HYBRID GRU-LSTM WITH INTEGRATED FUNDAMENTAL AND TECHNICAL ANALYSIS FOR STOCK PRICE PREDICTION

ADRIAN CHIENG HONG JIE

XIAMEN UNIVERSITY MALAYSIA

2024

XIAMEN UNIVERSITY MALAYSIA
厦門大學馬來西亞分校

FINAL YEAR PROJECT REPORT

HYBRID GRU-LSTM WITH INTEGRATED FUNDAMENTAL AND TECHNICAL ANALYSIS FOR STOCK PRICE PREDICTION

NAME OF STUDENT	:	ADRIAN CHIENG HONG JIE
STUDENT ID	:	AIT2109934
SCHOOL/ FACULTY	:	SCHOOL OF ELECTRICAL ENGINEERING AND ARTIFICIAL INTELLIGENCE
PROGRAMME	:	BACHELOR OF SCIENCE IN ARTIFICIAL INTELLIGENCE
INTAKE	:	2024/09
SUPERVISOR	:	DR HAKIM ABDULRAB LECTURER

SEPTEMBER 2024

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at Xiamen University Malaysia or other institutions.

Signature : ______________________________

Name : _____Adrian Chieng Hong Jie_____

ID No. : ________AIT2109934_____________

Date : ________22 December 2024________

APPROVAL FOR SUBMISSION

I certify that this project report entitled "**HYBRID GRU-LSTM WITH INTEGRATED FUNDAMENTAL AND TECHNICAL ANALYSIS FOR STOCK PRICE PREDICTION**" that was prepared by ADRIAN CHIENG HONG JIE has met the required standard for submission in partial fulfillment of the requirements for the award of Bachelor of Engineering in Artificial Intelligence (Honours) at Xiamen University Malaysia.

Approved by,

Signature : ______________________

Supervisor : Dr. Hakim Abdulrab

Date : 23/12/2024

ACKNOWLEDGEMENTS

I would like to thank all who have contributed to the successful completion of this project. I would like to express my heartfelt gratitude to my research supervisor, Dr. Hakim Abdulrab for his invaluable advice, guidance, and his enormous patience throughout the development of the research.

In addition, I would also like to express my gratitude to my loving parents, my dearest sister, and friends who have helped and given me encouragement.

ABSTRACT

In the rapidly evolving financial markets, it is challenging to accurately predict stock prices due to the inherent complexities, non-linear dynamics, and high volatility of financial data. These complex patterns are not well captured by traditional forecasting models such as ARIMA, as they are mostly linear and cannot deal with large, unstructured data. Despite their capability of capturing temporal dependencies, standalone deep learning models like GRU and LSTM are often not sufficient to take full advantage of the benefits of each to boost efficiency. While technical indicators such as moving averages and RSI help identify trends, fundamental data plays an important role to provide insights about the company's financial health. This absence of an integrated technique that incorporates the advantages of both technical and fundamental methods along with the difficulty to fine-tune deep learning architectures for financial datasets highlights a research gap that calls for hybrid models to overcome these shortcomings. Hence, the author developed a hybrid GRU-LSTM stock price prediction model with integrated fundamental and technical data, aiming to enhance predictive accuracy. Traditional models such as ARIMA, standalone GRU, and LSTM models were benchmarked to investigate their limitations in capturing the non-linearity dynamics of financial markets. The research utilized historical stock data retrieved from Yahoo Finance for two selected stocks—600719.SS and 000679.SZ. From the retrieved data, feature engineering was carried out to derive meaningful technical and fundamental indicators such as MACD, RSI, stochastic indicators, number of transactions, Price-to-Earnings (P/E) ratios, and Profitability. The proposed hybrid model outperformed the other models, achieving RMSE of 0.0141 for stock 600719 and 0.0360 for stock 000679. Furthermore, experiments aimed at identifying the optimal number of GRU and LSTM layers showed that balanced and moderate architectures provided better generalization and reduced overfitting.

KEYWORD: Stock Price Prediction, Long Short Term Memory, Gated Recurrent Unit, Technical Analysis, Fundamental Analysis

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

c_t	cell state
σ	sigmoid activation function
h_t	hidden state
r_t	reset gate
z_t	update gate
x_t	current input
tanh	tangent activation function
y_i	actual value
$\hat{y}_i$	predicted value

ARIMA	Autoregressive Integrated Moving Average
AI	Artificial Intelligence
GRU	Gated Recurrent Unit
LSTM	Long Short-Term Memory
EPS	Earnings Per Share
P/E	Price-to-Earnings
MSE	Mean Squared Error
RMSE	Root Mean Squared Error
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MACD	Moving Average Convergence Divergence
RSI	Relative Strength Index
NYSE	New York Stock Exchange
NSE	National Stock Exchange of India Limited
ML	Machine Learning
SVR	Support Vector Regressions
RF	Random Forests
XGBoost	Extreme Gradient Boosting

RNN	Recurrent Neural Network
D/E	Debt to Equity
P/S	Price to Sales
P/B	Price to Book
CNN	Convolutional Neural Network
EDA	Exploratory Data Analysis
000679	Dalian Friendship (Group) Co., Ltd.
600719	Dalian Thermal Power Co., Ltd.
MA	Moving Averages
EMA	Exponential Moving Averages
SMA	Simple Moving Averages
ReLU	Rectified Linear Unit
ADF	Augmented Dickey-Fuller
SARIMAX	Seasonal Autoregressive Integrated Moving Average with Exogenous Factors

CHAPTER 1

INTRODUCTION

1.1 Background

The stock market is a very complicated system that plays a crucial role in the world economy. The ability to forecast stock prices has always been a major issue to stakeholders such as retail investors, institutional investors, financial analysts and researchers. However, traditional models such as ARIMA, which are still widely used, quite often fail to capture the non-linearity within the financial markets, which are impacted by a multitude of factors. In recent years, the advancement of AI and the availability of large amounts of data to process in the financial sector have significantly boosted the search for more accurate prediction models.

Besides this, in most of the prior works, most of the attention has been paid to the technical analysis for stock price prediction which is based on historical price and volume data. This methodology fails to consider the economic factors that may determine a company's intrinsic value, hence underlining the importance of using fundamental analysis to adequately assess the strength of a firm's financial health.

Advancements in machine learning and the availability of financial data have paved the way for more accurate stock price predictions. For instance, hybrid deep learning models such as GRU and LSTM have emerged as the best approaches for modeling both short-term and long-term dependencies in financial time series data. These models combine technical indicators derived from past stock prices with fundamental factors that indicate the company's health in an attempt to give a better analysis of stock trends. This study aims to establish such hybrid models to overcome the drawbacks of conventional models and advance the knowledge base in stock price prediction.

This study is made up of five chapters, namely; Introduction, Chapter Literature Review, Research Methodology, Results and Discussion, and Conclusion. Chapter One is divided into five subtopics: 1.1 Background, 1.2 Problem Statement, 1.3 Aims and Objectives, 1.4 Research Scope, and 1.5 Thesis Organization.

1.2 Problem Statement

Stock price prediction is always challenging due to its dynamic and constant fluctuation nature. Conventional approaches to stock price forecasting generally involve the use of either fundamental or technical analysis, a bifurcation that rarely provides a holistic view of the forces that underlie the movement of stock prices. While technical analysis is primarily based on price and volume, fundamental analysis offers insights into a company's intrinsic value through financial data such as ratios, earning reports, and macroeconomic indicators. Exclusion of these aspects may cause a lack of comprehensive understanding of the stock behavior and thus lead to sub-optimal predictive results.

To overcome this limitation, this study proposes the incorporation of basic variables, including Price-to-Earnings ratio and profitability, into an enhanced deep learning model. Thus, we try to improve the predictive reliability beyond what can be achieved using the technical indicators alone.

The overall market perspective in the new model should lead to better and more accurate stock price predictions. The proposed model is going to be able to derive the power of technical and fundamental analysis to present helpful information to investors that may be relevant when deciding about investments. Thus, this study attempts to extend the existing literature by showing that stock market prediction can benefit from a fusion of technical and financial data in a deep learning model.

1.3 Aims and Objectives

This objective of this study is to improve the existing stock prediction model which attempts to incorporate technical and fundamental data into a hybrid deep learning model in the form of GRU and LSTM architectures. This improved model is expected to improve the accuracy and reliability of the results and help the decision-making process of the investors, analysts, and traders who are dealing in trading. The following objectives are proposed for this study:

a) To identify the shortcomings of traditional ARIMA, standalone GRU, and LSTM stock price prediction models.
b) To develop a hybrid model that integrates GRU and LSTM architectures with both fundamental and technical data, leveraging their complementary strengths.
c) Investigate the impact of varying the number of layers in GRU and LSTM architectures on the performance and accuracy of predicting stock price trends.
d) To evaluate the performance of the enhanced predictive model with evaluation metrics such as MSE, RMSE, MAE, and MAPE.

1.4 Research Scope

This research primarily aims at constructing a novel deep learning model based on GRU and LSTM that incorporates both the technical and fundamental data of the stocks in order to predict the stock prices. This research is intended to fill the gaps that are inherent in conventional and isolated DL models in modeling the dynamic and multifaceted nature of financial markets. In addition, it also examines the influence of incorporating basic data into predictive accuracy.

To achieve this objective, this paper adopted historical stock price data from the Yahoo Finance API for two particular stocks, 600719.SS and 000679.SZ. Besides, fundamental data, including Price-to-earnings ratio (P/E ratio) and profitability are also calculated for the same stocks.

Moreover, the study assesses how changes in the numbers of GRU and LSTM layers affect the performance of the model in terms of accuracy, computational cost, and capability to learn short- and long-term temporal dependencies. This analysis enables the research to suggest the best configuration for financial market applications.

Finally, owing to the constraints of the study, the number of stocks to be selected is restricted to two to allow controlled experimentation and detailed analysis to be conducted. However, the findings are expected to indicate the possibility of using hybrid GRU-LSTM models for stock market prediction to guide other studies involving more stocks and markets. This research seeks to apply technical and fundamental analysis to real-life financial predicting issues.

1.5 Thesis Organization

This paper is organized into five chapters, aiming to provide a systematic and logical flow of the research from the introduction, literature review, research methodology, results and discussion to the conclusion and future work. Each chapter builds upon the previous one to present a comprehensive study on the hybrid GRU-LSTM model for stock price prediction. The chapters are organized as shown in Table 1.1.

Table 1.1: Overview of Thesis Organization

Chapter	Content
Introduction	- Introduce the background of stock price prediction and its importance in the financial market. - Highlight the research problem, objectives, significance of study, and its scope. - Provide an overview of the structure of the thesis to guide the users along.

