



Predicting Stock Performance from Financial Ratios Using Machine Learning

Prepared By

Name: Marwa O. Essa

ID No: 243002281

Submitted in fulfillment of the requirements for the degree Master of
Science in Artificial Intelligence

At

College of Informatics

Midocean University

UAE

Supervisor Name: Dr. Ashraf Elsayed

College of Informatics, Midocean University, Comoros

Acknowledgements

I extend my deepest gratitude to my supervisor, Dr. Ashraf Elsayed, for his invaluable assistance, unwavering encouragement, and insightful critiques throughout my research journey. His steadfast commitment to academic excellence and meticulous attention to detail have significantly shaped the quality and depth of this research.

I am also sincerely grateful to the Teaching Staff Members at the College of Informatics, Midocean University, whose guidance, resources, and encouragement have been invaluable to my academic development.

To my family, I express my deepest love and appreciation. Your unconditional love, patience, and sacrifices have been the foundation of my resilience and success. Without your constant encouragement, this achievement would not have been possible.

Abstract

Using machine learning for stock prediction has gained a lot of interest recently. Many studies show that machine learning algorithms can forecast stock movements effectively by using historical data. However, most current methods mainly focus on short-term predictions based on past stock prices and technical indicators, such as the RSI and MACD. In this research, we take a different approach. We use 41 quarters of financial statement data to explore the possibility of predicting long-term stock returns based on fundamental indicators. The model includes important financial ratios, such as profitability, leverage, efficiency, liquidity, and market value ratios, to create a predictive framework. We trained three machine learning algorithms: RF, LSTM, and GRU, to predict relative return and selected portfolios by selecting stocks with higher predicted relative return using these machine learning models. The selected portfolios outperformed the TASI benchmark in the Saudi equity market, achieving higher returns and portfolio values. The findings emphasize the advantages of employing financial ratios in predicting the top 10 high-returning stocks in a mid-to-long-term investment situation, achieving over 66% higher excess returns for all models with lower risk profiles. Also, the selection of the top 5 high-return stocks achieved over 79% excess returns, proving the effectiveness of applying ML techniques based on financial ratios for stock prediction. In addition to financial ratios, the research also incorporates technical indicators derived from OCHLV data and using LSTM model to enhance the model's ability to capture short-term market dynamics. The model demonstrates strong predictive performance, achieving a low MAE of 0.0238 and a high coefficient of determination R^2 of 0.97, confirming the model's high predictive accuracy on technical features.

List of Publications

Table of Contents

Acknowledgements	iii
Abstract.....	v
List of Publications	v
Table of Contents	vi
List of Abbreviations	viii
List of Figures.....	ix
List of Tables	ix
Chapter 1 Introduction	1
1.1 Problem Definition.....	1
1.2. Thesis Objectives	1
1.3 Thesis Contribution.....	1
1.4 Thesis Organization	2
Chapter 2 Overview.....	3
2.1 Introduction	3
2.2 Essential Concepts	4
2.2.1 Random Forest (RF).....	4
2.2.2 Long Short-Term Memory (LSTM)	5
2.2.3 Gate Recurrent Unit (GRU)	8
Chapter 3 Related Work.....	10
3.1 Introduction	10
3.2 Related Work	10
Chapter 4 Proposed Approach.....	13
4.1 METHODOLOGY.....	13
4.1.1 Data Description.....	13
4.1.2 Financial Ratios.....	14
4.1.3 Data Preprocessing.....	15

4.1.4 Calculation of the Target Variable – Relative Return	16
4.1.5 Data Split and Moving Time Period	17
4.1.6 Evaluation Metrics	18
4.1.7 Training	20
Chapter 5 Experimental Results and Discussion.....	24
5.1 Results	24
5.1.1 High Portfolio Scores.....	24
5.1.2 High Excess Return.....	25
5.1.3 Top-k Precision	25
5.1.4 Predict Stock Price	26
Chapter 6 Conclusion and Future Work.....	27
6.1 Conclusion.....	27
6.2 Future Work	28
References	29

List of Abbreviations

- **Acc:** Accuracy
- **EBIT:** Earnings Before Interest and Taxes
- **EBITA:** Earnings Before Interest, Taxes, and Amortization
- **ER:** Excess Return
- **GRU:** Gated Recurrent Unit
- **LSTM:** Long Short-Term Memory
- **MACD:** Moving Average Convergence Divergence
- **MAE:** Mean Absolute Error
- **MSE:** Mean Square Error
- **ML:** Machine Learning
- **OHLCV:** Open, High, Low, Close, Volume
- **PS:** Portfolio Scores
- **RNNs:** Recurrent Neural Networks
- **ROA:** Return on Assets
- **ROC:** Return on Capital
- **ROCE:** Return on Common Equity
- **ROE:** Return on Equity
- **RF:** Random Forest
- **RSI:** Relative Strength Index
- **SGD:** Stochastic Gradient Descent
- **SVM:** Support Vector Machines
- **TASI:** Tadawul All Share Index
- **TRNN:** Time-series-based Recurrent Neural Network

List of Figures

2.1	Random Forest	21
2.2	The architecture of LSTM	23
2.3	The architecture of GRU	23
4.1	Rolling Window Data Splitting for Random Forest Training	23
4.2	Rolling Window Data Splitting for LSTM and GRU Training	23
4.4	Importance of features of RF model.....	23
5.1	Predicted vs Actual Price using OCHLV.....	23

List of Tables

4.1	13 Financial Ratios	21
4.2	Hyperparameters of LSTM model	23
4.3	Hyperparameters of GRU model	23
5.1	Models Evaluation	23
5.2	Evaluation of LSTM Performance on OCHLV Data	23

Chapter 1 Introduction

1.1 Problem Definition

Despite the increasing use of artificial intelligence in financial markets, most stock prediction models focus mainly on technical analysis. They often overlook the valuable insights found in corporate financial statements. Traditional machine learning models also have difficulty capturing complex, non-linear relationships between fundamental financial indicators and stock price behavior. This research aims to fill this gap by creating a machine learning-based model that predicts stock trends using fundamental financial data. Specifically, the research looks at how key profitability ratios, such as ROA, ROC, ROE, and ROCE, affect stock price movements. The main goal is to improve the accuracy, clarity, and reliability of long-term stock price predictions by using AI-driven methods.

1.2. Thesis Objectives

This research aims to develop a new relative-return-based stock selection technique based on financial statement data and machine learning methods. Furthermore, it integrates stock price forecasting as a secondary component for the sake of facilitating informed decision-making protocols. In general, the prime emphasis lies in facilitating financial analysts and investors in developing their decision-making process through a data-centric process for evaluating stocks for medium- and long-term horizons.

1.3 Thesis Contribution

Evaluating machine learning methods for medium- and long-term stock prediction based on fundamental analysis. We do so by comparing the prediction performance of three advanced machine learning methods based on fundamental analysis using financial statements.

