This supports the idea of combining GRU and BiLSTM networks into a single deep learning model that can analyze both short-term changes and long-term relationships at the same time. The proposed framework, which combines momentum, liquidity, and volatility dynamics, delivers higher forecasting performance, provides more credible investment advice, and supports sounder portfolio optimisation decisions. The study has a number of major contributions as follows:

- Yahoo Finance has collected and prepared the S&P 500's daily closing values from October 17, 2016, through October 13, 2023.
- Strong preprocessing steps such as missing value, redundancy, data denoising, normalization, one-hot encoding to guarantee the high quality of input data.
- Offers a realistic model to optimise a portfolio through the combination of a sophisticated deep learning algorithm to better facilitate investment decision-making.
- Introduced a new hybrid GRU-BiLSTM structure, which integrates the performance of GRU with the BiLSTM bi-directional contextualising information in predicting stock prices better.
- R2, RMSE, MAE, and MAPE were some of the key performance indicators utilized to assess the model's accuracy and dependability in making predictions.

1.2 Justification and novelty

The justification behind this study is to have more accurate and reliable portfolio optimization techniques that are able to capture complex time patterns within financial markets. Conventional paradigms usually do not account for short-term changes or long-term dependence on stock price data. This work is novelty-based on the combination of GRU and BiLSTM networks into a hybrid network that takes advantage of the computational advantages of GRU and the bidirectionality of context cognition of BiLSTM. Moreover, the model uses momentum, volatility, and liquidity dynamics as additional characteristics, which improve predictive ability and interpretability. The hybrid method offers a powerful, scalable framework for financial forecasting that is more effective than traditional methods and can be used to make viable investment decisions.

1.3 Organization of the Paper

The following is the paper's structure: The study is organized as follows: Section 2 details previous research on deep learning methods and portfolio optimization; Section 3 describes the dataset and data preprocessing process; Section 4 presents the experimental results; Section 5 summarizes the key findings and suggests future research directions; and Section 6 concludes the study.

2 LITERATURE REVIEW

The research studies on Portfolio Optimization Strategies have been reviewed and analyzed in detail to inform and improve this study, and Table 1 presents a summary of recent work.

Geethanjali *et al.* (2025) The suggested approach takes these problems into account by using RL algorithms for accurate stock price forecasting and Deep Q-Networks (DQN) for real-time BH/SS decision-making. The model's adaptive performance tracks market fluctuations to maximize portfolio returns while minimizing risk. Because it simultaneously improves portfolio returns and manages real-time risk proportions, the hybrid model outperforms conventional methods in dealing with market swings. The LSTM model shows satisfactory effectiveness in identifying stock price correlations ($R^2 = 0.78$), whereas the DQN model optimizes trading decisions through cumulative reward [22].

Bhagwanrao and Bahadur Tiwari (2025) proposed system however, rely on neural networks for the process of pattern identification, natural language processing (NLP) for sentiment analysis and the upsurge of ensemble models for reducing the risk diversification strategies, it also helps in enhancing the speed, flexibility, and security of the same. This research helps develop the emerging group of intelligent investment systems that provide a scalable, adaptive, and secure platform which changes the criteria of the modern investment practices. The proposed system shows better performance than other methods with the maximum accuracy up to 96.2 %. It also provides the maximum value return of 93.5%, proving its predictivity and efficacy[23].

Giri and Singh (2025) conditions have caused troubles to the traditional portfolio management system, which is based on static models, to allocate sub optimally. Using Deep Reinforcement Learning (DRL), this research finds a way around these limitations by optimizing investment portfolios in real time. Three DRL methods, namely Policy Gradient (DDPG), Proximal Optimization of Policies (PPO), and Advantage Actor-Critic (A2C) are scrutinized specifically. In this research, the DDPG outperforms the others in terms of its annual return 23.1%, the PPO has the highest Sharpe ratio 0.998 for risk adjusted returns and A2C achieves long-term stability with balanced risk[24].

Zouaoui, Naas, 2025, An estimated mean square error (MSE) of 0.0218% was used to predict the ideal weights of optimal portfolios with an estimated return of 1.7239%, risk of 1.1219%, and a Sharpe index value of 1.5365% in LSTM network-based portfolio management strategies. In the volatile and complicated cryptocurrency market in particular, ML is a great tool for optimizing portfolios [25].

Sharma and Nagpal (2024) employ and examine three well-known DRL methods, namely DQN, PPO, and TD3 for enhancing the portfolio's performance. For performance evaluation, they use a comprehensive dataset of historical price data and other financial ratios accumulated. Among the DRL models considered here, TD3 recorded the highest total return of 145.8%, the highest risk-adjusted returns with a Sharpe ratio of 1.40, the maximum drawdown of 11.8%, and the lowest volatility of 11.5%. These outcomes reveal the efficiency and stability of the application of DRL approaches in financial portfolio management a new and efficient strategy for improving portfolio management for investors and financial institutions using better computational algorithms[26].

Lawton and Jeewa (2024) using different lengths of prior data in calculating these metrics. While none of the agents trained using the risk-incorporated reward functions outperformed the agents trained using the portfolio value reward function in average cumulative return, the agents trained using the risk-based functions showed lower standard deviations in cumulative return across all experiments, suggesting the use of these reward functions to be more stable approaches to portfolio optimization. best results show a 9.3% average annualized return using a Sharpe ratio reward function that uses a 15-day window of historical data for calculation and a 10.4% average annualized return using a portfolio variance reward function that uses a 5-day window of historical data[27].