Chapter	Content
Literature Review	- Present a critical review of existing works related to stock price prediction. - Examine traditional models such as ARIMA, machine learning approaches, and deep learning techniques like LSTM and GRU. - Identify gaps in the literature and justifies the need for a hybrid GRU-LSTM model that integrates both technical and fundamental data.
Research Methodology	- Describe the research design, including data collection, preprocessing, and feature engineering steps. - Outline the GRU-LSTM model architecture, along with the configurations for varying GRU and LSTM layers. - Provide details on model training, validation and evaluation.
Results and Discussion	- Present the experimental results, including comparisons with baseline models such as ARIMA, standalone GRU, and LSTM. - Visualize and analyze key findings to demonstrate the effectiveness of the hybrid GRU-LSTM model. - Include discussions to link the results back to the research objectives and highlight the contributions of the study.
Conclusions and Future Work	- Summarize the key findings and contributions of the study. - Addresses the limitations of the research and provides recommendations for future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Stock prices prediction has been a difficult factor to crack in the financial markets for a long time due to the non-linearity and volatilities of the markets. Thus, attracting the attention of investors, analysts, and researchers. This paper focuses on the combination of GRU and LSTM techniques with fundamental and technical analyses to improve performance. This literature review comprises of the following sections Section 2.1: Introduction, Section 2.2: Stock Price Prediction Models, Section 2.3: Hybrid GRU- LSTM Architecture, Section 2.4: Fundamental and Technical Analysis, Section 2.5: Related Works, Section 2.6: Research Gaps, Section 2.7: Summary.

2.2 Stock Price Prediction Models

The prediction of stock market prices has always been an area of interest to both investors, analysts as well as researchers. In the last years, different approaches have been proposed to address it, such as statistical models, ML, and DL ones (Lu, 2024). The following section of this paper seeks to outline these models while noting down their development, advantages, and disadvantages.

2.2.1 Traditional Models

Traditional methods like the Autoregressive Integrated Moving Average (ARIMA) model which is a methodology of the traditional time series have for long been adopted in the prediction of stock prices because of their ease and efficiency. ARIMA works based on the outcomes and the related dependencies where the past values are inputted and the future values are produced. They are specifically useful when dealing with linear and stationary data patterns. Incorporating the given data from New York Stock Exchange (NYSE), National Stock Exchange of India Limited (NSE), and Indian markets, Mondal et al. (2014) has confirmed the effectiveness of the proposed method. The advantages of ARIMA include the ability to explain the interaction between arcs as it displays this dynamics well. In addition, it has good statistical base and performs well in short linear movements which is really important and useful.

For instance, Minhaj et al. (2022) stated that even though ARIMA has shown good results for the prediction of individual stock, it is not suitable for non-stationary or weakly stationary data. As a result, ARIMA tends to fail at identifying nonlinear trends and patterns, which increases its error rate in relation to market crashes, and the subjectivity of the parameters estimation may also lower the long-term forecast accuracy (Liu, 2024). In addition, the application of the ARIMA model is also limited in the processing of high-dimensional data. To this, it fails to add more variables like the fundamental and technical indicators to arrive at stock price prediction, making it less suitable for modern stock price prediction. In summary, ARIMA and similar statistical approaches will continue to be relevant as reference techniques for the forecasting of stock prices. However, due to high dynamics and volatility of financial markets, more sophisticated solutions are required.

2.2.2 Machine Learning Models

Machine learning approaches has become more important now due to the continuous growth of computational powers and size data sets as one of the most effective tools in the prediction of Stock Price movement compared to traditional approaches (Yang Shi, 2023).

Some specific ML algorithms, for instance, Support Vector Regression (SVR) algorithms and ensemble approaches like Random Forests (RF) and Gradient Boosting Machines have been used frequently. The high dimensionality and non-linear character, as well as the presence of more noise in stock markets, make SVR more appropriate (Chun Cai et al., 2012). SVR is used for capturing the non-linear dependencies with kernel functions, however, the method lacks scalability for big data and does not support sequential information input.

On the other hand, methods like RF and Extreme Gradient Boosting (XGBoost) are incredibly good in handling non-linearities related to features as well as the feature combination. While comparing with RF, SVR, and XGBoost, Sharma & Jain (2023) found that XGBoost was the most efficient model with MSE of 0.004.

However, these classical ML models do not include temporal dependencies as part of their disposition, which poses a limitation in the use of time series forecasting.

2.2.3 Deep Learning Models

The problem of stock price prediction has advanced substantially with the recent enhancements in DL methods, overcoming the drawbacks of the classical models and traditional ML methods. Out of all deep learning approaches, one model that stands out is the Recurrent Neural Network (RNN). The proposed RNN based architectures are found to be highly effective for sequence learning problems and offer better results than other architectures for different applications (Lipton, 2015). The general success of RNN-based models can be explained by the model's capacity to process inputs and outputs of variable lengths and the ability to learn both temporal features and convolutional perceptual representations simultaneously (Donahue et al., 2014).

Within the broader family of RNNs, two architectures have garnered particular distinction: LSTM networks and GRU. LSTM framework was proposed by Hochreiter and Schmidhuber (1997), it added a memory cell with the ability to include or discard information through the gating functions. As depicted in Figure 2.1, the cell state (c_t) runs through the cell horizontally, like a highway along which nformation can pass, enabling the network to maintain long-term dependencies. The forget gate decides what information to discard from the cell state, we use a sigmoid activation (σ) function for that. An input gate also using a sigmoid function determines what new information to write into the cell state and a tanh function creates candidate values that can be added. Last of all, the Output gate employs sigmoid activation function to establish control on what information in the cell state is allowed to influence the hidden state (h_t) at the current step of the loop. The hidden state is then passed back into the cell at the next timestep, allowing the model to retain sequences for extended set patterns. This architecture allows LSTM to maintain the temporal dependencies over the long horizon than the standard RNN and thus applicable to the financial time-series forecasting (Greff et al., 2017).

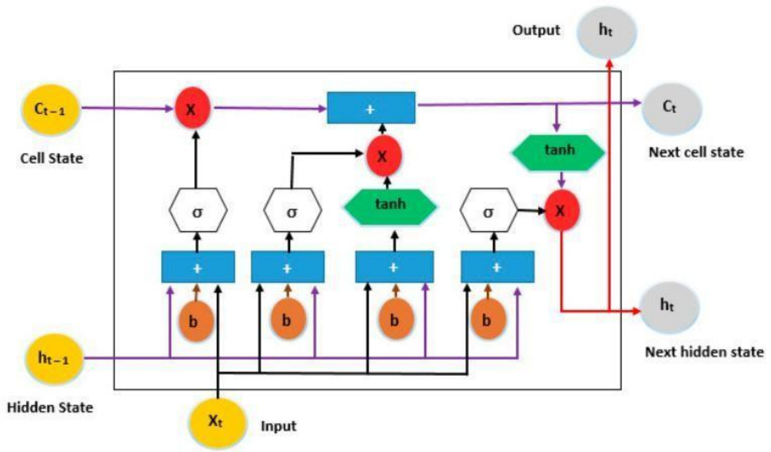

Figure 2.1: Internal Structure of LSTM (Hamayel & Owda, 2021).

Due to this ease of integration of temporal data, LSTM networks have been widely used in stock price prediction. Because LSTM networks are able to remember past states, they are good at learning trends or patterns that appear over long periods of time, which is very significant in stock price that is significantly influenced by past history. For example, Fischer and Krauss (2018) demonstrated that LSTMs can achieve superior prediction accuracy compared to classical machine learning approaches in forecasting the S&P 500 index8 as they are better able to learn temporal dependencies of stock prices.

Another positive feature of LSTM is that they are able to manage noise in the data. In most cases, stock prices are erratic: influenced by numerous factors that may simultaneously surface. LSTM networks are designed to minimize the noise and extract the underlying features of the data as seen in Ma (2020); Goyal & Kumar (2019) who adopted the LSTM networks in predicting stock prices and noticed that they outperformed the ARIMA and ANNs. Bathla (2020) further confirmed that in the context of the stock price forecasting task, the LSTM achieved higher accuracy than the SVR.

This could however be costly and time consuming to train a LSTM network although LSTM can be computationally intensive. The above model is relatively complex to estimate and thus consumes a lot of computational power, which may be a drawback to some users. Nonetheless, extensive research has demonstrated that the structured dropout can decrease the computation time while maintaining the same level of accuracy (Sarma, 2021). In addition, the LSTM networks are known to overfit resulting to high levels of accuracy but low generalization ability especially when trained with little data. This vice can be checked through the use of techniques such as regularization, dropout, and cross-validation. As mentioned before, for LSTM networks, preprocessing step such as handling missing data and outliers, is important for obtaining good quality data (Cao et al., 2020).

Next, taking a look at the GRU, another RNN architecture that is also used extensively in the forecasting of temporal data, similar to that of LSTM. While LSTM has relatively more parameters to solve the vanishing gradient problem, GRU is designed to deal with short but effective representation and is computationally efficient but still enables over long temporal dependencies. GRU has only two gates unlike LSTM having three gates : two gates: the reset gate, the update gate.

Specifically, the reset gate decides how much of the previous knowledge is discarded as fresh inputs are fed, while the update gate maintains how much of the fresh information becomes part of the memory and how much of the old knowledge is kept. Collectively, these gates enable GRU to control the amount of memory that flows in and out of the network on a cycle by cycle basis, ensuring that it is well suited for the processing of sequential data. For instance, a lower reset gate value results to more memory being passed over while a higher update gate value increases the model's ability of creating new information. If the reset gates are set to 1 and the update gates to 0, then a GRU applies to the formula of a general neural network model.

To be precise, GRU is indeed faster than LSTM as far as the number of parameters and training time are concerned because of its simpler structure. This efficiency is a plus in the GRU model making it suitable for use in processing the real-time data such as in real time stock price prediction where high speeds are essential. Nevertheless, the reduced complexity of the GRU architecture may reduce its effectiveness in tasks that involve more complex modeling of long-term dependencies – in this case, a dedicated memory cell, as in LSTM, will be useful. This efficiency makes GRU very suitable for real-time stock prediction use as noted by Zhang & Aggarwal (2021).

Figure 2.2 illustrates the internal structure of a GRU, showcasing its two gates: the reset gate (r_t) and the update gate (z_t). In particular, the reset gate defines whether the mechanism should forget the amount of the previous hidden state (h_{t-1}) when calculating the candidate hidden state, allowing for selective memory erasure. Update gate makes an effort in preserving the previous information (h_{t-1}) and information into the current hidden state (h_t).