1.4 Thesis Organization

This thesis contains six chapters that address the key aspects of the research. Chapter 1: Introduction presents the problem definition, outlines the objectives, and highlights the contributions of this research. Chapter 2: Overview covers the basic concepts concerning research. It describes principles of stock market analysis, specifically a description of the theoretical foundations that underline the methodological strategy of this thesis. In addition, it contains basic information about the RF, LSTM, and GRU models, which we used in this research. Chapter 3: Related Work reviews the literature on stock prediction techniques, including traditional statistical models and contemporary machine learning strategies. It focuses on previous research on predicting stock performance using financial ratios using machine learning. Chapter 4: Proposed Approach describes the research methodology adopted in this research, including data description, Data Preprocessing, moving time-period strategy, and the applied machine learning models RF, LSTM, and GRU. Chapter 5: Experimental Results and Discussion reports the experimental configuration, evaluation metrics, and results of the proposed models. The results are benchmarked against the TASI and assessed based on excess return, Top-K stock selection, and portfolio score, followed by a discussion of their implications for investment strategy in practical terms. Finally, Chapter 6: Conclusion and Future Work summarizes the major findings and contributions of this research, highlighting the success of integrating financial ratios with machine learning for medium- and long-term stock forecasting. Moreover, it specifies the limitations of the research and suggests directions for further research to build on and improve the current research.

Chapter 2 Overview

2.1 Introduction

The main reason for predicting stock price movements is the chance for profit. Since financial markets began, investors have looked for ways to forecast stock performance to find profitable opportunities. This involves buying stocks expected to increase in value and selling them for a profit. Over the years, many stock prediction methods have appeared, but their consistency and reliability are still debated.

Stock prediction methods can generally be grouped into three main categories: Fundamental Analysis: This approach predicts stock prices by looking at the financial health of companies using publicly available data, including financial statements and economic indicators. It aims to estimate a stock's true value.

Technical Analysis: This technique relies only on past market data such as prices and trading volumes. In contrast to fundamental analysis, technical analysis assumes that all public information is already included in a stock's price. It is based on three main ideas:

- Market action discounts everything.
- Prices are moving in trends.
- History repeats itself.

Sentiment Analysis: This method assesses investor sentiment by examining news articles, analyst reports, and online commentary related to specific stocks or the overall market. It seeks to identify the current mood or emotional tone influencing investor behavior.

Among these methods, technical analysis and sentiment analysis are mainly used for short-term predictions, for days or even hours, while fundamental analysis is usually applied for mid- to long-term forecasts, spanning quarters or years. Recently, ML and data mining techniques into financial modeling have gained significant attention. Many existing ML-based stock prediction studies focus on short-term forecasting using technical indicators or sentiment data. However, these models often neglect real-world factors like transaction costs, which can affect the practical use and reliability of their results.

This research aims to fill this gap by examining machine learning methods for long-term stock prediction based on fundamental analysis. The research

specifically investigates the predictive power of financial statement data in estimating future stock performance. To achieve this, we compare the prediction performance of three machine learning models trained on fundamental financial features. Our dataset includes quarterly financial data from 51 companies listed on the TASI (Tadawul All Share Index) between 2015 and 2025. We assess model performance by ranking 5 and 10 stocks based on their predicted relative returns and building portfolios accordingly. The actual returns of these portfolios serve as the main measure for evaluating prediction accuracy and model effectiveness.

2.2 Essential Concepts

In [2] [6], Random Forest (RF) was shown to have better portfolio performance. However, LSTM and GRU models excel in modeling temporal and sequential dependencies. Due to these strengths, we chose these models in this work.

2.2.1 Random Forest (RF)

Random Forest (RF) is a powerful ensemble learning technique that has become popular for stock selection and financial forecasting. Initially introduced by Breiman in 2001 [12] and further developed in conjunction with Cutler [10], RF is based on the concept of decision trees. While training, the model develops several decision trees, each of which is grown using a randomly drawn subset of data and input variables. Such randomness helps to promote tree diversity and avoids the possibility of overfitting, thus leading to improved generalization capability of the model.

By using bootstrap aggregation (bagging), RF also stabilizes predictions and reduces variance, rendering it a reliable and efficient means of stock performance prediction. In stock selection, RF uses historical financial and market data to recognize patterns that could predict stocks' future trend or value. These predictive patterns, as shown in Figure 2.1, offer useful recommendations to investors and portfolio managers in making sound decisions.

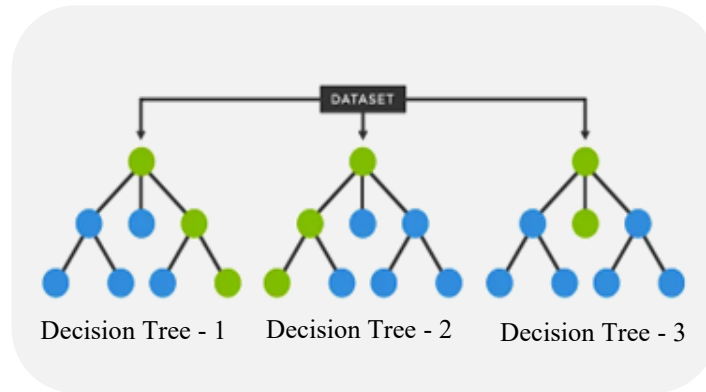


Figure 2.1 Random Forest [3]

2.2.2 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) networks were introduced by Hochreiter and Schmidhuber in 1997 [13] as a solution to the vanishing gradient problem in RNNs. The vanishing gradient effect happens when the gradient of the loss function vanishes quickly when performing backpropagation through time, making it difficult for standard RNNs to learn and capture long-term dependencies.

LSTM expands the RNN architecture by adding a memory cell that can retain information for long periods [4], thus keeping valuable contextual information intact even when there are large intervals between related inputs. It makes LSTM very suitable for modeling sequential data. Unlike standard RNNs, LSTM adds a gating mechanism controlling information flow into and out of the memory cell. Three gates specifically include input gate, forget gate, and output gate control how information is added, forgotten, and propagated. Each of the gates is controlled by a sigmoid activation function, which produces numbers between 0 and 1, effectively modulating the amount of information flow. As a result of these characteristics, LSTMs have been universally used in many fields such as natural language processing, speech recognition, and financial time series prediction, where long-term temporal dependencies need to be captured. It is defined by the following equations:

The input gate activation

$$i_t = \sigma(w_i \cdot [h_{t-1}x_t] + b_i) \quad (2.1)$$

Where:

x_t is the input at time t

h_{t-1} is the hidden state at time t-1

w_i is the weight matrices

b_i is the bias vector

The forget gate controls the flow of information from the memory cell into the hidden state. It determines what information to keep or discard from the memory cell. It is defined by the following equations:

$$f_t = \sigma(w_f \cdot [h_{t-1}x_t] + b_f) \quad (2.2)$$

Where:

f_t is the forget gate activation

w_f is the weight matrices

b_f is the bias vector.

The output gate controls the flow of information from the memory cell to the hidden state and output. It is defined by the following equations:

$$o_t = \sigma(w_o \cdot [h_{t-1}x_t] + b_o) \quad (2.3)$$

Where:

O_t is the output gate activation

w_o the weight matrices

b_o is the bias vector

The memory cell has the following equations:

$$\tilde{c}_t = \tanh(w_c \cdot [h_{t-1}, x_t] + b_c) \quad (2.4)$$

Where:

$\tilde{C}_t$ is the memory cell, w_c is the weight matrices, b_c is the bias vector.