Singh *et al.* (2024) employed for feature selection, dynamically assigning importance weights to key financial indicators and prioritizing relevant data points. Hyperparameters are fine-tuned for optimal performance. The model's effectiveness is evaluated through rigorous backtesting against historical data, with performance metrics such as annualized return, Sharpe ratio, and maximum drawdown compared to other indices. The DNN achieved a remarkable accuracy of 99.13%. Finally, the model is deployed in a live trading environment, where continuous monitoring and periodic retraining ensure adaptation to evolving market conditions and sustained portfolio optimization[28].

Research Gaps: Despite significant advancements in portfolio optimization using machine learning and deep reinforcement learning, several gaps remain. Most existing models rely heavily on historical price data and overlook the dynamic influence of macroeconomic factors and market sentiment. Many approaches focus on single-algorithm solutions, limiting their ability to capture both short-term fluctuations and long-term dependencies simultaneously. Computational complexity and high resource requirements restrict the scalability of hybrid and ensemble models. Risk management strategies are often simplified, ignoring real-time volatility and market shocks. Feature selection and interpretability of deep learning models remain challenges, reducing transparency for investors. Additionally, few studies explore live deployment and continuous adaptation of models in real-time trading environments. Addressing these gaps can enhance predictive accuracy, robustness, and practical applicability of portfolio optimization frameworks.

Table 1: Recent Studies on Portfolio Optimization Strategies Using Machine Learning

Author(s) & Year	Dataset Used	Key Findings	Challenges	Limitations	Future Work
Geethanjali <i>et al.</i> (2025)	Historical stock price data	Proposed a hybrid RL–LSTM model integrating DQN for real-time Buy/Hold/Sell decisions. Achieved $R^2 = 0.78$ with improved portfolio return and adaptive risk management.	Market volatility handling, real-time decision-making, and adaptive learning for fluctuating environments.	Model performance depends on cumulative rewards; limited generalization across multiple sectors.	Extend to multi-asset and cross-market learning frameworks for global trading optimization.
Bhagwanrao & Bahadur Tiwari (2025)	Financial market and sentiment datasets	Integrated Neural Networks and NLP for sentiment-based ensemble investment model achieving 96.2% accuracy and 93.5% return. Enhanced flexibility, scalability, and security.	Pattern identification and sentiment integration for diversified investment decision-making.	Focused primarily on sentiment data; may underperform with high-frequency trading scenarios.	Incorporate real-time news analytics and reinforcement learning for adaptive market behavior.
Giri & Singh (2025)	Stock portfolio datasets	Utilized DRL (A2C, DDPG, PPO) for real-time portfolio optimization. DDPG yielded 23.1% return; PPO had highest Sharpe ratio (0.998).	Overcoming static portfolio allocation inefficiencies in traditional systems.	Computational cost and training complexity of multiple DRL algorithms.	Combine multi-agent DRL for cooperative trading and dynamic asset rebalancing.

Zouaoui & Naas (2025)	Cryptocurrency market data	Applied LSTM for optimal portfolio weight prediction with low MSE (0.0218%), return (1.7239%), risk (1.1219%), Sharpe ratio (1.5365%).	Portfolio optimization in highly volatile crypto markets.	LSTM limited by sequential dependency; less responsive to abrupt market	Integrate hybrid LSTM–Attention models to improve interpretability and volatility adaptation.
Sharma & Nagpal (2024)	Historical price and financial ratio datasets	Evaluated DQN, PPO, TD3; TD3 achieved best performance with total return (145.8%), Sharpe ratio (1.40), drawdown (11.8%).	Enhance portfolio efficiency and reduce volatility using DRL-based trading.	Requires large historical datasets; may overfit to specific assets.	Implement federated DRL and transfer learning for broader generalization.
Lawton & Jeewa (2024)	Stock market data with variable time windows	Compared reward functions; risk-based reward improved stability. Achieved 9.3–10.4% annualized return depending on reward function.	Improving reward design for stability in portfolio optimization.	Limited by dependency on predefined window sizes; inconsistent generalization across timeframes.	Adaptive, self-tuning reward mechanisms for better robustness in changing markets.
Singh <i>et al.</i> (2024)	Historical financial data for backtesting	DNN with dynamic feature selection achieved 99.13% accuracy; validated via live trading with continuous retraining for sustained optimization.	Automated feature weighting and adaptive portfolio optimization.	High dependency on quality of financial indicators; limited interpretability.	Extend model with explainable AI (XAI) and reinforcement-based online retraining

3 RESEARCH METHODOLOGY

The proposed methodology for S&P 500 stock price prediction and portfolio optimisation presents an end-to-end machine learning framework that integrates predictive analytics with strategic asset allocation in Figure 1. The first step is extensive data preprocessing, including handling missing data points, eliminating redundant features, denoising, and applying one-hot encoding and Min-Max normalisation to make the data consistent. Once the importance of features has been determined, the dataset is divided into two parts of the training and testing set to apply a Hybrid GRU-BiLSTM model that considers both forward and backward time dependence to predict stock prices. These expected prices are then inputted to an optimization model based on a MPT that utilizes both a mean-variance analysis and optimization of Sharpe ratios to build diversified risk-adjusted portfolios based on S&P 500 assets. Statistical error measures such as MAPE, RMSE, MAE, and R^2 are used to evaluate model performance, whereas cumulative returns, volatility, maximum drawdown, and Sharpe ratio evaluate portfolio effectiveness, demonstrating the strength and validity of the suggested hybrid deep learning method for financial forecasting and optimal investment decision-making.