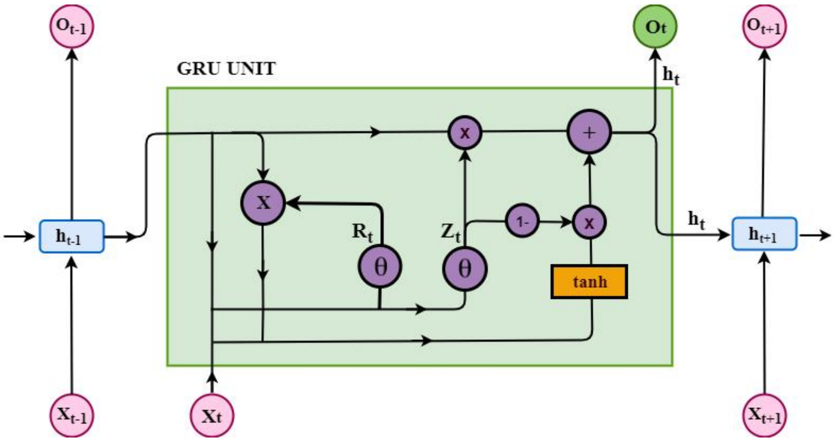

Figure 2.2: Internal Structure of GRU (Bibi et al., 2020)

The reset-modified hidden state to get the candidate hidden state is summed with the current input (x_t), passing through hyperbolic tangent (tanh) activation function. The final hidden state (h_t) is a combination of the previous and candidate state but the contribution of each state is regulated by the update gate. This gating mechanism makes GRU less computationally expensive than LSTM but still capable of capturing sequence information.

2.3 Hybrid GRU-LSTM Architecture

The hybrid GRU-LSTM has been a reliable predictive model that utilizes the merits of the GRU and the LSTM networks because of the unique feature to handle time series data properly. GRU and LSTM are both of the RNN type used for temporal data with each of the methods having unique characteristics that can suit certain functions efficiently.

Table 2.1 presents an overview of the research and application of hybrid GRU-LSTM models which is usable in versatile domains including but not limited to financial forecasts, traffic prediction, anomaly detection and, COVID-19 trends. This breadth indicates the versatility of a model in different types of time-series and in multi- factor environments.

In addition, more features are used in many studies, including weather, the time of day, and holidays (Zeng et al., 2022), or combine technical and fundamental indicators (Trivedi & Patel, 2022). This ability to connect external and heterogeneous data sources to the hybrid architecture further speaks to one of the models relative strengths, namely, its ability to model multiple-way dependencies and represent complex interdependencies within the data.

The table presented herein demonstrates that the GRU-LSTM combined model performs better than learner-only GRU, LSTM, or any other established model like ARIMA. One might expect such a superiority in metrics like MSE, RMSE or even MAPE to be attributed to the fact that GRU is more efficient than LSTM and at the same time the benefits from LSTM's capability in dealing with long term dependencies. For example, the hybrid model's performance was always better as compared to other conventional models.

To overcome challenges in their respective domains, specialists have modified the GRU-LSTM model. For instance, Al-kathari et al. in 2023 proposed a swarm intelligence enhanced GRU-LSTM for intrusion detection; Chen et al., in the same year, used attention mechanisms for improving cryptocurrency trend detection. These adaptations draw attention to the model's modularity and its ability to incorporate other optimization techniques for increased efficiency.

In all the other comparative studies, the hybrid architecture showed better overall forecast accuracy. This improvement is most apparent in areas such as financial markets by using big data (Islam & Hossain, 2020) and traffic systems where accurate predictions are crucial (Zafar et al., 2022). These outcomes confirm that the proposed model is effective when temporal relationships are important for decision-making.

Table 2.1: Summary of Literature Review on GRU-LSTM Hybrid Model Applications

Author(s)	Objective	Model	Key Findings
Islam & Hossain (2020)	Predict future closing prices of FOREX currencies.	Hybrid GRU-LSTM with 20 GRU and 256 LSTM hidden neurons.	Outperformed standalone GRU, LSTM, and SMA models in terms of MSE, RMSE, and MAE.
Trivedi & Patel (2022)	Forecast stock prices for HDFC Bank in India.	GRU-LSTM	Outperformed standalone GRU and LSTM in terms of error metrics.
Zeng et al. (2022)	Predict parking occupancy using multi-factor time series data.	Stacked GRU-LSTM model with external factors like weather, holidays, and time of day.	Achieved better accuracy and reduced prediction time compared to traditional and deep learning models.
Zafar et al. (2022)	Predict urban traffic speed using heterogeneous data.	Hybrid LSTM-GRU	Outperformed CNN, standalone GRU, and LSTM with lower RMSE and MAPE.
Liu et al. (2019)	Short-term stock price prediction for two stocks.	Regularized GRU-LSTM model with two GRU layers and one LSTM layer.	Regularization improved generalization and reduced overfitting, yielding superior performance over GRU and LSTM.
Prakash et al. (2023)	Forecast COVID-19 case trends in India.	GRU-LSTM hybrid compared with other hybrid models (CNN-LSTM, Bi-LSTM)	GRU-LSTM less effective than simpler models like LSTM for COVID-19 data trends.
Chen et al. (2023)	Predict cryptocurrency price trends.	GRU-LSTM hybrid with additional attention mechanisms	GRU-LSTM with attention achieved high accuracy, outperforming standalone and other hybrid models.

2.4 Fundamental and Technical Analysis

Consequently, the methods for stock prediction are chiefly divided into two; the fundamental analysis and the technical analysis (Huang et al., 2021). Both methods have their advantages and disadvantages, and the application of both has been proposed to improve the predictive capability for financial forecasting.

Qin and Boicu (2023) define fundamental analysis as the examination of financial statements of a company and general economic factors. This consists of evaluating the financial performance of the company, the performance of the employees, the board of directors, yearly company report, balances sheets and income reports, terrestrial and climatic factors inclusive of natural disasters, and bargaining political information. Some of the essential valuation ratios that people usually consider comprise the debt-equity ratio (D/E), Price/sales ratio (P/S), Price/book ratio (P/B), and Earnings per share (EPS) and the aspect of profitability. The rationale used with fundamental analysis is that stock prices will in the long run reflect their intrinsic value and understanding these fundamentals can provide insights into long-term trends.

Technical analysis, on the contrary, is the consideration of the data of the activity on the market such as the price fluctuations history, the sentiment, the flow of funds, and cycle. This method is favoured by technical analysts who wish to make short-term forecasts of the movement in stock prices. While on the other hand, fundamental analysts analyze a stock with the outlook to the financial health of the company, technical analysts are the ones who predict price of a stock in the light of its previous prices. Another benefit that is associated with the application of technical analysis is that it has a simple rationale and application. This is evidenced by such that it purely relies on price and volume trading data to project future prices. It has no regard to any form of economic factors, the prevailing market condition and or other factors that may influence this dynamic stock market, company health etc. However, the disadvantage of this strategy is that the user is locked in to that investment plan and is vulnerable in long term because it does not consider fundamental business health of company (Kaushik, 2024).

2.5 Related Works

A large number of the existing research studies that use machine learning for predicting stocks are rooted in technical analysis. Based on such studies, machine learning models accept price data or technical indicators extracted from prices of the past as inputs. Several writers attribute the increased use of technical analysis-based models because huge amount of stock's technical data is available as compared to the fundamental data which is available on quarterly basis.

A number of previous research works have used state-of-the-art deep learning methodologies for predicting the short-term stock prices. For example, Lu et al. (2020), Navastara et al. (2023) and Xue et al. (2023) used LSTM, Bi-LSTM, CNN and GRU models. This approach mainly incorporated price and technical history as inputs, and were said to have provided better forecasting functionality compared to statistical models. Interestingly, LSTMs where highly appreciated for dependency in time (Hamayel & Owda, 2021), whereas while using GRUs the similar accuracy was achieved with fewer parameters (Chen et al., 2020) which is perfect for conditions with low computational power.

Combining of fundamental analysis into forecasting models has also been an area of discussion. Subsequently, Zhu et al. (2021) integrated LSTM networks with the fundamental analysis indicators and stated that the forecast accuracies were exhibited higher when compared to just the basic LSTM. Yang (2020) extended this approach equally by including the sentiment analysis which make use of the social media data to understand market sentiment and also enhance the predictive capability. In addition, in order to combine the signals of multiple models, including technical and fundamental analysis models, such techniques as Random Forest and Gradient Boosting have been used. Both techniques have been reported by Wang et al. (2021) to improve the prediction robustness and minimize overfitting.

Both Bohn (2017) and Goel and L. (2024) utilized all three methodologies namely technical analysis, fundamental analysis, and sentiment analysis for arrival at stock prediction. Bohn (2017) benchmarked one set of machine learning models for long term stock price prediction while Goel and L. (2024) did short term prediction.

Bohn (2017) conducted an experiment involving approximately1500 stocks that appear in the S&P 500 between 2002 and 2016. Linear regression models were obtained and ranks were created depending of the model's prediction for each of the validation and test week.

He then used the Spearman rank correlation coefficient between the predicted rank and actual rank as the measure of model performance. These findings indicated that the neural network model employing iterative feature selection was capable of producing accuracy comparable to a model developed using domain specialist knowledge from an investment firm. Likewise, in Goel and L. (2024), regression model formulation was to forecast the next day's stocks whereas time-series model formulation was to forecast the next five days stock price.

By using LSTM and GRU, it is possible to overcome one of the obstacles of RNN, with regard to gradient disappearance. Furthermore, these two models have their uses in areas like computer vision. For example, Zafar et al., (2022) developed a traffic speed forecasting model based on the LSTM-GRU model in urban regions. Zafar et al. (2022) decided to compare its hybrid model with LSTM, GRU, CNN, LSTM-CNN, and CNN-LSTM. The current study shows that all the developed models have their biases in the forecasts, and the hybrid GRU-LSTM model has the best performance in RMSE, MAE and MAPE. Per excellence, these two models have their own strengths when it comes to forecasting of stock price index. For example, Liu et al. (2019) proposed the regularized GRU-LSTM for predicting stock prices of companies. It was clear that new model hybridization rated better than both the GRU and LSTM standalone model with an RMSE score of 0.252. Table 2.2 displays a review on existing LSTM-Based stock prediction literature and their gaps.