The output of the LSTM at time t is given by:

$$h_t = o_t * \tanh(c_t) \quad (2.5)$$

Where:

h_t is the hidden state at time t.

Figure 2.2 shows the internal architecture of LSTM unit. it shows how input data x_t and previous hidden state h_{t-1} are processed through several gates: the forget gate f_t , input gate i_t , candidate cell state $\tilde{C}_t$, and output gate o_t . These gates control the flow of information and update the cell state, allowing the model to retain long-term dependencies. The updated hidden state h_t is then passed to the next step. Non-linear activation functions such as sigmoid σ and tanh are used to regulate the data through each gate. This gating mechanism helps LSTM networks overcome the vanishing gradient problem found in traditional RNNs, making them effective for sequential data modeling like time series prediction.

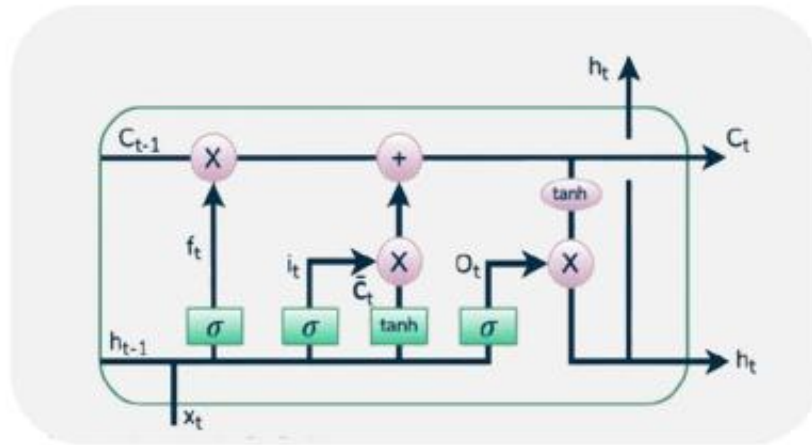


Figure 2.2 The architecture of LSTM [14]

2.2.3 Gate Recurrent Unit (GRU)

The Gated Recurrent Unit (GRU) is a type of recurrent neural network (RNN) architecture proposed by Kyunghyun Cho et al. [9] in 2014. It is very similar to the Long Short-Term Memory (LSTM) network [15], frequently achieving the same performance. Nevertheless, the GRU is more computationally efficient, needing fewer parameters, which decreases training complexity as well as memory requirements.

GRU is particularly suited for handling sequential data by facilitating the easy flow of information through time steps. This makes it very effective in scrutinizing historical stock data for dominant patterns and trends that can potentially affect future performance. Therefore, GRU serves a central role in stock forecasting models, facilitating more accurate predictions and enabling better-informed investment choices. The Gated Recurrent Unit (GRU) is also ubiquitous in sequence modeling and financial time series forecasting, where long-term temporal dependencies must be modeled. It is characterized by the following equations:

$$r_t = \sigma(w_r \cdot [h_{t-1}x_t] + b_r) \quad (2.6)$$

Where:

r_t is the reset gate activation

w_r are the weight matrices

b_r is the bias vector

The **update gate** determines how much of the past information should be carried forward to the current state:

$$z_t = \sigma(w_z \cdot [h_{t-1}x_t] + b_z) \quad (2.7)$$

Where:

z_t is the update gate activation

w_z are the weight matrices

b_r is the bias vector.

The candidate hidden state computes the new content to be added, regulated by the reset gate:

$$\tilde{h}_t = \tanh(w_h \cdot [r_t * h_{t-1} \parallel x_t] + b_h) \quad (2.8)$$

Where:

$\tilde{h}_t$ is the candidate hidden state, w_h are the weight matrices, and b_h is the bias vector.

The hidden state of the GRU at time t is given by blending the previous hidden state with the candidate hidden state, controlled by the update gate:

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \quad (2.9)$$

Where : h_t is the updated hidden state at time t .

Figure 2.3 shows the internal architecture of the GRU. It consists of two main gates: the reset gate r_t and the update gate z_t . These gates regulate the flow of information across time steps. The reset gate determines how much of the previous hidden state h_{t-1} should be considered when computing the candidate activation $\tilde{h}_t$. The **update gate** controls how much of the previous hidden state is retained versus how much of the new candidate activation $\tilde{h}_t$ is used to form the current hidden state h_t . This balance allows the GRU to preserve long-term dependencies when necessary while remaining computationally efficient.

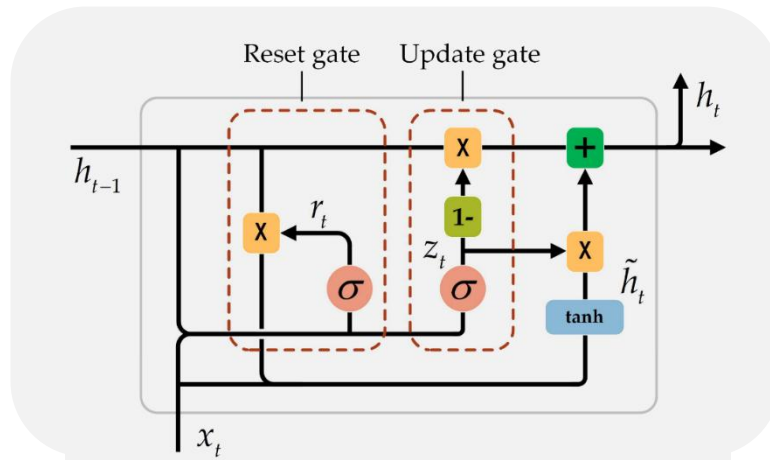


Figure 2.3 The architecture of GRU [15]

Chapter 3 Related Work

3.1 Introduction

Stock price forecasting has always been a finance and data science problem since markets are dynamic, volatile, and multi-factorial. Previously, stock prediction was largely reliant on statistical and econometric models that involved histories of past prices. However, with swift advancements in ML and DL in recent years, efforts have been made to utilize more sophisticated methods involving both technical indicators and fundamentals.