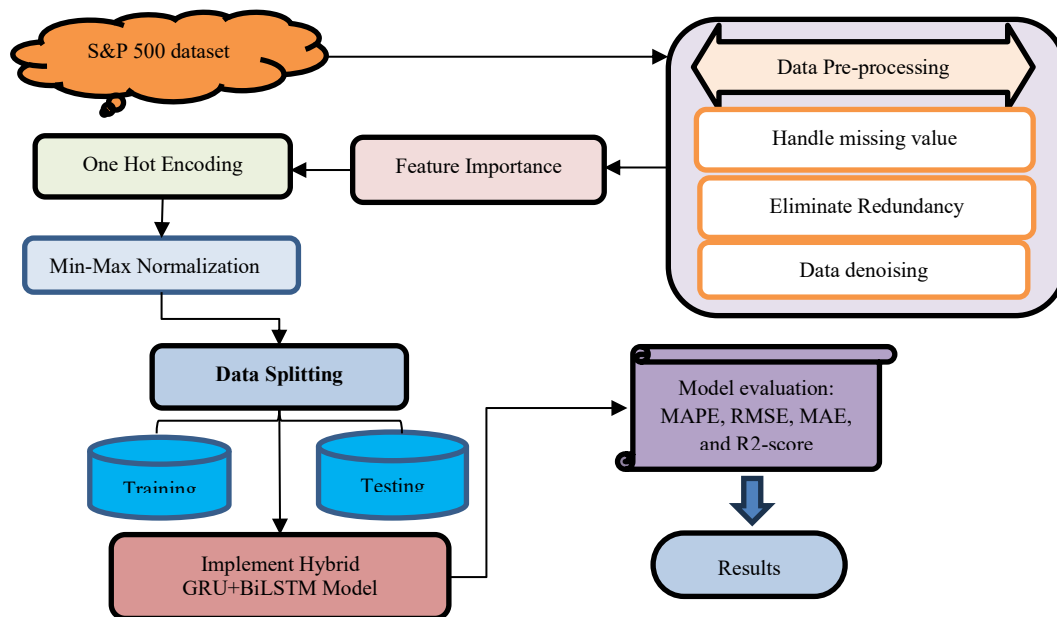


Figure 1: Proposed flowchart for portfolio Optimization Strategies Using Hybrid Deep Learning

The proposed methodology is detailed step-by-step in the section that follows:

3.1 Data Gathering and Analysis

The data used in this study came from Standard & Poor's 500 (S&P 500) historical prices on Yahoo Finance, covering October 13, 2023, to October 17, 2016. The dataset consists of daily closing stock prices, which serve as the model's dependent variable. Data visualizations such as bar plots and heatmaps were used to examine attack distribution, feature correlations, etc., are given below:

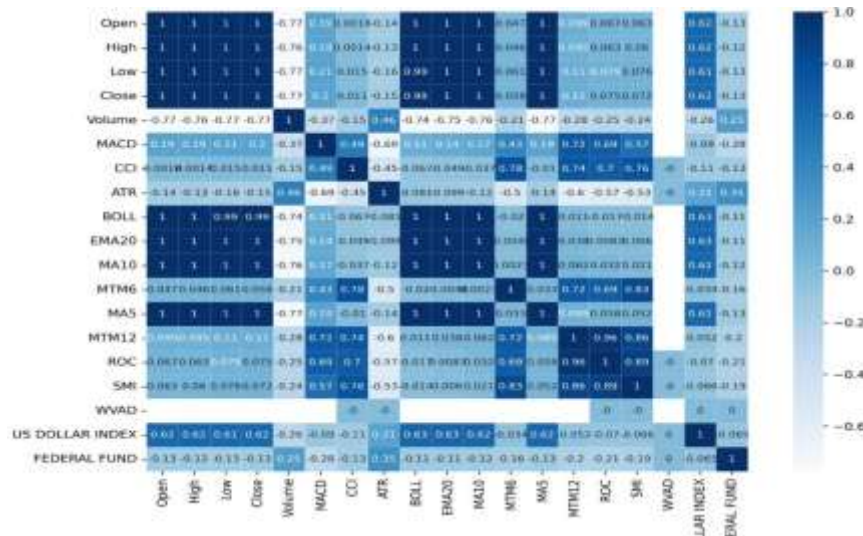


Figure 2: Correlation Matrix Heatmap on S&P 500 dataset for Portfolio Optimization Strategies

A correlation heatmap illustrating the pairwise relationships between S&P 500 stock market features and technical indicators in figure 2. Various variables such as Open, High, Low, Close prices, trading volume, moving averages (MA50, EMA20), technical indicators (CCI, ATR, BOLL, RSI, MACD), and macroeconomic factors (US Dollar Index, Federal Fund Rate) are shown in the color-coded matrix, which varies from dark blue (strong positive correlation, +1.0) to light blue (weak or negative correlation). This allows for the extraction of useful features and the removal of unnecessary ones.