Table 2.2: Review on Existing LSTM-Based Stock Prediction Methods

Author(s)	Method Used	Dataset	Challenges/Limitations	Future Work
S. Salimath et al., (2022).	LSTM-GRU	Indian	Only 25 companies were considered for their study.	-develop a website -collect real-time data
P. Sandhya et al., (2022).	LSTM-RNN	Yahoo Finance	There was no discussion of data fluctuations or unusually high or low figures.	Incorporate other data like global changes, natural catastrophes, etc.
U. Bisarya et al., (2022).	CNN-LSTM	S&P 500	MAPE only at 1.788%.	Graph NN-based models would have yielded more accurate results.
Liu et al., (2019)	Regularized GRU-LSTM	Dalian Thermal Power (600719) and Dalian Friendship (000679)	Poor stability of the model.	Find optimization methods to increase stability of the model
Xue et al. (2023)	Multi-branched LSTM	Shanghai Composite Index	Lowest accuracy rate is only 37%.	-Expand the pool of technical indicators -Use multi-LSTM network to make join prediction
Lu et al. (2020)	CNN-LSTM	Shanghai Composite Index	Only considers the impact of stock price data on closing prices.	Incorporate sentiment analysis of stock- related news
Navastara et al. (2023)	CNN-BiLSTM	Indonesian Stock Exchange	Focus on one single feature only.	Incorporate a broader range of features exploring alternative hyperparameter selection.
Y. K. Saheed et al., (2022).	LSTM-GPU	S&P 500, NYSE, Nasdaq	Average performance with RMSE=1.78, MAPE=0.3	Novel DL-RNN model could have given more promising results.

2.6 Research Gaps

While more and more researchers use modern machine learning in the field of financial forecasts, no significant studies have been done to combine fundamental analysis with machine learning. Most past studies rely on technical analysis because technical data is more accessible and more frequently developed. This has led to one form of data, the fundamental data that is published less often, being used less than technical data. The scarcity of research related to fundamental analysis joined with machine learning evokes the conclusion regarding the existence of more standardized models that involve fundamental financial characteristics for the long-term movement of shares' prognosis (Qin & Boicu, 2023).

After that, a number of studies pinpoint the flaws that stem from the quality and quantity of available financial data. For example, Khanapuri et al., (2024) affirmed that the small amount of data limits the efficiency and effectiveness of stock price prediction models. Smaller and less diverse datasets may affect not only the validity of the identified models but also the generalization and stability of the findings. Also, the use of standard deviation in risk measures makes it important to adopt a more accurate stock covariance matrix analysis to increase prediction precision and reliability.

Since the stock market possesses a high level of non-linearity which makes many existing models to be insensitive to external variables such as geopolitics or technologies affecting stock markets. Most of the existing studies can be noted to be giving attention to internal financial values without accounting for contextual factors. As for the further scientific investigation, such external factors to be considered and incorporated into the picture to enhance the effectiveness of the stock market predictions derived by the analytical models (Khanapuri et al., 2024).

LSTM and CNN as the representative of complex neural network models, typically have two drawbacks: overfitting and non-interpretable. However, even to capture these kind of complex patterns these models can be effective yet methods become a 'black box' and it could be difficult to establish a clear correspondence between features and future values. In the future, many studies should be aimed at creating reliable models for explaining the results of the random forest decision-making process, which will help investors increase confidence and use the model (Khanapuri et al., 2024).

The absence of best practices in applying machine learning to fundamental analysis can be evidently seen. First of all, the procedures used to integrate fundamental analysis with ML are largely dispersed. In order for there to be a generalizability of the studies' findings, the methodologies used by researchers are different as well as data collected, thereby making it difficult to compare research studies and arrive at well-specified conclusions. Setting up norms for model creation, test, and assessment would enhance the reproduction levels in research findings (Qin & Boicu, 2023).

2.7 Summary

The literature review looks into the combination of GRU and LSTM models with fundamental analysis when it comes to improving the accuracy of stock prices, a problem that has been of interest to investors, analysts, and researchers for many years. The review focuses on showing that stock price prediction has gone through a transition from conventional approach to contemporary AI apparatus. The foregoing hypotheses imply that stock prices are intrinsically random and efficient, in the sense that the information is fully reflected in stock prices. Nevertheless, recent empirical research shows that markets, particularly, emergent ones, are not completely efficient, thus, implying that the selection based on the obtained data can be better than the random one.

LSTM and GRU models' ability to take time-series data and capturing long-term dependencies built into a particular set of values is applicable to stock prices in particular, thus enabling their use in such work. The traditional RNN has good capability to avoid noise and has the ability to identify the deeper patterns of data but have high complexity and are very sensitive to overfitting problems. On the other hand, GRU networks are better in terms of performance because of its simpler architecture and can be used in real time applications.

The review also evaluates research that has implemented machine learning in stock prediction, primarily using technical analysis. It has been evident that studies using models such as LSTM, Bi-LSTM, CNN, and GRU have had different levels of impact. Attention is also being paid to the further synthesis of fundamental indicators with machine learning algorithms. Those papers that use LSTM networks along with fundamental analysis have found better result in accurate predictions of prices and ensemble ranger methods like RF and Gradient Boosting have indicated the efficacy to make robust predictions.

Nevertheless, there are still study gaps which need to be addressed. This is the case because most of the work done is technical work since there is more data available compared to fundamental work. Another drawback leading to the variability of the results of the models and their inability to generalize is the quality and quantity of financial data. Moreover, previous theories may exclude such variables, for example, global political climate and technological changes. In applying machine learning to fundamental analysis, current complex neural network models face the problems of overfitting and lack of interpretability; however, there is still no generally accepted set of best practices.

To fill in these gaps, this research introduces the feasibility of combining the GRU and LSTM models with fundamental and technical data to supply the superior stock price forecasts to assist investors more in their decision-making.

CHAPTER 3

RESEARCH METHODLOGY

3.1 Introduction

This methodology for this thesis focuses on integrating technical and fundamental analysis within a hybrid GRU-LSTM model to predict stock price trends. This chapter is composed of eleven sections that include 3.1 Introduction, 3.2 Research Design, 3.3 Data Collection, 3.4 Data Preprocessing, 3.5 Model Architecture, 3.6 Model Training and Validation, 3.7 Model Evaluation, 3.8 Implementation Details, and 3.9 Summary.

3.2 Research Design

The research employs a quantitative and experimental methodology that utilizes machine learning techniques like GRU and LSTM to predict stock prices. Figure 3.1 shows the overall workflow in a pipeline, starting from data acquisition, going through systematic preprocessing and ending in development, training and evaluation of the model. The proposed framework mainly consists of three modules including Data Module, Pre-Processing Module and GRU-LSTM Module.

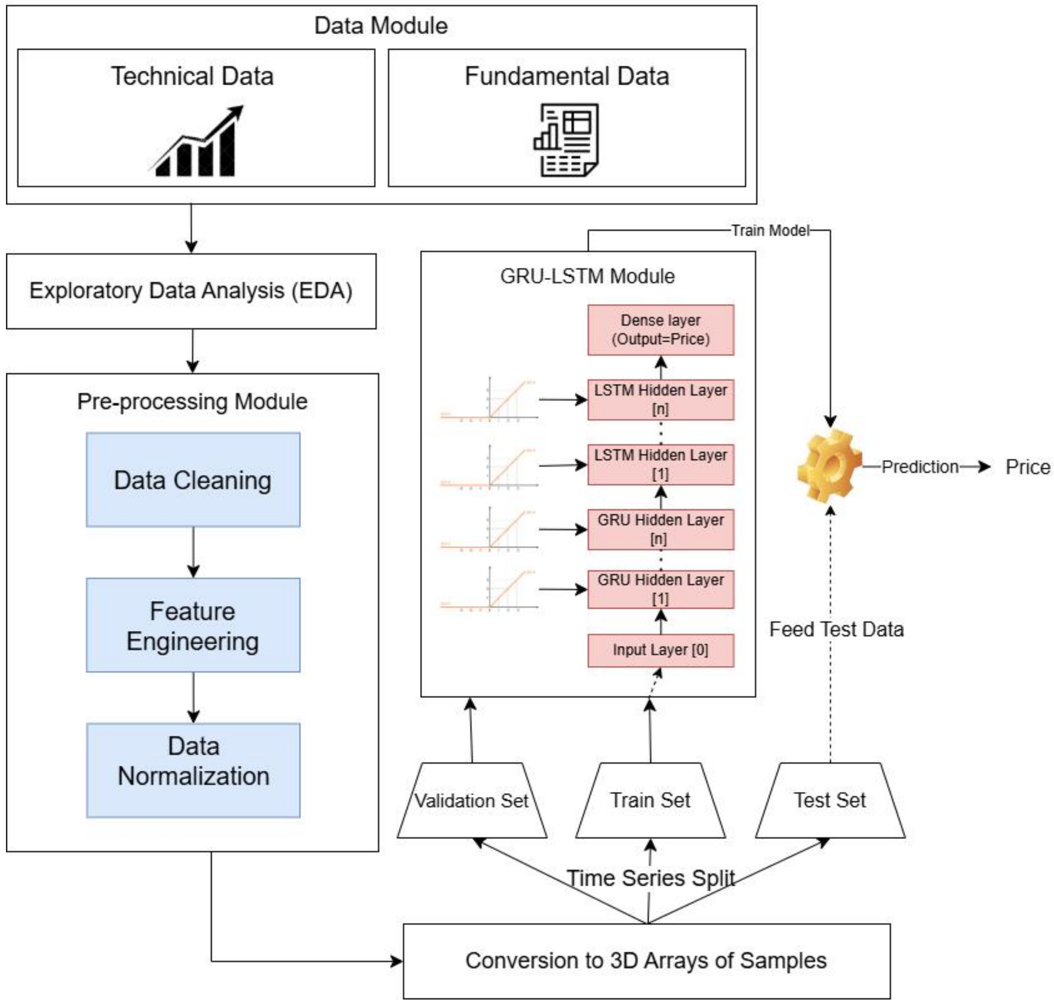

Figure 3.1: Overview of Research Flow

i. **Data Module:** Both technical and fundamental data sources are collected to ensure a holistic representation of market conditions. Technical data include factors such as price history, volume, and derived indicators (e.g., Moving Averages, MACD, Stochastic Oscillator, and RSI). While fundamental data that include P/E Ratio and profitability are also calculated. These raw datasets are subjected to Exploratory Data Analysis (EDA) to discover distributions, find outliers and decide on feature engineering.

ii. **Pre-processing Module:** After EDA, the preprocessing module transforms the data by filling nulls, filtering outliers, and standardizing formats. Then time-series indicators are extracted through feature engineering. For example, the the MACD lines, signal line, and RSI are derived to allow for richer temporal context that helps the model learn long-term dependencies. The data is then normalized using the MinMaxScaler to make sure all the features are on the same scale so to increase model stability and training efficiency.

iii. **GRU-LSTM Module:** The forecasting model consists of a hybrid GRU–LSTM architecture to extract short-term variations and long-term trends in the time series. GRU layers effectively learnt shorter-term dependencies but longer-term memory was recognized by the LSTM layers, the dense layer outputs a single price predicted. In this module, different configurations of GRU and LSTM layers are also tested to find the optimal configuration.

iv. **Training and Validation:** As the model is trained on a training set, weights are adjusted iteratively across timesteps to minimize a loss function. Validation set is used after every epoch of training, which acts as a checkpoint during development to see how well the model is performing on unseen data. To reduce training time, early stopping and learning rate schedulers are used in combination with our models to improve generalization.

v. **Testing and Evaluation:** After tuning the model's hyperparameters and architecture, a final evaluation is performed on the test set. In this stage, the model’s predictive power and robustness in factually unseen market conditions is tested. Performance is quantified with metrics such as MSE, RMSE, MAE, and MAPE. The predictions and metrics obtained from those files are then used for analysis of the model performance, comparing the models and for potential improvements in the future.