3.2 Related Work

In recent years, numerous studies have explored the application of machine learning techniques in stock prediction. Gupta, H., & Jaiswal, A. [1] in 2024 reviewed multiple approaches, including traditional techniques like Moving Averages and ARIMA, alongside deep learning models such as RNN, LSTM, CNN, and Full CNN. Using S&P 500 index data for evaluation, the research compared these models based on error metrics such as MAE, RMSE, and MSE. The findings emphasize that models with lower MAE values offer more accurate predictions, enabling investors to better anticipate market trends and maximize returns. Additionally, Tsai, P.-F., Gao, C.-H., & Yuan, S.-M. [2] in 2023, the study highlights the practical value of machine learning models RF, FNN, GRU, and FinGAT in enhancing portfolio management. These models outperform traditional benchmarks like TW50, improving stock selection and risk-adjusted returns. Despite challenges related to fiscal year misalignment, they demonstrate robust predictive performance. the models deliver higher precision than random selection, offering investors reliable strategies across Top-10 and Top-20 portfolios. Overall, the findings support the use of these models for informed investment decisions in dynamic market environments. On the other hand, Lu, C., & Xu, Y, in [3] 2023 This study introduces a TRNN for efficient stock price prediction using price-volume data. By incorporating trading volume and applying sliding windows for preprocessing, the model captures key market trends and turning points. Unlike traditional RNN and LSTM models, TRNN enhances both prediction accuracy and training efficiency through two-dimensional representation of price-volume relationships. Experimental results

demonstrate that TRNN outperforms standard RNN and LSTM models in both accuracy and computational efficiency. Kunathananon, T. [4] in 2022 the study highlights the effectiveness of LSTM in stock price prediction by capturing long-term dependencies, handling noise, and minimizing human bias. LSTM outperformed models like linear regression and DNN, particularly for SET50 stocks. While performance varied across industries, the results demonstrate LSTM's value as a low-cost, accurate forecasting tool. The study suggests future work should explore additional factors and updated data to enhance prediction accuracy, reinforcing LSTM's role in improving financial decision-making. Lu, Z.-Y. [5] In 2022, the study proposed a CMPS for stock portfolio management by integrating stock quotes and financial indices. Using deep Q-networks and self-attention mechanisms, the model captured time-series patterns and inter-stock correlations. An enhanced version, CMPS-RF, incorporated a risk-free asset strategy to manage market risk. Experimental results on the Shanghai Stock Exchange 50 showed that both CMPS and CMPS-RF outperformed benchmark models in profitability and achieved superior risk-adjusted returns, highlighting their potential for robust investment strategies. Huang, Y., Capretz, L. F., & Ho, D. [6] in 2021 highlighted the effectiveness of LSTM in predicting stock prices using financial statements and market data. LSTM demonstrated superior performance compared to linear regression and DNN models, particularly for stocks in the SET50 index, due to its ability to capture long-term dependencies and reduce noise and bias. While model performance varied across industries, the results support the use of LSTM as a low-cost, accurate forecasting tool. The study also suggests that incorporating more recent and diverse data could further enhance prediction accuracy. Heo, J., & Yang, J. Y. [7] in 2016 explored stock price fluctuation prediction using SVM based on fundamental analysis. Financial indicators such as assets and profits were used as inputs to assess a company's condition and predict future price movements. Results showed that SVM-based forecasts outperformed expert predictions, particularly in the short term. However, predictability declined over time, aligning with the Efficient Market Hypothesis, which suggests that market prices eventually reflect all available information. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. [8] in 2014, This paper introduces the RNN Encoder-Decoder architecture, comprising two recurrent neural networks: one for encoding input sequences into fixed-length vector representations, and

another for decoding these into output sequences. Joint training is applied to optimize the conditional probability of a target sequence given a source sequence. The model enhances machine translation performance by supplying learned phrase-level conditional probabilities to a traditional log-linear framework. Additionally, it demonstrates the ability to capture semantically and syntactically rich phrase representations. Cutler, A., Cutler, D. R., & Stevens, J. R. [9] in 2012. RFs ensemble learning methods composed of multiple decision trees, where each tree is constructed using a randomly sampled subset of features. As the number of trees increases, the generalization error converges almost surely. The overall performance depends on both the strength of individual trees and the correlation among them. Randomized feature selection at split nodes improves robustness and yields competitive error rates compared to methods like AdaBoost, particularly in noisy environments. Additionally, internal out-of-bag estimates facilitate the evaluation of model error, variable importance, and feature interactions, making the approach applicable to both classification and regression tasks. Han, S., & Chen, R.-C. [10] in 2007 the study introduced a stock prediction method that integrates SVM with financial statement analysis. Unlike traditional technical analysis, this approach leverages financial indices derived directly from periodically released company reports, resulting in more reliable and stable predictions. The experimental results showed improved accuracy compared to using SVM alone. Furthermore, the method enables the discovery of association rules among financial indicators and supports customized evaluation aligned with shareholder expectations. Despite its strengths, limitations exist—such as potential bias in expert-evaluated benchmarks, limited sample size, and data manipulation risks. Future research aims to expand the dataset and refine prediction accuracy based on real-world comparisons.

Chapter 4 Proposed Approach

4.1 METHODOLOGY

Three machine learning models: Random Forest (RF), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU), are used in this research to forecast stock performance using ratios from financial statements. The models leverage past quarterly data to predict future stock returns. The models are evaluated using the Mean Absolute Error (MAE) and other metrics such as Excess Return, Top-k Precision, and Portfolio Score.

4.1.1 Data Description

Within the Saudi stock market (TASI), our stock pool comprises 51 companies selected based on sectoral representation. These include the Energy sector, which consistently holds the highest market capitalization; the Materials sector, recognized as the largest non-oil contributor; and the Capital Goods sector. The TASI index serves as the benchmark for evaluating portfolio performance, providing a standard against which to compare the returns of model-generated portfolios.

The selected stocks span the period from Q1 2015 to Q1 2025, covering 41 quarters of financial data. Based on this comprehensive dataset, we developed four predictive models to estimate the relative return of stocks. The goal is to identify the top 5 and top 10 performing stocks each quarter to construct optimized investment portfolios. Portfolio returns are then compared to the TASI index to evaluate relative performance.

Additionally, OHLCV (Open, High, Low, Close, Volume) data was collected for the same stock pool to support realistic investment simulations. Opening prices were used to represent the feasible purchase price of each stock at the beginning of the investment period. This approach ensures that return calculations reflect actual market conditions, offering a more accurate assessment of portfolio performance.

The dataset was compiled through web scraping from the Saudi Exchange (Tadawul) [15] and other reputable financial data providers [16][17].

4.1.2 Financial Ratios

We use 13 financial ratios as input features for the predictive models, consistent with those employed in research [5]. These ratios, along with their respective calculation formulas, are presented in Table 1.

Ratio	Calculation	Description
Current Ratio	$\text{Current Assets} \div \text{Current Liabilities}$	Measures a company's ability to pay short-term obligations with short-term assets.
Debt to Equity	$\text{Total Debt} \div \text{Shareholders' Equity}$	Shows the proportion of debt used relative to equity financing.
Long Term Debt/Capital	$\text{Long-Term Debt} \div (\text{Long-Term Debt} + \text{Shareholders' Equity})$	Indicates the proportion of long-term debt in the total capital structure.
Asset Turnover	$\text{Revenue} \div \text{Average Total Assets}$	Measures how efficiently assets are used to generate revenue.
Inventory Turnover	$\text{Sales} \div \text{Inventories}$	Shows how often inventory is sold and replaced over a period.
Receivable Turnover	$\text{Net Credit Sales} \div \text{Receivables}$	Indicates how effectively receivables are collected.
Days Sales in Receivables	$(\text{Receivables} \div \text{Net Credit Sales}) \times 365$	Measures the average collection period for receivables.
Book Value Per Share	$(\text{Shareholders' Equity} - \text{Preferred Equity}) \div \text{Total Outstanding Shares}$	Represents the equity value allocated to each outstanding share.
Gross Profit Margin	$(\text{Revenue} - \text{COGS}) \div \text{Revenue}$	Shows the percentage of revenue remaining after direct costs.
EBITA Margin	$\text{EBITA} \div \text{Revenue}$	Measures profitability before interest, taxes, and amortization.
EBIT Margin	$\text{EBIT} \div \text{Revenue}$	Measures profitability before interest and taxes.
Net Income Margin	$\text{Net Income} \div \text{Revenue}$	Shows the percentage of revenue remaining as net profit.
Return on Assets	$\text{Net Income} \div \text{Total Assets}$	Measures efficiency in using assets to generate profit.
Return on Capital	$\text{EBIT} \div (\text{Total Debt} + \text{Shareholders' Equity})$	Evaluates returns generated from total capital employed.
Return on Equity	$\text{Net Income} \div \text{Shareholders' Equity}$	Measures return generated on total shareholders' equity.
Return on Common Equity	$\text{Net Income Available to Common Shareholders} \div \text{Common Equity}$	Focuses on returns to common shareholders, excluding preferred equity.