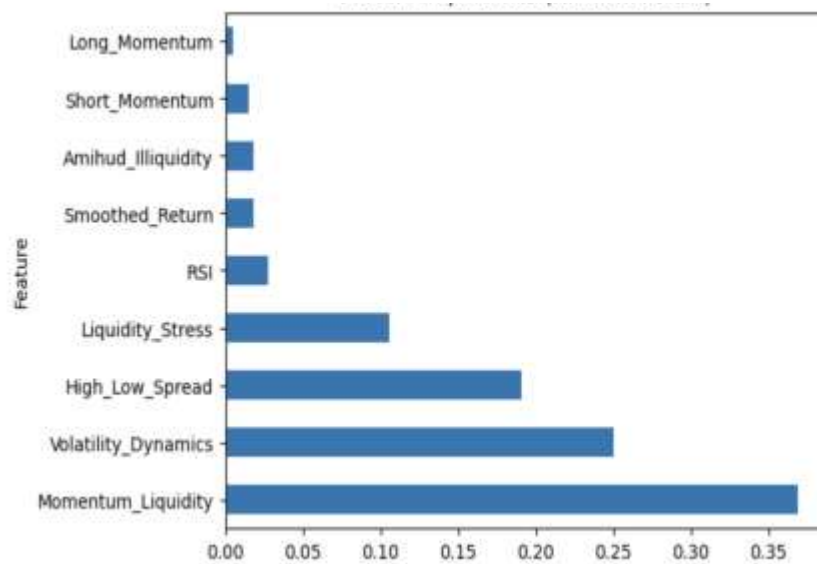


Figure 3: Feature Importance

Figure 3 illustrates the importance of various features in determining model performance. The most important feature that appears to be the most influential is Momentum Liquidity, then Volatility Dynamics, High Low Spread and Liquidity Stress, which are also the important features. At the same time, the RSI, Smoothed Return, Amihud_Illiquidity, Short Momentum, and Long Momentum are also comparatively less significant, meaning that they have a less significant effect on the total forecast.

3.2 Data Pre-processing

Data preparation involved concatenating, cleansing, and engineering features from S&P 500 data. The preprocessing stage entailed the management of missing data, eliminating redundancy, data denoising, and data leveling and normalization. The most important steps in preprocessing are summarized as follows:

- **Handle missing value:** A critical preprocessing step in ensuring data completeness and dependability is handling missing values. The entries that were missing were also recognized and addressed accordingly with the help of some techniques like a mean or median imputation to maintain the integrity of the data.
- **Eliminate Redundancy:** Elimination of redundancy entails finding and deleting redundant or highly correlated features that give redundant information. The model is more efficient and understandable because it holds unique and relevant variables.
- **Data denoising:** Data denoising is aimed at eliminating any undesired noise/variation in the data to ensure better signal quality. Such methods as smoothing, filtering or wavelet are utilized so that important patterns are retained and irrelevant variations are removed.

3.3 Feature Importance

Feature significance is used to assess each variable's contribution to the model's predictive capability, facilitating the evaluation of which variables exert a substantial influence on the output and which have minimal impact. Such an analysis does not only ensure better interpretation of the model but also helps select features in a manner that make the model more efficient and accurate. This makes it important to focus on features of the greatest influence like Momentum, Liquidity, and Volatility Dynamics without necessarily focusing on the other aspects that might not make a big impact on the model and trim down the model to make it simple and effective.

3.4 One Hot Encoding

The process of converting data into a different format in order to make it usable by processing, storage, or transmission is called data encoding, and may change categorical or text data into numerical forms comprehensible to machine learning models. One-hot encoding is used to encode data and is a method of data encoding, where categorically-coded variables are transformed into a binary, vector format, where each category is coded as a distinctive pattern of 0s and 1s.

3.6 Min-Max Normalization

The minmax method of normalizing the records was used to make sure that the values fell within range 0 to 1 so that the performance of the classifier could be improved and outliers would be minimized. In order to make the normalization, the following mathematical formula (1) was used:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

X_{min} represents the minimum value, X_{max} stands for the maximum value, X' stands for the normalized value, and X is the beginning value of the feature.

3.7 Data Splitting

The dataset was split into halves: 70% for training and parameter estimation, and 30% for testing and performance evaluation. This allowed us to check how well the model worked.

3.8 Classification equation of hybrid model of Bi-LSTM+GRU

3.8.1 Gated Recurrent Unit (GRU) Model

Training a model with an LSTM neural network takes longer because of the network's more intricate internal structure and the difficulty of adjusting its parameters. GRU, a streamlined LSTM variant. The GRU model provides similar predictive accuracy to the LSTM model while requiring less time to train. Because GRU merges the input and forget gates of LSTM into a single update gate, the memory module is simplified to only two gating components—the update gate and the reset gate structure—as seen in figure 4. This is the formula that the update gate uses to capture data, as shown in equation (2):

$$Z_t = \sigma(W_Z * [h_{t-1}, X_t]) \quad (2)$$

A critical component in determining the degree of historical data preservation is the reset gate, abbreviated as R_t . A smaller reset gate value indicates a greater capacity to store historical data.