3.3 Data Collection

Data has been collected for this study using Yahoo Finance, which provides historical stock data for various stock exchange listed companies worldwide. The data that is being collected includes the stock prices, volume, and a range of technical indicators to create a basis for training the predictive model. Here is an outline of the data collection process:

The two chosen stocks represent the leading financial markets within China. The first one is Dalian Friendship(Group)Co.Ltd.(000679), publicly traded on the Shenzhen Stock Exchange; and the second one is Dalian Thermal Power Co.,Ltd.(600719), which is publicly traded as well on the Shanghai Stock Exchange. Y. Liu et al. (2019) suggested that they are two of the leading financial centers in China that provide distinctive qualities and market environments. This guarantees a wider outlook of the Chinese stock market and therefore improving the efficiency and generalization of the predictive model.

The historical data for the stocks were retrieved from the Yahoo Finance API, for the period of 5 years for 000679 and 10 years for 600719 due to data unavailability for stock 000679. The data covers a long enough period to incorporate a range of market conditions, including bull and bear markets. Python script in Figure 3.2 gets two stocks historical market data from the Yahoo Finance API. The data have been stored for preprocessing and for analysis.

```python
# Get the data for the specific ticker
data = yf.Ticker("600719.SS")

# Fetch data for the specific date range
data = data.history(start="2014-12-08", end="2024-12-06")
```

```python
# Get the data for the specific ticker
data = yf.Ticker("000679.SZ")

# Fetch data for the specific date range
data = data.history(start="2019-12-07", end="2024-12-06")
```

Figure 3.2: Code Snippet for Data Collection from Yahoo Finance API

The retrieved dataset is used to predict stock price, including the attributes located in Table 3.1. Every column shows one of the features of the data they are necessary for preprocessing, feature extraction and model development.

Table 3.1: Description of Columns in the Retrieved Dataset

Columns	**Description**
Open	Opening price of the stock for the day
High	The highest price recorded during the day
Low	The lowest price recorded during the day
Close	The closing price of the stock
Volume	The total number of shares traded during the day
Dividends	The amount paid per share to shareholders as a distribution of profits.
Stock Splits	Instances of stock splits during the recorded period

The first five rows (head) and last five rows (tail) of the dataset were checked to get a basic idea of what the dataset looks like. This gives an insight into the structure, their columns, and the initial values. The example output can be seen in Table 3.2 and Table 3.3, displaying the records from the retrieved dataset along with its salient features including Open, High, Low, Close, Volume, Dividends, and Stock Splits.

Table 3.2: Head of Retrieved Dataset

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
2014-12-10 00:00:00+08:00	4.472044	4.872304	4.457219	4.867362	33190136	0.0	0.0
2014-12-11 00:00:00+08:00	5.074904	5.287388	4.897011	4.985958	41764806	0.0	0.0
2014-12-12 00:00:00+08:00	4.916777	4.921718	4.768532	4.847596	20167496	0.0	0.0
2014-12-15 00:00:00+08:00	4.793240	5.069963	4.724059	5.025490	24678240	0.0	0.0
2014-12-16 00:00:00+08:00	4.882187	4.911836	4.817948	4.862421	19750792	0.0	0.0

Table 3.3: Tail of Retrieved Dataset

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
2024-12-04 00:00:00+08:00	8.14	8.14	7.84	7.90	29012454	0.0	0.0
2024-12-05 00:00:00+08:00	7.88	8.18	7.83	8.08	29140540	0.0	0.0
2024-12-06 00:00:00+08:00	7.99	8.19	7.98	8.06	26129010	0.0	0.0
2024-12-09 00:00:00+08:00	8.03	8.87	8.00	8.52	75950422	0.0	0.0
2024-12-10 00:00:00+08:00	8.52	8.74	8.27	8.32	55824305	0.0	0.0

Fetching the first five rows of the data using data. head(), the output shows the first few records of the dataset from December 10, 2014. These rows give an indication of how the start of the dataset looks. Meanwhile, the last five rows of the dataset correspond to the data points at the end of the dataset, until December 10, 2024. This allows to make sure that the dataset does cover the intended 10 year range.

The dataset has 2430 rows and 7 columns as shown by the data.info() function. This is to make sure we have enough data that can be used to train, validate and test the machine learning model. This database extends far beyond price and volume information, including corporate actions such as dividends and stock splits.

First, you need this very extensive data to build your final prediction model with technical and fundamental analysis. Unfortunately, for the two selected stocks, these two columns are empty.

3.4 Data Preprocessing

After exploratory data, the next step is data preprocessing, an essential process in data analysis and model training process in order to clean and prepare the raw data to create a structured dataset with meaningful features. In this study, the raw stock data was preprocessed using multiple preprocessing methods, to achieve maximum performance of the hybrid GRU-LSTM model.

3.4.1 Data Cleaning

The data analysis began only after thorough inspection to ensure the quality and consistency of the dataset. In this stage, the dataset was inspected for missing values, duplicate entries, and inconsistencies. Using functions such as data.isnull(). sum() and data.info() and it confirmed that there were no null values in the dataset so it was a clean data. Furthermore, to further optimize the dataset, unnecessary columns like Dividends and Stock Splits, which did not offer substantial value to the model's prediction, were eliminated. The dataset was filtered to include only relevant columns for further analysis. Fro example, the Date column is dropped since it does not add numerical value to create additional features, hence are not useful in improving the predictive performance of the model. This preprocessing step helped maintain the relevance and accuracy of the subsequent modeling efforts.

3.4.2 Feature Engineering

This study's preprocessing method extensively relied on feature extraction, which is critical to augment the predictive potential of the hybrid GRU-LSTM model. In order to have sophisticated insights into stock price trends, different technical and fundamental indicators were calculated and added to the dataset. The first input features that would be fed into the model were these indicators which are used to create a quantitative picture and were used to measure market conditions.

The Stochastic measures the relative position of the closing price within its price range over a given time period. The calculation was done using rolling min and max prices over the designated period (k) and the %K stochastic value as the relative location of the closing price. The %D stochastic value was then calculated using a moving average of %K over another specified period (d). These are essential for detecting overbought and oversold conditions, which are commonly indications of potential price reversals. The formula of %K and %D stochastic features are as shown in (3.1) and (3.2).

$$\%K = 100 \times \frac{Close - Low_k}{High_k - Low_k}$$

Where,

- Low_k =Lowest price in the last k periods
- $High_k$ =Highest price in the last k periods (3.1)

$$\%D = SMA_d(\%K)$$

Where,

- SMA_d =SMA of %K over d periods (3.2)

The momentum of stock prices would also be analyzed by extracting the MACD indicator. This is done using the Exponential Moving Averages (EMA) for short (12-period) and long (26-period) price trends. It calculated the MACD line (the difference between these two EMAs) together with the MACD signal line, the 9-period EMA of the MACD line. It helps indicate bullish and bearish movements through crossovers of the MACD line and signal line. The computation of MACD and Signal Line follows the equation (3.3) and (3.4).

$$MACD = EMA_{12} - EMA_{26} \tag{3.3}$$

$$SignalLine = EMA_9(MACD) \tag{3.4}$$

Furthermore, to smooth out the various prices and catch short-term and long-term trends, moving averages — specifically the 50-period and 200-period EMAs and Simple Moving Averages (SMAs) — were also calculated. Where the EMA values were computed with exponentially weighted moving averages and SMA values were computed with simple rolling average over their periods. It also means these indicators can help with trend analysis as they are very useful for finding important stock price resistance and support levels. Equation (3.5) is the formula for SMA and equation (3.6) is the formula for EMA.

$$SMA_n = \frac{\sum_{i=1}^{n} \Pr ice_i}{n} \tag{3.5}$$

$$EMA_n = \alpha \cdot \Pr ice_t + (1-\alpha) \cdot EMA_{t-1}$$

Where,

- $\alpha = \frac{2}{n+1}$ is the smoothing factor (3.6)

The Relative Strength Index (RSI) was also calculated to measure the speed and change of price movements and can serve as a momentum indicator. It is an indicator used to assess if an asset is overbought or oversold. The RSI was calculated as follows: first, for each day it finds the price changes, then it calculates the 7-day rolling averages for gains and losses. Finally, calculate the RSI as the ratio of average gains to average losses. By default, this option will provide the insight into the stock buy/sell/hold signals. The formula is shown in equation (3.7).

$$RSI = 100 - \frac{100}{1 + \frac{AverageGain}{AverageLoss}} \tag{3.7}$$

For the selected stocks used in this study, detailed fundamental data is not available. So to include fundamental variables in the model, proxy variables were constructed. Hence the proxy variables—P/E Ratio and Profitability—were taken from the technical data points and sufficed as a nearly relevant metric.

The P/E ratio was calculated as the ratio of the closing price to the volume. This proxy was employed to simulate the price-to-earnings ratio, which is a fundamental indicator of a stock's valuation. This proxy strikes a balance between simplicity and relevance in the absence of actual earnings data. It measures how the market perceives the stock's valuation based on trading behavior, which is consistent with how traditional P/E is intended to be used.