Table 4.1 13 Financial Ratios

4.1.3 Data Preprocessing

The original data set had a wide range of both financial and non-financial variables, but it was also a dataset with a lot of missing values. A large part of the missing values formed large clusters for some features and other missing values were randomly placed in the dataset. A combination of feature removal and data imputation techniques were found to be adequate to resolve these missing values.

We removed features that had over 50 percent missing data, the remaining features still contained some sparse missing values. These were resolved using linear interpolation with adjacent quarter values. If, for example, the ROE for Q3 2015 was absent, we would estimate it by taking the 2Q and 4Q values for 2015, averaging them, and assigning that as the ROE for Q3 2015.

Subsequently, the dataset was standardized to normalize the features. Each input variable was transformed to have a mean of 0 and a standard deviation of 1. This standardization ensures that all features contribute equally during model training. In addition to improving consistency, normalization enhances the model's convergence by reducing the risk of being trapped in local minima. Z-score normalization was used, defined by the following equation:

$$z = \frac{x - \mu}{\sigma} \quad (4.1)$$

Where:

μ is the mean of x

σ is the standard deviation of x

After completing the data preparation, the final dataset included 51 companies, each with 41 quarterly records over a 10-year period, resulting in a total of 2,091 records. For further analysis, 13 financial ratios were extracted, as detailed in Section 4.1.2.

4.1.4 Calculation of the Target Variable – Relative Return

The absolute returns quarterly of stock is defined as follows in Equation (4.2):

$$R_{absolute}(q) = \frac{cp(q)}{op(q)} - 1 \quad (4.2)$$

Where:

$R_{absolute}(q)$ is absolute returns quarterly of stock

$cp(q)$ is the stock's price on the last day of quarter q

$op(q)$ is the stock's price on the first day of quarter q .

To control movements in the market, and better isolate company specific performance, we measure the relative return by adjusting the absolute return of the stock with respect to a market benchmark. The relative quarterly return is defined in Equation (4.3):

$$R_{relative}(q) = R_{absolute}(q) - R_{TASI}(q) \quad (4.3)$$

$R_{absolute}$ is the quarterly absolute return of stock for quarter q , and the R_{TASI} is the quarterly return of the TASI as a benchmark portfolio. The absolute return of a stock is susceptible to influence not just from the company's performance but also from the broader stock market conditions. This inherent volatility makes it a less stable target for model learning. Consequently, we perform preprocessing by converting absolute returns into relative returns, which serves as the training target for our modeling efforts.

4.1.5 Data Split and Moving Time Period

The RF models encompass a total of 32 quarters for each moving period. The initial data from the moving period span from 2015Q1 to 2023Q1, and so forth. We partitioned these data into three segments: the training set, the validation set, and the test set, comprising 24 quarters, 8 quarters, and 1 quarter, respectively as in Figure 4.1. The training set facilitates model weight optimization, and the validation set aids in finetuning hyperparameters, ensuring model convergence without overfitting. Subsequently, we employ the test set to predict the relative return of the 51 stocks.

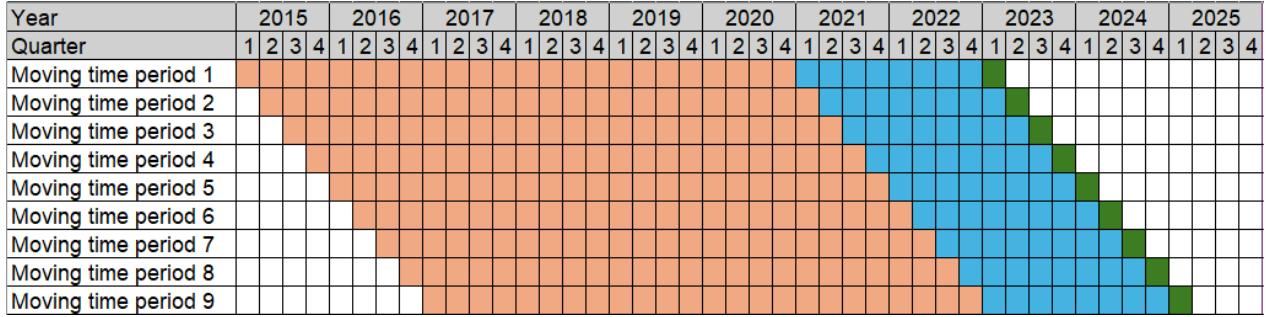


Figure 4.1. Rolling Window Data Splitting for Random Forest Training

The LSTM and GRU models span a total of 35 quarters for each moving period, with the initial data originating from 2015Q1 to 2023Q3 and so on. We segment this data into the training set, validation set, and test set, consisting of 20 quarters, 6 quarters, and 4 quarters, respectively as in Figure 4.2. Like the previous models, the training set facilitates model weight optimization, and the validation set aids in hyperparameter fine-tuning, guaranteeing model convergence without overfitting.

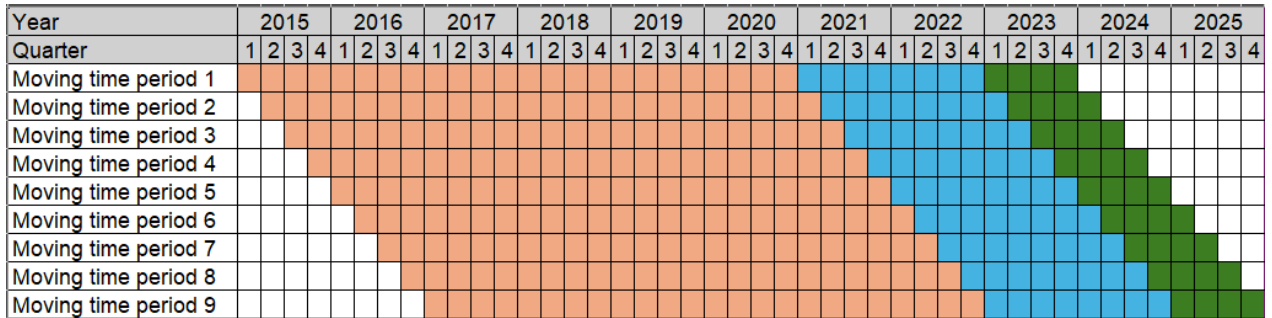


Figure 4.2. Rolling Window Data Splitting for LSTM and GRU Training

4.1.6 Evaluation Metrics

To evaluate the effectiveness of the predictive models and the constructed portfolios, three key performance metrics were utilized:

4.1.6.1 Excess Return

This metric measures the difference between the average return of the constructed portfolio and that of the benchmark index (TASI), providing insight into the value added by the model's stock selection strategy.