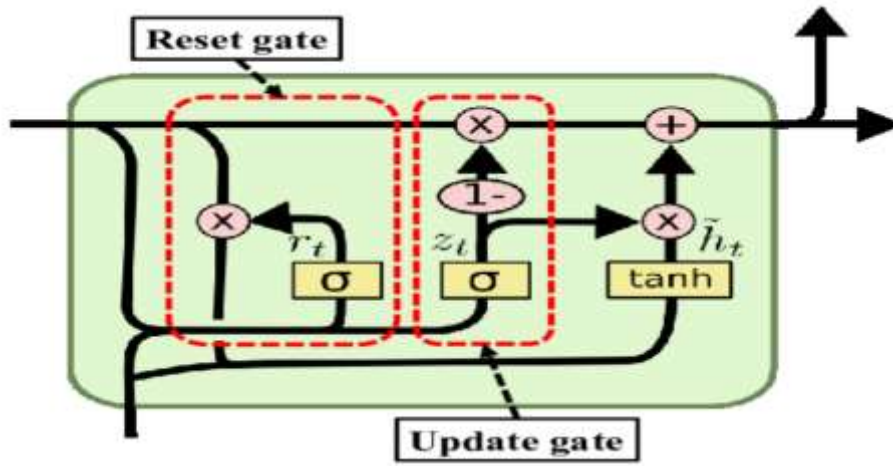


Figure 4: structure of GRU model

Here is the formula for the reset gate to obtain information: eqn (3):

$$R_t = \sigma(W_r * [h_{t-1}, X_t]) \quad (3)$$

Where h_t represents the unit's output state at time t and $\bar{h}_t$ stands for the assumed state value of h_t . The current unit's data is stored and delivered to the next unit using this value, while the forecasted value for the output at the prior instant is calculated using equation (4).

$$\bar{h}_t = \tanh(W_{\bar{h}} * [r_t * h_{t-1}, X_t]) \quad (4)$$

The predicted results of the water quality parameter data can be represented by equation (5).

$$h_t = (1 - Z_t) * h_{t-1} + Z_t * \bar{h}_t \quad (5)$$

The input data value at the current moment t is represented by X_t , the output value of the water quality parameter data in the memory cell at moment $t - 1$, is denoted by h_{t-1} , the weight matrices in the cell are W_z , W_r , and $W_{\bar{h}}$, "[" indicates the connection of two matrices, "*" shows the matrix product, σ is the activation function, and $\tanh$ is the bisecting curve of the activation function.

3.8.2 Bidirectional Long Short-Term Memory (Bi-LSTM) Model

The forward hidden layer in a Bi-LSTM transmits information from the past to the future, while the backward hidden layer performs the same function in the opposite direction. The data representation capabilities of Bi-LSTM are superior to those of standard LSTM when used to deep learning systems. The following explanations (6 to 8) provide light on the Bi-LSTM output:

$$h_t^f = LSTM(x_t, h_{t-1}^f) \quad (6)$$

$$h_t^b = LSTM(x_t, h_{t-1}^b) \quad (7)$$

$$y_t = W_o h_t + b_o \quad (8)$$

The output layer's bias vector is represented as b_o , the weights from the forward and backward layers are written as W_{hy}^f and W_{hy}^b , respectively. h_t^f and h_t^b are the elements that constitute h_t . When learning, Bi-LSTM makes use of both past and future data simultaneously, or "t."

3.8.3 Proposed Hybrid model of Bi-LSTM+GRU

This study's overarching goal is to optimize portfolios utilizing a BiLSTM and a hybrid recurrent unit (GRU) model based on DL. The recommended hybrid architecture combines the strengths of BiLSTM networks with GRU to manage sequential data with both short-term and long-term dependencies. GRUs are computationally and curb the vanishing gradient issue and BiLSTM is able to process the sequence forward and backward, improving contextual comprehension as well as forecasting precision. The hybrid method is specifically applicable to time-series forecasting and sequence modelling problems whereby the focus is on identifying the temporal patterns. GRU updates are characterised by eqn (9 to 12):

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (9)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (10)$$

$$\bar{h}_t = \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h) \quad (11)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \bar{h}_t \quad (12)$$

The GRU+BiLSTM hybrid model is a good compromise between computational capabilities and predictive performance, and thus is a solid option when working with sequential data. GRU component enhances faster learning and retains critical memory, and BiLSTM component preserves bi-directional relationships leading to be better at forecasting.

Layer (type)	Output Shape	Param #
Input (InputLayer)	(None, 60, 10)	0
BiLSTM (Bidirectional)	(None, 60, 128)	73,728
Dropout (Dropout)	(None, 60, 128)	0
GRU (GRU)	(None, 64)	37,440
Dense (Dense)	(None, 32)	2,080
Output (Dense)	(None, 1)	33
Total Parameters	—	113,281
Trainable Parameters	—	113,281
Non-trainable Parameters	—	0

Figure 5: model summary of Bi-LSTM+GRU

the Hybrid BiLSTM + GRU model proposed to optimize a portfolio, with the composition of the layers, the dimensions of the output, and the number of parameters presented in figure 5 The input layer of the model accepts time-series data with 10 features of 60 timesteps, and the bidirectional LSTM layer (73,728 parameters) is used to represent the contextual dependencies in both directions. A GRU layer (37,440 parameters) is added to prevent overfitting. It is computationally simple and can learn consecutively. Thereafter, the extracted representations are refined by a dense layer containing 2,080 parameters and the final output layer is used to produce the stock price prediction.

Experimental findings indicate that this architecture performs better than single GRU or LSTM models for error reduction and convergence rate.

3.9 Evaluation metrics

Model evaluation is one of the most important processes in machine learning projects because it provides information about a model's performance and makes results interpretable and presentable. Predicting the exact values of regression exercises isn't always easy, so the focus is on how close the predicted values are to the real values. The models were assessed in this research based on four performance measures, namely, R², MAE, RMSE, and MAPE.