Profitability was determined based on the daily relative change in the closing price defined as the closing price difference between two consecutive days divided by the previous closing price. More precisely, the calculated profitability is the daily returns of the stock, and shows the trend of short-term performance. These are interpreted such that positive values are gains and negative values are losses.With no profit and revenue data, this proxy expresses the dynamics of stock performance over time. Daily returns are an important metric for investors, so this proxy adds meaningful information to the feature set.

The computed features were combined with the validated dataset as the new input to the GRU-LSTM model. Including such technical and fundamental indicators, the study used domain-specific knowledge to enrich the feature set, hence, enhancing the model's proficiency in predicting stock prices.

3.4.3 Data Normalization

In order for both layers of the GRU-LSTM model to fit correctly for the training and testing phase, data normalization is performed using Min-Max Scaler. This maps the input features (x) and target variables (y) to be between 0 and 1. Normalization lessens the effect of the feature magnitude and leads to a speeding up of convergence during the train. The formula for Min-Max normalization is:

$$X_{scaled} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

Where:

- X : Original feature value.
- $X_{\min}$: Minimum value of the feature in the dataset.
- $X_{\max}$: Maximum value of the feature in the dataset.
- X_{scaled} : Normalized value. (3.1)

This normalization adjusts the data, such that 0 corresponds to its minimum and 1 corresponds to its maximum, in the range of [0, 1]. Additionally, it ensures that all features equally contribute to the training, preventing features with larger scales from dominating the others. The use of large feature values also results in reduction of gradient oscillations which will help in faster convergence of model training in turn. Activation functions work even better if the input data is normalized.

3.4.4 Data Splitting

The time series split method is used here to create a cross-validation strategy for time-series data. While standard k-fold cross-validation creates random splits of data, the Time Series Split preserves the order in time of data. This is particularly important for time-series tasks to avoid information leakage from the prediction of future data points to past data points. This is vital in time-series tasks, as retaining temporal integrity is necessary to avoid data leakage.

In this study, the data is split into 5 folds. In each split, a portion of the dataset will be used for training and the remaining part for validation, ensuring that the test set always follows the training set in time. Specifically, 80% for the training set and 20% for test set. Then, retrieving 20% from the training set for validation. This approach ensures realistic evaluation by mimicking the real-world scenario where models are applied to unseen future data.

The model gets trained on the training set and evaluated on the validation set. Results are saved in the results list. Separating these components normally provides some temporal ordering over training versus testing, but preserves the integrity of a time-series concept.

3.5 Model Architecture

The model that is proposed in this paper uses a hybrid architecture of combining GRU and LSTM layers as illustrated in Figure 3.3. Y. Liu et al. (2019) suggested that the hybrid model combines both GRU and LSTM layers in a way to take advantage of the 'best of both worlds' where GRU provides computational efficiency and LSTM provides robust handling of long-term dependencies. Input layer, GRU, LSTM, and dense output layer comprise the architecture of the model to process sequential data efficiently to predict stock price.

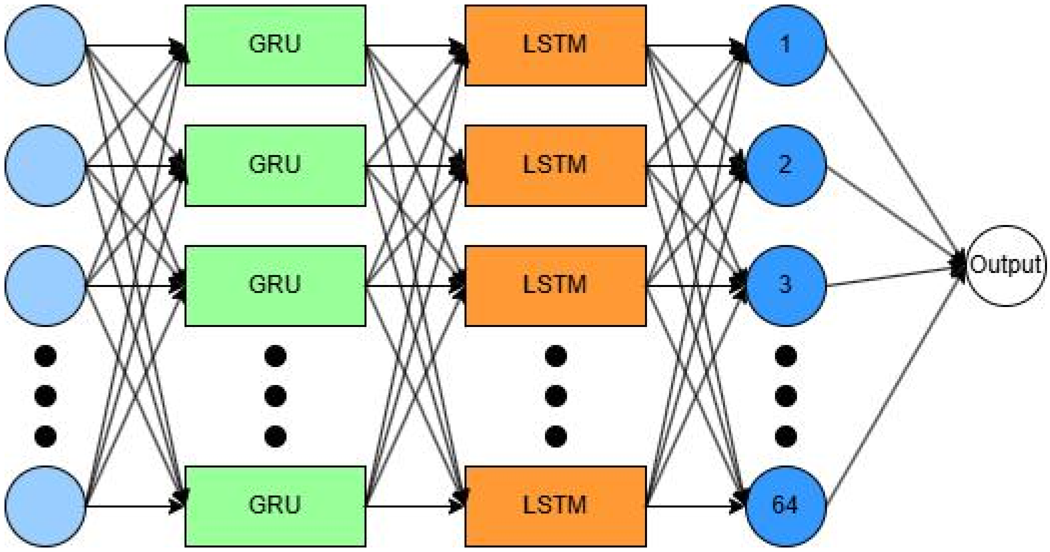

Figure 3.3 Hybrid GRU-LSTM Model Architecture

The model receives 3D time-series data in an input layer. This means we define an input shape of (None, 1), where None can be of variable length and 1 is the single feature input. This structure allows for compatibility with sequential data formats required by recurrent layers.

The number of GRU layers is dynamically generated from the parameter gru_layers. A Rectified Linear Unit (ReLU) activation function has been applied to each of the 64 units contained within each GRU layer, affording the model the ability to learn non-linear characteristics of training data. Finally, a drop out rate of 0.2 is applied to prevent overfitting. Next, return_sequences=True will allow for passing intermediate outputs to the next layers which is useful in capturing the sequential dependency. These layers quite successfully capture short term correlations in the time-series data.

Sequentially after GRU layers, LSTM layers are added to extract more complex sequential dependencies and long-term interactions. The author dynamically set the number of LSTM layers being made with the lstm_layers parameter. Two LSTM layers with drop out are used to mitigate overfitting. By combining GRU and

LSTM layers, a well-balanced architecture is built as it learns the short as well as long term pattern of the data.

The last part of the architecture is a dense output layer (regression layer) which outputs the only output, that is the predicted stock price. Because it is training, the dense layer uses the MSE loss function to predict accurately. The author employs the Adam optimizer, which adjusts the learning rate dynamically to guarantee convergence.

In contrast to Y. Liu et al. (2019), their model contains only single configuration, two GRU and one LSTM layers, this paper will assess the performance of different layer configuration. The model is trained and evaluated on different configurations with respect to the number of GRU and LSTM layers, which vary from one to nine layers. This helps improve generalization and prevent overfitting, and early stopping is used based on the validation loss with a patience of five epochs. The learning rate is also reduced using ReduceLROnPlateau with min lr = 1e-6 to prevent the model from getting stuck in local minima during training.

Lastly, the model is evaluated based on multiple metrics to assess its performance: MSE, RMSE, MAE and MAPE. These metrics provide an understanding of how well the model can predict. Furthermore, predictions are charted against ground truth to visualize the performance achieved with different model parameters.

3.6 Model Training and Validation

The process of training and validation is crucial, as it ensures the GRU-LSTM hybrid model is able to learn the patterns in stock price data and generalize to unseen data effectively. In this section, we will describe the training setup, our validation strategy and how we optimized the model and ensured that our predictions are robust and accurate.

The GRU-LSTM model is then trained with a prepared dataset that is split into training, validation and test sets in order to improve the models performance. The dataset is divided into three parts: 64% for the training set, in which the model weights are learned; 16% for the validation set, used to check and fine-tune the hyperparameters and monitor overfitting; and the last 20% using the test set, only used to evaluate predictive performance. In addition, input data is transformed to 3D arrays compatible with the order of time of GRU and LSTM layers. That is, the three-dimensional structure is defined as number of samples, number of time steps, number of features.

The training is done with a well-optimized configuration so that learning is efficient and effective. The Adam optimizer is employed to minimize the loss function, adjusting the learning rate dynamically during training as indicated in Table 3.4. The batch size is set to 64 for efficient computation, while the model is trained over 20 epochs to achieve sufficient convergence without overfitting. Besides, MSE loss function is used in this task to minimize the difference between predicted stock prices and actual stock prices. This architecture enables our model to learn relationships from the data and subsequently retain them with a minimal overhead.

Table 3.4: The Parameters of Model

Parameter	**Value**	**Parameter**	**Value**
Learn rate	0.001	Optimizer	Adam
Batch_size	64	Epochs	10,20,30
Loss function	MSE	Dropout	0.2
Activation Function	ReLU	Hidden Units	64 (GRU), 50 (LSTM)
Output-Type	Regression	Output-Units	1

3.7 Model Evaluation

The evaluation of the model is important to validate its prediction accuracy and robustness. In this study four commonly-used evaluation metrics are utilized in order to measure the model's performance which are MSE, RMSE, MAE, and MAPE. As discussed below, each metric provides unique insights into the model's effectiveness at forecasting stock prices:

1. **Mean squared error (MSE):** The average of the squared differences between predicted and actual values. Equation (3.1) shows it formula.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2$$

Where:

- y_i : Actual value
- $\hat{y}_i$: Predicted value
- n: Number of data points (3.1)

2. **Root mean squared error (RMSE):** An estimation of the residual between the actual value and predicted value. Equation (3.2) shows its formula.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)} \tag{3.2}$$

3. **Mean Absolute Error (MAE):** This index measures the average magnitude of the errors in a set of predictions, without considering their direction. Equation (3.3) shows its formula.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)$$

(3.3)

4. **Mean absolute percentage error (MAPE):** This metric is an indicator of an average absolute percentage error; lower MAPE is better than higher MAPE. Equation (3.4) shows its formula.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$$

(3.4)

3.8 Implementation Details

Different libraries in Python that aid in data processing, visualization, model development, and evaluation are used for the implementation of this study. These libraries play a vital role in managing the workflow of the project and ensuring that the GRU-LSTM hybrid model is functioning accurately. Since it was possible to use some libraries to reduce the implementation effort, Table 3.5 shows the libraries used in this study with their specific tasks and contributions to the them.

Table 3.5: Libraries Used and Their Task

No.	Library	Task
1	NumPy	Array operations and numerical computations
2	pandas	Reading, cleaning, and manipulating tabular data
3	seaborn	Statistical data visualization and advanced plotting

No.	Library	Task
4	matplotlib	General 2D plotting of charts, color maps, and data visualization
5	datetime	Handling date and time objects, conversions, and indexing
6	yfinance	Fetching financial market data such as stock prices directly from Yahoo Finance
7	math	Evaluating model performance using various error metrics
8	prettytable	Mathematical functions like square root for error metrics calculations
9	scikit-learn metrics	Displaying tabular data in a visually appealing and formatted way
10	plotly	Creating interactive, web-based visualizations and subplots
11	scikit_learn_model_selection	Splitting time-series data into training and testing subsets
12	scikit_learn preprocessing	Normalizing and scaling data before model training
13	TensorFlow and Keras models	Defining and running deep learning models
14	TensorFlow Keras layers	Building, customizing, and stacking neural network layers in the model

3.9 Summary

This study is a novel contribution to the literature as the research methodology followed in the study provide a systematic process on how to combine technical and fundamental analysis into a hybrid GRU-LSTM model for stock price prediction. It also experiments with different configurations of GRU and LSTM layers. The methodology involves several stages, starting from data collection, preprocessing, and feature engineering, followed by designing the model architecture, training, validation, and finally evaluation.