Excess returns are a useful measure to evaluate the performance of an investment portfolio. By comparing portfolio returns to those of a benchmark, investors can determine whether the portfolio is generating value above the benchmark.

The excess return on a benchmark $ER_{TASI}(q)$ of portfolio p for quarter q is formulated as follows in Equation (4.4):

$$ER_p = \{\prod_{q=1}^n \mathbf{1} + R_p(q)\} - \{\prod_{q=1}^n \mathbf{1} + R_{TASI}(q)\} \quad (4.4)$$

Where:

$R_p(q)$ is the actual absolute return

$R_{TASI}(q)$ is the actual absolute return of TASI index

4.1.6.2 Top-k Precision

This assesses the accuracy of the model in identifying the true top-performing stocks within each quarter, offering a direct evaluation of predictive ranking quality. The precision of k is used to measure the percentage of overlap between a list of top k stocks with the highest predicted returns, which is ‘predicted top k stocks’, and a list of top k stocks with the highest actual return, which is ‘actual top k stocks’. The higher the precision of the top- k , the more accurate the prediction of the model will be. Top- k precision can be calculated as follows via Equation (4.5):

$$\text{Top} - k \text{ Precision} = \frac{|\text{Predicted } k \cap \text{Actual } k|}{k} \quad (4.5)$$

Where:

k The number of top stocks selected

$\cap$ Intersection, the number of stocks that appear in both the predicted and actual top-k lists

$$K \text{ Precision}_{\text{random}} = \frac{K}{\text{Total number of stocks}} \quad (4.6)$$

For Top-5 out of 51 stocks: $5/51 = 9.8\%$

For Top-10 out of 51stocks: $10/51 = 19.6\%$

4.1.6.3 Portfolio Score

A portfolio score is a composite metric used to provide a quick assessment of an investment portfolio's quality. It typically incorporates factors such as risk level, risk-adjusted performance, diversification, fees, and resilience under market stress. While the specific methodology and components may vary by provider, the overarching goal is to offer a comprehensive overview of how effectively a portfolio aligns with an investor's objectives and risk preferences, thereby guiding potential areas for improvement.

As a risk-adjusted metric, it integrates both return and standard deviation to evaluate not just the magnitude but also the consistency of portfolio performance over time. Together with other evaluation criteria, portfolio score serves as a robust and practical framework for assessing both predictive accuracy and the real-world utility of investment strategies.

In our research, we use the same modified portfolio score [2] to measure the equally weighted portfolio as used in previous related works. In Equation (4.7), a higher portfolio score indicates the selection of a portfolio with high return and low risk.

$$\text{Portfolio Score} = \frac{\overline{R_p}}{\sigma_p} \quad (4.7)$$

Equation (4.8) to calculate average actual relative return $\bar{R}_p$ after subtracting the quarterly TASI index return, as shown below:

$$\bar{R}_p = \frac{1}{N} \sum_{q=1}^{q=N} R_p(q) \quad (4.8)$$

Where:

$R_p(q)$ is the actual absolute return for quarter q .

$$\sigma_p = \sqrt{\frac{1}{N} \sum_{q=1}^{q=N} (R_p(q) - \bar{R}_p)^2} \quad (4.9)$$

Where:

- N is total number of quarters
- $R_p(q)$ is actual relative return in quarter q
- $\bar{R}_p$ is average actual relative return across all quarters

4.1.7 Training

The training set, validation set, and test set are constructed by progressing through sequential time periods using a rolling window approach.

Once a model is trained and its hyperparameters fine-tuned using the validation set, it is evaluated on the corresponding test set within that rolling period. In each test window, the model generates predicted relative returns for all 51 stocks in the investment universe. These predicted returns are then sorted in descending order to facilitate stock ranking.

Following this ranking process, two portfolios are formed: the Top-5 portfolio, composed of the five stocks with the highest predicted returns, and the Top-10 portfolio, containing the top ten predicted performers. This stock selection and portfolio formation process is repeated across all rolling periods, resulting in a dynamic, time-aware investment simulation, as shown in Figure 4.3.

Three distinct models were developed in this study: Random Forest (RF), implemented using the scikit-learn library in Python; and two recurrent neural networks, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), both implemented using the PyTorch deep learning framework.

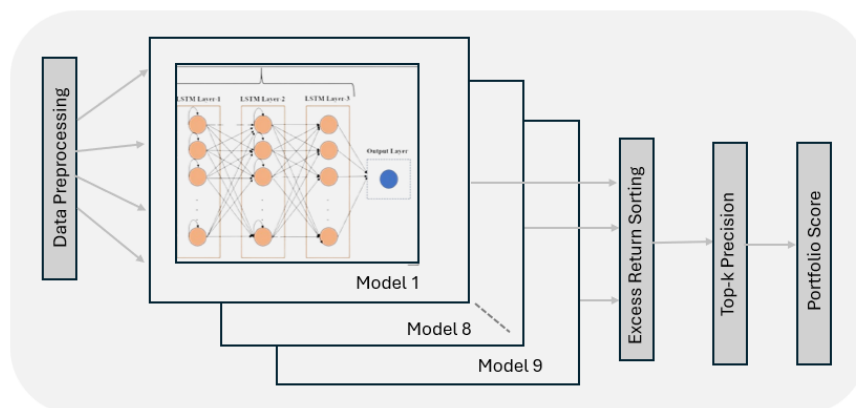


Figure 4.3 Model Training Architecture

These models were trained independently on the same dataset, with their respective architectures and hyperparameters optimized using the validation set. The use of multiple models allows for comparative analysis of performance across different machine learning paradigms, ensemble-based versus deep sequential learning.

After predictions are generated for each rolling window, the top-k strategy is implemented, where the top 5 and top 10 ranked stocks are selected to construct investment portfolios. These portfolios are hypothetically for the duration of the test quarter, after which they are rebalanced based on the next period's predictions. This process continues for all rolling windows, simulating a real-time, mid-term and long-term investment strategy.

4.1.7.1 Random Forest Hyperparameters

We use scikit-learn to develop the RF model. Hyperparameters are tuned with the validation set for preventing overfitting where the number of estimators is 1000 and the optimization criterion used SE.