3.9.1 R-Squared

R² is a statistic that determines the fitting of the regression model. R² can lie between 0 and 1 with the higher the value, the more the model fits the data. R² values between 0 and 1 indicate that the model adequately describes the data on the response around the mean, whereas R² values between 1 and 0 indicate that the model adequately describes all of the variability. To determine R², use the formula (13):

$$R^2 = \frac{\sum_{i=1}^n (y_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (13)$$

3.9.2 Mean Absolute Error (MAE)

MAE is a widely used metric for measuring the accuracy of a predictive model. It determines the mean value of error in a sequence of predictions, but does not take their direction into consideration. A smaller value of MAE indicates high performances. The equation of MAE calculation is (14):

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i^p| \quad (14)$$

Where,

Y is the actual value,

$\hat{Y}$ is an approximate value and n is the observations.

3.9.3 Root Mean Squared Error (RMSE)

This measure is a square root of MSE. RMSE measures the distance between predictions of a model and the reality. Reduces the level of RMSE values show improved model performance. The equation to establish the RMSE is (15):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (15)$$

3.9.4 Mean Absolute Percentage Error (MAPE)

MAPE is a calculation of errors based on percentages; it is an average percentage deviation between predictions and their target values in the data. The MAPE may also be regarded as the MAE that is returned in a percentage (16).

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left(\frac{y_i - \hat{y}_i}{y_i} \right) \times 100 \quad (16)$$

A combination of these measures provides information on the model's accuracy and forecasting efficiency.

4 RESULTS AND DISCUSSION

This section outlines the experimental design and presents the performance of the proposed model during both the training and testing phases. The stock market trend prediction model was conducted in an experimental environment based on a system with Intel(R) Xeon E3-1230 v6(3.50 GHz) processor and an NVIDIA Quadro M2000 gpu to perform the stock market trend forecasting. Table 2 summarizes the system's performance data, which included 16 GB of RAM, the CDW10 platform, and enough resources to accomplish deep learning tasks. Classification results of the suggested portfolio optimization strategies on the S&P 500 data show that the Hybrid GRU-BiLSTM model performs quite well. With a root-mean-squared error of 14.523 and a mean absolute error of 10.379, the model performed admirably in predicting the actual portfolio returns with few errors. Regarding the measured data, the Mean Absolute Percentage Error (MAPE) was 0.005, which is considered satisfactory. The model is highly successful and efficient in the role of portfolio optimisation, as indicated by the R² of 95.8 per cent, which further suggests that it can capture a large part of the data variation.

Table 2: Classification results of the proposed Portfolio Optimization Strategies using the S&P 500 dataset

Matrix	Hybrid GRU+BiLSTM Model
RMSE	14.523
MAE	10.379
MAPE	0.005
R ²	95.8

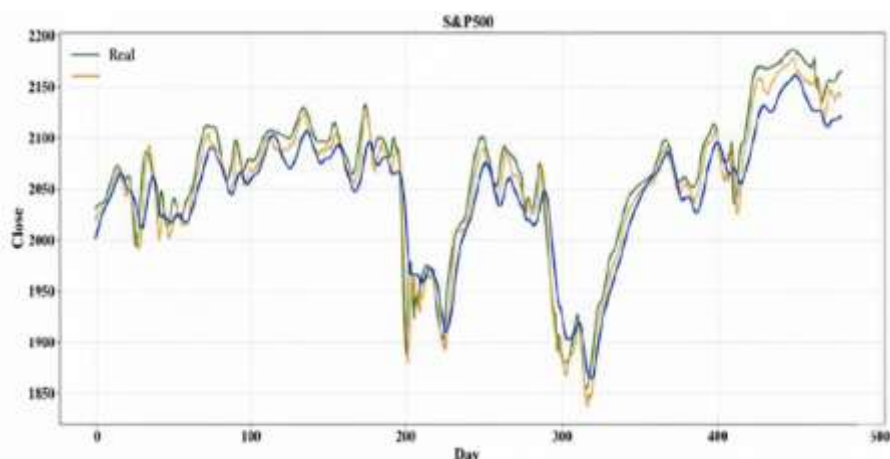


Figure 6: The line charts for the Hybrid GRU+BiLSTM Model

Figure 6 shows the actual performance of the S&P 500 index in 500 days against an estimate suggestive price of a model. In the Real data, there are notable volatility and a number of different periods: the first 200 days are characterized by an increase followed by stabilization, days 200-350 are illustrated by a sharp decline and recovery, and the last 500 days are marked by a high volatility and general growth in prices that reaches the level of 2200.



Figure 7: Loss curves for the Hybrid GRU+BiLSTM Model

The loss function converged to a model in 50 iterations. The loss is large (almost 0.00180) at the beginning but decreases rapidly in the early iterations and eventually levels off at a middle value around the second iteration. The second minor yet meaningful sharp decreasing trend is at approximately iteration 8 that pushes the loss to about 0.00145 in figure 7. Beyond this, the loss curve almost levels off and holds constant as the number of iterations continues to 50, which suggests that the optimization process has reached a minimum loss value.