For data collection, the Yahoo Finance API was used to retrieve the historical stock data for 2 stocks in the Chinese stock exchanges Dalian Friendship (000679) and Dalian Thermal Power (600719). It consisted of technical features like price history, trading volumes, and corporate actions such as dividends and stock splits. This approach provided a comprehensive representation of the varying market conditions to effectively assess the model.

Data was pre-processed for cleaning up and outlier checks, as well as using different data types for features to create a uniform format. This step is very crucial due to the need for feature engineering which was critical to incorporate technical indicators like Stochastic Oscillator, MACD and RSI along with moving averages. Adding these attributes provided time and trend-based insight to the dataset which aids in accomplishing predictive modelling with more accuracy. Data normalization is also done to ensure the uniformity of the distribution between each feature that helps in training at a faster rate.

The advantages of both recurrent layers were provided through a GRU-LSTM hybrid architecture. GRU layers took care of short-term dependencies, because they were computationally efficient, and LSTS took care of long-term dependencies. This architecture consisted of variable stacks of GRU and LSTM branches, dropout layers to prevent overfitting, and dense output layers for precise prediction.

The respective models were trained and validated using a split dataset with an 80%-20% ratio (using a time-series split method to maintain the chronological aspect. Model optimization was improved through hyperparameter tuning and early stopping and learning rate adjustments. Predictive accuracy was evaluated using performance metrics such as MSE, RMSE, MAE and MAPE.

Thus, this systematic approach that integrates advanced neural network models with extensive data preprocessing and evaluation techniques highlights the contribution to the field of financial forecasting, providing accurate and reliable stock price predictions.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter presents the results of stock price predictions of the proposed GRU-LSTM hybrid model. Subsequently, these findings are used to address the research questions; specifically, how ARIMA, the stand-alone GRU, and LSTM model perform, the impact of fundamental and technical indicators, and the impact of the GRU and LSTM layers when used independently.

The baseline models are first introduced and compared against the proposed GRU-LSTM hybrid model in this chapter so the benchmark can be established. Next is an assessment of model performance, with metrics like MSE, RMSE, MAE and MAPE used. For a more detailed analysis, the results presented to further analyze the outcome are focused on two of the stocks of interest, 600719 and 000679 comparing between the different GRU and LSTM layers. To demonstrate the efficiency of the model as well as its stability, the results are shown in a series of tables, figures and graphs. Lastly, the results were discussed to elaborate on their relevance for emerging patterns and to establish the link with the research objectives and previous research. This chapter is composed of eight sections that include 4.1 Introduction, 4.2 Feature Analysis, 4.3 Evaluation of Baseline Models, 4.4 Performance of GRU-LSTM Hybrid Model, 4.5 Effect of GRU and LSTM Layer Configurations, 4.6 Comparison Between Stocks, and 4.7 Summary.

4.2 Feature Analysis

In order to enrich the dataset and make the model more effective in identifying important patterns and trends, this paper also derived some new features to enhance the dataset which will help in finding the various important patterns and trends. To study short and long-term stock prices, the 50-day and 200-day MA were used, respectively. The rolling averages and exponential moving averages were used to do as well. K and D stochastic values were utilized as stochastic indicators to measure momentum variables comparing the close price of the stock to high and low stock prices based on a certain time period. The MACD also were calculated along with its signal line to see when a buy or sell signal is generated. Finally, RSI was calculated to understand the speed of price change and find if the price is overbought or oversold. Collectively, all these features provided a better understanding of the stock price movements and helped to better the forecast performance of the model.

The K and D stochastic indicators have been computed from the data and are presented in Figure 4.1. The stochastic values give information on the direction of the market, indicating when the market is overbought or oversold. This feature is very useful in pursuing price trends and producing buy/sell signals. Peaks and troughs represent over-bought (values close to 100) or over-sold (values close to zero) situations, which are reversal signals. The intersections of the K and D lines are used as buy/sell signals. For example, if the K line is seen to crossover the D line upwards, this depicts a bullish situation, while if it is seen to crossover the D line downwards, it depicts a bearish situation.

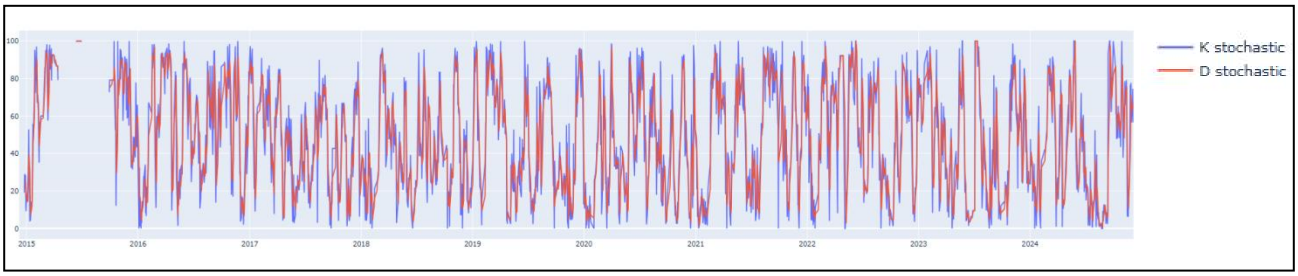

Figure 4.1: Stochastic Oscillator (%K and %D) Over Time

Figure 4.2 shows the MACD (purple line) and signal line (orange line) of the convergence and divergence of moving averages. These indicators are paramount in identifying momentum shifts with the crossovers as key buy or sell signals in the model's feature set. The MACD line differs between the 12-day and 26-day exponential moving averages. The signal line is a 9-day EMA of the MACD, which is used to identify buy/sell signals. Crossovers between the MACD and signal line indicate potential changes in momentum, where a MACD crossing above the signal line suggests a bullish trend, while the opposite implies a bearish trend.

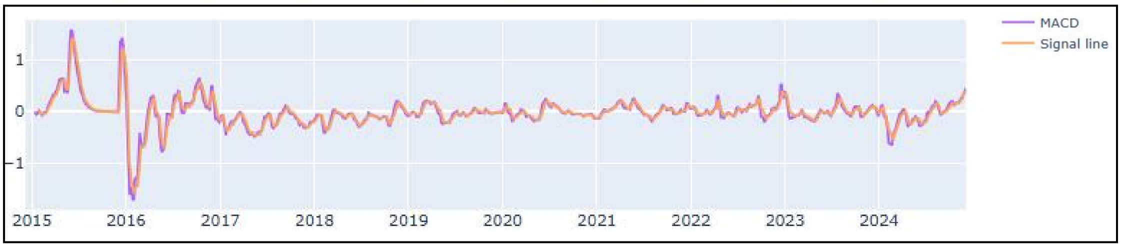

Figure 4.2 MACD and Signal Line Trends with Close Prices

Figure 4.3 and Figure 4.4 illustrate the 50-day (blue line) and 200-day (red line) EMA and MA overlaid on the stock price (grey line). The 50-day EMA captures short-term trends, while the 200-day EMA highlights long-term trends. Crossover points between the EMAs provide critical trading signals. A 50-day EMA crossing above the 200-day EMA suggests a bullish trend ("Golden Cross"), whereas the reverse indicates a bearish trend ("Death Cross").

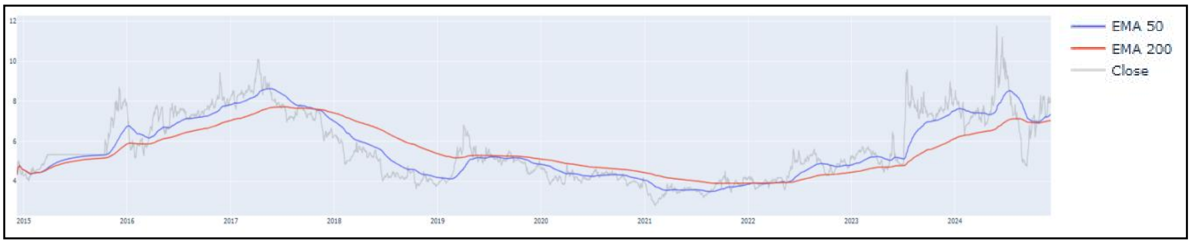

Figure 4.3: Exponential Moving Averages (50 and 200 days) with Close Prices

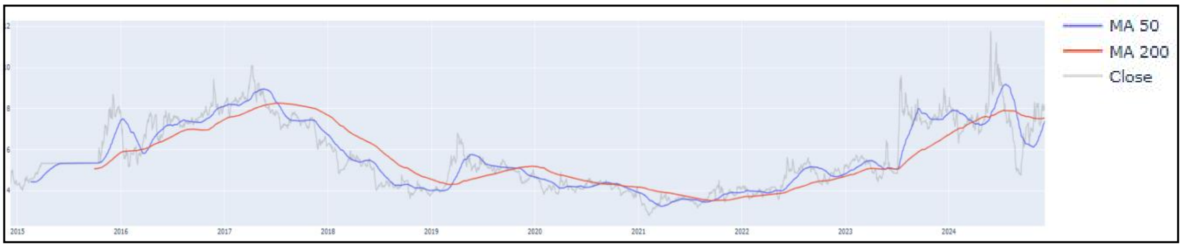

Figure 4.4: Moving Averages (50 and 200 days) with Close Prices

Relative Strength Index (RSI) calculated over the dataset is shown in Figure 4.5. RSI Values above 70 are considered an overbought indication of an upcoming price decline. When RSI values fall below 30, the price is considered oversold — which means it should increase soon. The RSI is important because it helps to identify overbought and oversold conditions and gives us critical momentum insights that help us predict price reversals.