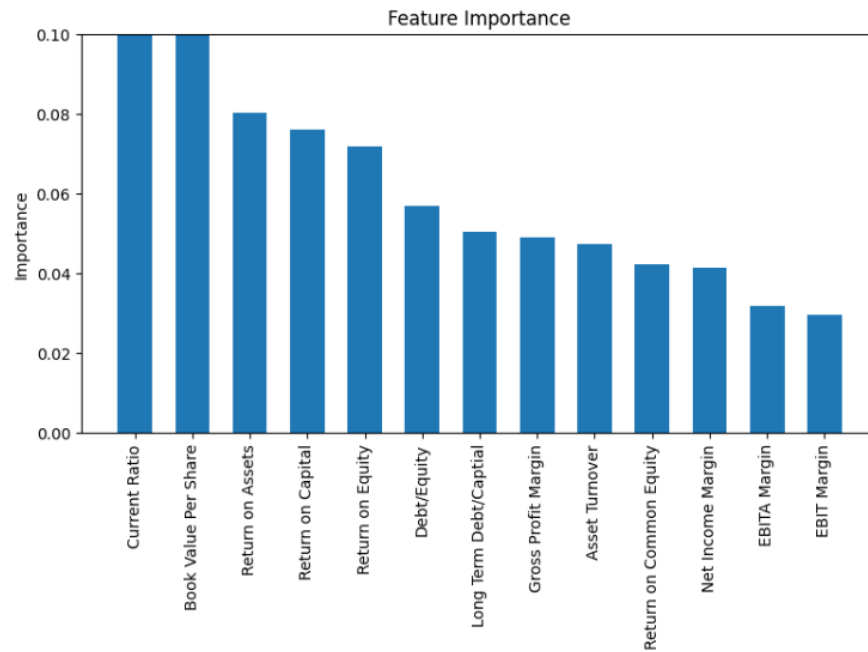


Figure 4.4. Importance of features of RF model

Figure 4.4 presents the feature importance derived from RF model. The importance score reflects the contribution of each financial ratio in predicting stock returns. Among the features, the Current Ratio and Book Value Per Share emerged as the most influential, indicating their strong predictive power in the model. Other notable features include Return on Assets, Return on Capital, and Return on Equity, all of which are key profitability indicators. Features such as Debt to Equity, Long-Term Debt to Capital, and Asset Turnover also contributed meaningfully. At the lower end, features like EBIT Margin and Net Income Margin showed relatively less impact. This distribution suggests that liquidity and value-related ratios are more informative for the RF model in forecasting relative returns than some profitability or efficiency measures.

4.1.7.2 LSTM Hyperparameters

We use Pytorch to develop LSTM models. LSTM are tuned using the validation set to prevent overfitting, as shown in Table 4.2.

Name of Hyperparameters	Optimized Result
Window size	8
Hidden size	64
Epochs	100
Batch size	128
Dropout rate	0.4
Input size	13
Layers	2
Loss Function	MAE
Learning rate	0.0005
Optimizer	Adam

Table 4.2 Hyperparameters of LSTM model

4.1.7.3 GRU Hyperparameters

We use Pytorch to develop GRU models. Hyperparameters of GRU are tuned using the validation set to prevent overfitting, as shown in Table 4.3.

Name of Hyperparameters	Optimized Result
Window size	4
Hidden layers	1
Hidden dimension	36
Epochs	100
Batch size	128
Loss function	MAE
Dropout rate	0.4
Input size	13
Layers	2
Learning rate	0.0001
Optimizer	Adam

Table 4.3 Hyperparameters of GRU model

Chapter 5 Experimental Results and Discussion

5.1 Results

The evaluation compares three models RF, LSTM, and GRU using three key portfolio performance metrics: Excess Return, Top-k, and Portfolio Score, between the TASI benchmark, and Top-5 and Top-10 stock selections. The comparison is made against the TASI benchmark and includes Top-5 and Top-10 stock selection strategies. Table 5.1 presents the evaluation results. Among the tested models, the GRU model achieved the highest Portfolio Score of 0.61 for Top-10 selection, reflecting strong performance in terms of return relative to risk. The TASI benchmark, serving as a baseline, had a natural Top-5 and Top-10 inclusion rate of 9.8% and 19.6%, respectively, given the 51-stock universe.

	RF		LSTM		GRU		TASI
Top - K	Top 5	Top 10	Top 5	Top 10	Top 5	Top 10	
Portfolio Score	0.51	0.48	0.55	0.56	0.58	0.61	0.29
Excess return	79.3%	48.8%	65.1%	61.4%	58.2%	66.8%	Baseline 0%
Top-k Precision	16.2%	23.6%	19.1%	21.4%	22.8%	29.5%	-
MAE	1.63	1.14	0.22	0.13	0.14	0.11	-
STD	20.1%	12.5%	11.2%	13.6%	14.3%	10.2%	11.3

Table 5.1 Models Evaluation

For TASI Top-5 out of 51 stocks: $5/51 = 9.8\%$

For TASI Top-10 out of 51stocks: $10/51 = 19.6\%$

5.1.1 High Portfolio Scores

In Table (5.1) among the models, the GRU model achieved the highest performance, which made a Portfolio Score of 0.61, as Section 4.1.4.3 and Equation (4.8) shows that as a metric designed to strike an equilibrium between stock return profitability and investment risk. It is calculated by averaging the returns over quarters and dividing by the investment risk. Higher portfolio scores denote stable and advantageous portfolio from a financial perspective.

5.1.2 High Excess Return

As shown in Table 5.1, the RF model achieves the highest excess return of 79.3% in the Top-5 stock selection. However, this high return is accompanied by greater volatility, as indicated by its relatively high STD = 20.1%, which may imply increase investment risk.

In contrast, the GRU model delivers an excess return of 66.8% in the Top-10 portfolio, which is also notably strong and competitive. More importantly, GRU achieves this return with the lowest STD is 10.2% among all models, indicating superior risk-adjusted performance and return stability. This makes GRU more suitable for investors because of ER with lower risk.

The LSTM model also demonstrates strong performance, particularly in the Top-5 selection, with an excess return of 65.1%. While it falls slightly short of RF in raw return, its lower STD is 11.2% reflects better risk control and prediction consistency.

5.1.3 Top-k Precision

The Top-k Precision metric evaluates the alignment between the predicted top-k stocks, those expected to produce the highest returns, and the actual top performing stocks in each quarter Section 4.1.4.1. Higher precision reflects the model's capability to accurately identify high-return stocks.

As shown in Table 5.1, the GRU model achieves the highest Top-10 Precision at 29.5%, followed by RF at 23.6%, and LSTM at 21.4%. These values notably outperform the baseline of TASI benchmark of 19.6% , confirming that all three models add predictive value over random chance when constructing Top-10 portfolios.

In the more selective Top-5 setting, GRU records the highest precision at 22.8%, ahead of LSTM at 19.1% and RF at 16.2%. These values exceed the baseline of TASI benchmark of 9.8%, demonstrating that even with smaller portfolio sizes, the models maintain predictive strength. It is worth noting that Top-10 portfolios generally achieve higher precision than Top-5 portfolios. This reflects a common trade-off: broader portfolios reduce volatility and improve alignment with actual outcomes, while narrower selections may target higher returns but with reduced predictive consistency.

5.1.4 Predict Stock Price

In addition to financial ratios, we incorporated technical features derived from OCHLV data to enhance the model's ability to capture short-term stock movement patterns.

Figure 5.1: Predicted vs Actual Close Price using OCHLV data. The plot shows a strong linear relationship, indicating accurate forecasting by the LSTM model.