4.1 Comparative analysis

Table 3 shows a comparative accuracy analysis of Hybrid GRU-BiLSTM model with the existing models to determine its usefulness. The purpose of this evaluation is to compare and contrast several deep learning and machine learning models that have been used to optimize portfolio strategies using the S&P 500 dataset. The results demonstrate that the traditional models, which comprise LSTM, XGBoost, and Linear Regression (LR), exhibit reasonable predictive accuracy with R2 values of 77%, 75.6%, and 79.9%, respectively. A far more accurate and trustworthy indication to use the model to optimize a portfolio would be the proposed Hybrid GRU-BiLSTM model, which outperformed these alternatives with a R 2 of 95.8, demonstrating higher capability to detect non-linear patterns in the data.

Table 3: Comparison of Different Machine learning and deep learning Models for Portfolio Optimization Strategies

Model	R2
LSTM[29]	77
XGBoost[30]	75.6
LR[31]	79.9
Proposed Hybrid GRU-BiLSTM model	95.8

The Hybrid GRU-BiLSTM model proposed has an R2 value of 95.8%, significantly higher than those of the conventional ML and single DL algorithms. The model improves the computational efficiency of GRU with the BiLSTM bidirectional time processing feature by effectively addressing both short- and long-term relationships in financial time-series datasets by means of their intricate, synergistic use, preventing overfitting and information loss and providing the model, with robust predictions and accuracy, to optimize a portfolio and make effective investment decisions.

5 CONCLUSION AND FUTURE STUDY

In the modern financial market environment, where markets can change rapidly, predetermined portfolio management methods cannot handle complex situations and volatility. practical application of DRL to construct intelligent and autonomous trading policies of the financial portfolio. The analysis of the results reveals that the proposed hybrid GRU-BiLSTM model is superior to the classic models, including LSTM, XGBoost, and Linear Regression, when estimating S&P 500 stock prices. The hybrid model has a high precision and reliability in long-term and short-term forecasting as it has a high R2 of 95.8, which represents both short-term and long-term temporal effects. This shows the merit of combining GRU and BiLSTM networks with sequential financial data,

highlighting the model's strength and potential to optimise portfolio strategies. Investor sentiment analysis, news analytics, and volatility indices will be incorporated into future work using natural language processing to increase prediction accuracy. Also, dynamic portfolio rebalancing using reinforcement learning and explainable AI methods will be adopted to develop autonomous, interpretable, and adaptive financial decisions.