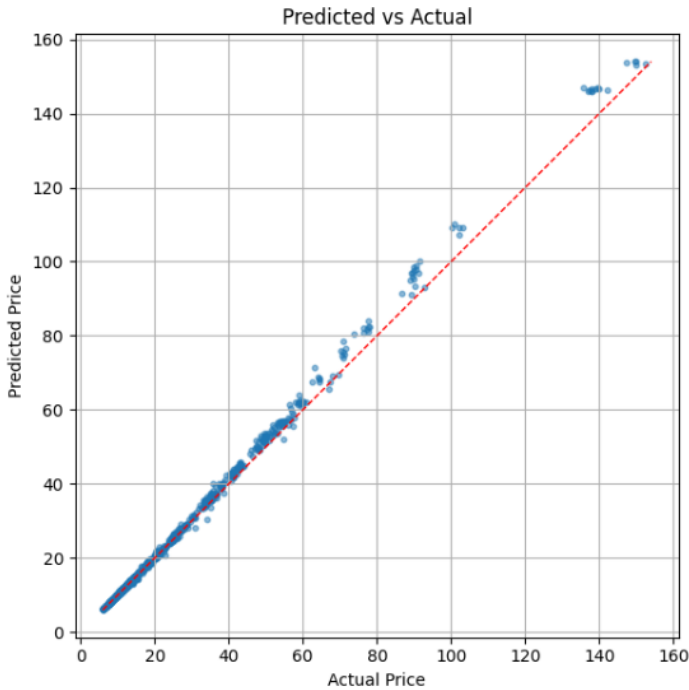


Figure 5.1 Predicted vs Actual Price (OCHLV Input Features)

Matrices	Values
MAE	0.02638
R^2	0.97

Table 5.2 Evaluation of LSTM Performance on OCHLV Data

Table 5.2 shows evaluation metrics of the LSTM model's performance. MAE is 0.0238, indicating that the model's predictions deviate from actual values by a small margin on average. R^2 is 0.97, means 97% of the variance in actual prices. This high R^2 value further confirms the model's robustness and reliability in predicting short-term stock price movements using technical indicators.

Chapter 6 Conclusion and Future Work

6.1 Conclusion

The successful application of three machine learning models RF, LSTM, and GRU for predicting quarterly stock returns and constructing predictive investment portfolios, carries significant managerial implications for enhancing portfolio management strategies.

- **Improved Stock Selection:** The superior performance of these models, particularly when benchmarked against the TASI index, positions them as valuable tools for stock selection. This enables portfolio managers to make more informed and data-driven decisions, offering more effective alternatives for discerning investors.
- **Optimal Risk–Return Balance:** As evidenced by the STD values in Table 5.1, the models demonstrate a strong balance between return and risk. This highlights the effectiveness of the risk-return management strategy, allowing managers to achieve higher returns without significantly increasing portfolio risk, an essential aspect of optimizing risk-adjusted performance.
- **Methodological Robustness:** The consistent outperformance of our models, as shown through portfolio scores exceeding the TASI benchmark, reflects their methodological strength. This reliability reinforces their practical value in maintaining disciplined and effective portfolio strategies.
- **Precision Enhancement:** While the precision of Top-5 and Top-10 selections may not always be statistically significant, the models clearly outperform random stock selection. This enables portfolio managers to achieve more predictable results, with precision ranges of 6.4% to 13% for Top-5 portfolios and 1.8% to 9.9% for Top-10 portfolios, enhancing accountability and outcome reliability.
- **Expanded Investment Options:** The availability of alternative portfolio configurations beyond the TASI index or random selection empowers investors with more tailored investment strategies. Managers can recommend Top-5 portfolios for those seeking higher excess returns with moderate risk, or Top-10 portfolios for more conservative investors aiming to reduce volatility while maintaining strong excess returns.
- **Integration of Technical Indicators for Short-Term Forecasting:** The inclusion of technical indicators significantly enhances the model's ability to capture short-term market dynamics. The model demonstrated strong predictive performance,

achieving a low MAE of 0.0238 SAR and explaining 97% of the variance in actual stock prices. These results validate the effectiveness of integrating deep learning techniques with technical features for accurate and timely short-term stock price forecasting.

In conclusion, the managerial implications highlighted in this research point out the strategic advantages and adaptability of the proposed models. They offer valuable insights for optimizing portfolio performance in dynamic market conditions by enabling enhanced risk-adjusted returns and reliable predictive capabilities. Furthermore, the integration of technical indicators into deep learning models significantly improved short-term forecasting accuracy, effectively capturing rapid market fluctuations. These findings affirm the versatility of machine learning-based approaches in supporting both short- and long-term investment decision-making.

6.2 Future Work

Looking ahead, future work could focus on enhancing model performance by incorporating additional predictive features such as macroeconomic indicators or sentiment data. There is also goal to improve the optimization of model hyperparameters and explore alternative Machine Learning models capable of capturing more complex temporal and cross-sectional patterns in financial data. Moreover, extending the evaluation framework to include real-world considerations like transaction costs, market liquidity, and risk-adjusted performance metrics would provide a more comprehensive assessment of practical investment outcomes.

References

1. Gupta, H., & Jaiswal, A. (2024). A study on stock forecasting using deep learning and statistical models. arXiv.
2. Tsai, P.-F., Gao, C.-H., & Yuan, S.-M. (2023). Stock selection using machine learning based on financial ratios. *Mathematics*, MDPI, 4758.
3. Lu, C., & Xu, Y. (2023). TRNN: An efficient time-series recurrent neural network for stock price prediction. *Information Sciences*, 640, 119–134.
4. Kunathananon, T. (2022). Estimating stock price based on information from financial statement using machine learning approach (Master's thesis, Chulalongkorn University). Chula Digital Collections.
5. Lu, Z.-Y. A Deep Reinforcement Learning-Enabled Portfolio Management System with Quarterly Stock Re-Selection Based on Financial Statements. Master's Thesis, National Yang Ming Chiao Tung University, Taiwan, China, 2022.
6. Huang, Y., Capretz, L. F., & Ho, D. (2021). Machine learning for stock prediction based on fundamental analysis. *Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–10. Orlando, FL, USA.
7. Heo, J., & Yang, J. Y. (2016). Stock price prediction based on financial statements using SVM. *International Journal of Hybrid Information Technology*, 9(2), 57-66.
8. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. arXiv preprint arXiv:1406.1078
9. Cutler, A., Cutler, D. R., & Stevens, J. R. (2012). Random forests Ensemble machine learning: Methods and applications (pp. 157–175). Springer.
10. Han, S., & Chen, R.-C. (2007). Using SVM with financial statement analysis for prediction of stocks. *Communications of the IIMA*, 7(4), Article 8.
11. Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32
12. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780

13. GeeksforGeeks. (2025, July 23). Random forest algorithm in machine learning. GeeksforGeeks. <https://www.geeksforgeeks.org/machine-learning/random-forest-algorithm-in-machine-learning>
14. Naveen. (2022, April 6). *What is the difference between LSTM and GRU?* NoMiDL. <https://www.nomidl.com/deep-learning/what-is-the-difference-between-lstm-and-gru>
15. Saudi Exchange. <https://www.saudiexchange.sa/>
16. Market Screener. <https://www.marketscreener.com/>
17. Investing.com. <https://www.investing.com/>