REFERENCES

- [1] W. Wu, J. Chen, Z. Ben Yang, and M. L. Tindall, "A cross-sectional machine learning approach for hedge fund return prediction and selection," *Manage. Sci.*, 2021, doi: 10.1287/mnsc.2020.3696.
- [2] W. Zhuang, C. Chen, and G. Qiu, "A new deep reinforcement learning model for dynamic portfolio optimization," *J. Univ. Sci. Technol. China*, 2022, doi: 10.52396/JUSTC-2022-0072.
- [3] M. Guan and X. Y. Liu, "Explainable Deep Reinforcement Learning for Portfolio Management: An Empirical Approach," in *ICAIF 2021 - 2nd ACM International Conference on AI in Finance*, 2021. doi: 10.1145/3490354.3494415.
- [4] Y. C. Lin, C. T. Chen, C. Y. Sang, and S. H. Huang, "Multiagent-based deep reinforcement learning for risk-shifting portfolio management," *Appl. Soft Comput.*, 2022, doi: 10.1016/j.asoc.2022.108894.
- [5] S. Gu, B. Kelly, and D. Xiu, "Empirical Asset Pricing via Machine Learning," *Rev. Financ. Stud.*, 2020, doi: 10.1093/rfs/hhaa009.
- [6] M. Leippold, Q. Wang, and W. Zhou, "Machine learning in the Chinese stock market," *J. financ. econ.*, 2022, doi: 10.1016/j.jfineco.2021.08.017.
- [7] K. Erwin and A. Engelbrecht, "Meta-heuristics for portfolio optimization," *Soft Comput.*, 2023, doi: 10.1007/s00500-023-08177-x.
- [8] B. R. Ande, "Federated Learning and Explainable AI for Decentralized Fraud Detection in Financial Systems," *J. Inf. Syst. Eng. Manag.*, vol. 10, no. 35s, pp. 48–56, 2025.
- [9] S. K. Tiwari, "Quality Assurance Strategies in Developing High-Performance Financial Technology Solutions," *Int. J. data Sci. Mach. Learn.*, vol. 05, no. 01, pp. 323–335, 2025, doi: 10.55640/ijdsml-05-01-26.
- [10] S. R. Kurakula, "The Role of AI in Transforming Enterprise Systems Architecture for Financial Services Modernization," *J. Comput. Sci. Technol. Stud.*, vol. 7, no. 4, pp. 181–186, May 2025, doi: 10.32996/jcsts.2025.7.4.21.
- [11] T. Shah, "Leadership in digital transformation: Enhancing customer value through AI-driven innovation in financial services marketing," *Int. J. Sci. Res. Arch.*, vol. 15, no. 03, pp. 618–627, 2025.
- [12] A. M. Aboussalah and C. G. Lee, "Continuous control with Stacked Deep Dynamic Recurrent Reinforcement Learning for portfolio optimization," *Expert Syst. Appl.*, 2020, doi: 10.1016/j.eswa.2019.112891.
- [13] M. H. F. Afonso, J. V. de Souza, S. R. Ensslin, and L. Ensslin, "Como construir conhecimento sobre o tema de pesquisa? Aplicação do processo proknow-c na busca de literatura sobre avaliação do desenvolvimento sustentável," *Rev. Gestão Soc. e Ambient.*, 2012, doi: 10.24857/rgsa.v5i2.424.
- [14] S. Almahdi and S. Y. Yang, "An adaptive portfolio trading system: A risk-return portfolio optimization using recurrent reinforcement learning with expected maximum drawdown," *Expert Syst. Appl.*, 2017, doi: 10.1016/j.eswa.2017.06.023.
- [15] C. C. Cantarelli, B. Flybjerg, E. J. E. Molin, and B. van Wee, "Cost Overruns in Large-Scale Transport Infrastructure Projects," *Autom. Constr.*, 2018.
- [16] A. Parupalli and H. Kali, "An In-Depth Review of Cost Optimization Tactics in Multi-Cloud Frameworks," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 3, no. 5, pp. 1043–1052, 2023, doi: 10.48175/IJARST-11937Q.
- [17] V. Verma, "LSTM-based to Predicting Stock Market Trends Using Machine Learning and Sentiment Analysis," *Int. J. Res. Anal. Rev.*, vol. 8, no. 3, 2021.
- [18] H. Si, S. Kavadias, and C. Loch, "Managing innovation portfolios: From project selection to portfolio design," *Prod. Oper. Manag.*, 2022, doi: 10.1111/poms.13860.
- [19] Y. Zhang, P. Zhao, Q. Wu, B. Li, J. Huang, and M. Tan, "Cost-Sensitive Portfolio Selection via Deep Reinforcement Learning (Extended Abstract)," in *Proceedings - International Conference on Data Engineering*, 2023. doi: 10.1109/ICDE55515.2023.00312.
- [20] R. Q. Majumder, "Machine Learning for Predictive Analytics: Trends and Future Directions," *Int. J. Innov. Sci. Res. Technol.*, vol. 10, no. 04, pp. 3557–3564, 2025.
- [21] N. Malali, "The Impact of Digital Transformation on Annuities : Personalization , Investment Strategies , and Regulatory Challenges," *J. Glob. Res. Math. Arch.*, vol. 11, no. 12, pp. 1–7, 2024, doi: 10.5281/zenodo.15279540.
- [22] K. S. Geethanjali, S. Patil, P. K. R, and J. K, "Reinforcement Learning-Driven Portfolio Management: Adaptive Strategies for Risk and Returns Using LSTM and DQN," in *2025 International Conference on Electronics, AI and Computing (EAIC)*, 2025, pp. 1–6. doi: 10.1109/EAIC66483.2025.11101392.
- [23] N. N. Bhagwanrao and B. Bahadur Tiwari, "Optimizing Investment Approaches using AI-Driven Predictive Analysis and Cloud Analysis," in *2025 World Skills Conference on Universal Data Analytics and Sciences (WorldSUAS)*, 2025, pp. 1–6. doi: 10.1109/WorldSUAS66815.2025.11199193.
- [24] A. K. Giri and P. Singh, "Adaptive Portfolio Management in Volatile Markets Using Deep Reinforcement Learning," in *2025 3rd International Conference on Device Intelligence, Computing and Communication Technologies (DICCT)*, 2025, pp. 95–100. doi: 10.1109/DICCT64131.2025.10986527.
- [25] H. Zouaoui, M. Naas, and I. Articles, "Portfolio Optimization Based on MPT-LSTM Neural Networks : A case study of Cryptocurrency Markets," *Financ. Account. Bus. Anal.*, vol. 7, no. 1, pp. 82–98, 2025.
- [26] R. Sharma and M. Nagpal, "Deep Reinforcement Learning for Financial Portfolios: A New Approach to Adaptive Strategy

- Development,” in *2024 4th Asian Conference on Innovation in Technology (ASIANCON)*, 2024, pp. 1–4. doi: 10.1109/ASIANCON62057.2024.10838067.
- [27] S. R. Lawton and A. Jeewa, “Deep Reinforcement Learning for Automated Stock Trading Using Different Reward Schemes,” in *2024 IEEE International Conference on Future Machine Learning and Data Science (FMLDS)*, 2024, pp. 552–557. doi: 10.1109/FMLDS63805.2024.00101.
- [28] S. K. Singh, T. Sharma, K. Santosh, K. Reddy, J. P. Swagatha, and R. Saravanakumar, “Utilizing Deep Neural Networks for Portfolio Optimization in Financial Markets,” in *2024 International Conference on Intelligent Systems and Advanced Applications (ICISAA)*, 2024, pp. 1–5. doi: 10.1109/ICISAA62385.2024.10829330.
- [29] J. Kim, H. S. Kim, and S. Y. Choi, “Forecasting the S&P 500 Index Using Mathematical-Based Sentiment Analysis and Deep Learning Models: A FinBERT Transformer Model and LSTM,” *Axioms*, vol. 12, no. 9, 2023, doi: 10.3390/axioms12090835.
- [30] B. Shi, C. Tan, and Y. Yu, “Predicting the S&P 500 stock market with machine learning models,” *Appl. Comput. Eng.*, vol. 48, no. 1, pp. 255–261, 2024, doi: 10.54254/2755-2721/48/20241621.
- [31] J. Gu, X. Lin, S. Chen, and Y. Lu, “AI-Enhanced Factor Analysis for Predicting S & P 500 Stock Dynamics 1 Introduction”.