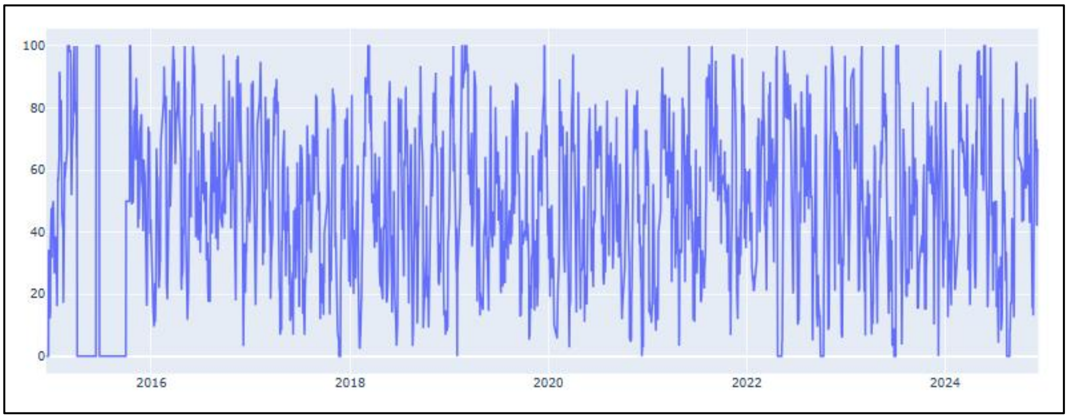

Figure 4.5: RSI Trends over Time

The addition of stochastic indicators, moving averages, MACD, and RSI adds more powerful trend and momentum characteristics to the input data for the hybrid GRU-LSTM model.

Figure 4.6 presents the final dataset, which includes all the features derived for this study, both technical and fundamental, to enable stock price prediction. This dataset includes daily stock market data for a given time frame and includes all the relevant attributes.

	Open	High	Low	Month	Day	Year	stochastic_k	stochastic_d	MACD	EMA_50	EMA_200	MA_50	MA_200	RSI 7	P/E Ratio	Num_Transactions	Profitability
Date																	
2020-09-24 00:00:00+08:00	4.319782	4.319782	4.239970	9	24	2020	2.777839	19.107743	0.000000	4.331234	4.309514	4.372457	4.313609	32.352941	1.246447	24096.50	0.000000
2020-09-25 00:00:00+08:00	4.249947	4.269900	4.190089	9	25	2020	2.439079	8.809673	0.000000	4.329610	4.309046	4.369265	4.312409	33.333333	1.691154	14835.50	0.000000
2020-09-28 00:00:00+08:00	4.220018	4.220018	4.020490	9	28	2020	5.172368	3.463096	0.000000	4.327022	4.308192	4.360685	4.310411	15.909074	1.281642	21603.35	0.000000
2020-09-29 00:00:00+08:00	4.090325	4.130230	4.050419	9	29	2020	10.344860	5.985436	0.000000	4.321500	4.306178	4.353701	4.308412	11.904849	1.294177	21528.51	0.007389
2020-09-30 00:00:00+08:00	4.120254	4.120254	4.060395	9	30	2020	17.073094	10.863441	0.000000	4.316687	4.304418	4.348115	4.305966	15.384597	1.598720	15585.00	0.002445
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2024-11-26 00:00:00+08:00	8.330000	8.330000	7.860000	11	26	2024	59.333356	54.444472	0.138956	7.245080	7.012347	6.996600	7.520200	69.444429	0.133991	578099.02	0.000000
2024-11-27 00:00:00+08:00	7.850000	8.600000	7.850000	11	27	2024	77.639722	71.213261	0.175071	7.257529	7.016674	7.064400	7.524400	83.435561	0.127927	634119.53	0.045685
2024-11-28 00:00:00+08:00	8.230000	8.520000	8.000000	11	28	2024	65.838510	67.603863	0.186215	7.276793	7.022774	7.129000	7.527300	72.941179	0.151666	520770.58	0.000000
2024-11-29 00:00:00+08:00	7.970000	8.050000	7.750000	11	29	2024	56.521738	66.666657	0.180858	7.291954	7.027897	7.192200	7.527500	66.111107	0.213831	359450.00	0.000000
2024-12-02 00:00:00+08:00	7.970000	8.080000	7.910000	12	2	2024	66.459642	62.939964	0.187363	7.303877	7.032246	7.258000	7.527900	69.230779	0.272781	285474.93	0.020253

Figure 4.6: Final Dataset After Feature Engineering

4.3 Evaluation of Baseline Models

4.3.1 Evaluation of ARIMA Model

Two key analyses were conducted to assess the stationarity of the time series data essential for implementing ARIMA models: The Augmented Dickey-Fuller (ADF) test and the rolling mean and standard deviation. Stationarity is a significant characteristic in time series analysis because non-stationary data leads to the production of inaccurate and misleading model predictions.

This paper uses the ADF test to determine whether a given time series has a unit root, meaning it tends to increase or decrease. If the p-value is greater than 0.05, then the series is said to have a unit root, and the mean and variance need to be transformed. In addition to the ADF test, rolling mean and standard deviation can be used to present a graphical approach to the stability of statistical properties over time. Figures 4.7 and 4.8 show the ADF test results for the stationarity of the stocks 600719 and 000679, respectively.

Looking at the results for stock 600719, the test statistic is -1.979 with a p-value of 0.2958, as presented in Figure 4.7. The p-value is also greater than the conventional levels of signification 5%, therefore, the null hypothesis of non-stationarity cannot be rejected. This suggests that the time series may be non-stationary in its current form and hence needs to be transformed, for instance, through differencing, before modeling is done.

Likewise, for stock 000679, the test statistic is -2.829 with a p-value of 0.0542, as shown in Figure 4.9. Although the test statistic is close to the critical values, the P-value also suggests that the null hypothesis cannot be rejected at a 5% level of significance. Hence, this series also needs to be transformed in order to achieve stationarity.

```
Results of Dickey Fuller Test
Test Statistics: -1.9790420461278726
p-value: 0.29583735259342214
Number of lags used: 27
Number of observations used: 2402
critical value (1%): -3.433075354750434
critical value (5%): -2.8627440256609367
critical value (10%): -2.5674109531399383
```

Figure 4.7: Results of the ADF Test for Stationarity on Stock 600719

```
Results of Dickey Fuller Test
Test Statistics: -2.8294466539143577
p-value: 0.054189073530332
Number of lags used: 25
Number of observations used: 2204
critical value (1%): -3.4333204775429715
critical value (5%): -2.862852263745625
critical value (10%): -2.5674685818969634
```

Figure 4.8: Results of the ADF Test for Stationarity on Stock 000679

Similarly, for stock 000679, the test statistic is -2.829 with a p-value of 0.0542, as presented in Figure 4.9. While the test statistic is closer to the critical values, the p-value still indicates that the null hypothesis cannot be confidently rejected at the 5% significance level. Therefore, this series also requires appropriate transformation to ensure stationarity.

These results underscore the usefulness of data transformation techniques like differencing or log transformation to normalize the mean and variance of the time series data for better modeling and forecasting.

The rolling mean and standard deviation of the log returns of the closing prices of stock 600719 and 000679 are depicted in Figures 4.9 and 4.10, respectively. These visualizations are employed to check the stationarity of the time series by looking at the behaviour of the rolling statistics over time.

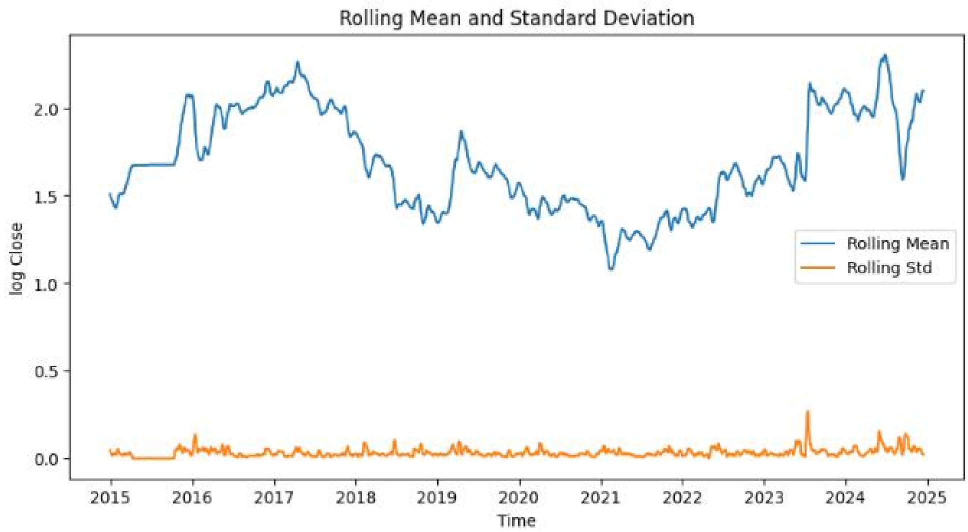

Figure 4.9: Rolling Mean and Standard Deviation of Log-Transformed Closing Prices on Stock 600719

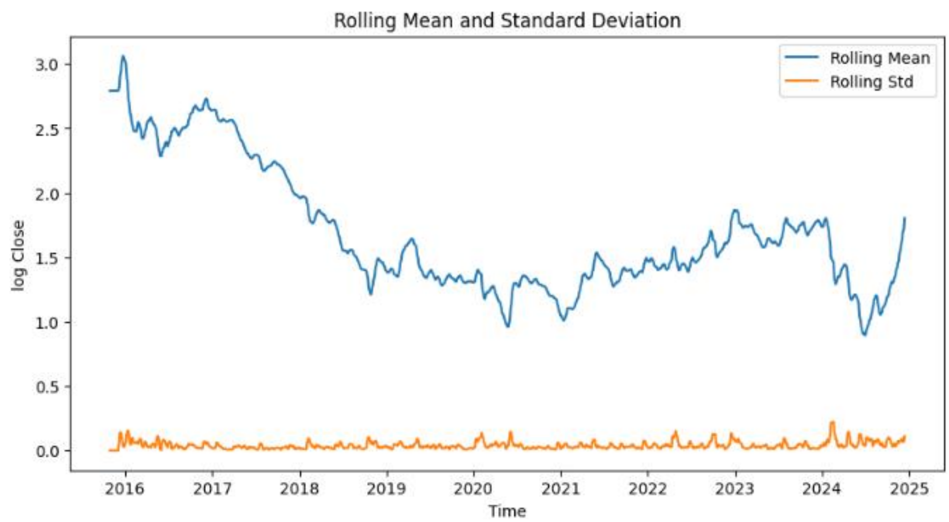

Figure 4.10: Rolling Mean and Standard Deviation of Log-Transformed Closing Prices on Stock 000679

In Figure 4.9, the rolling mean for stock 600719 oscillates over the time series, which means that the time series is non-stationary. To the same effect, the rolling standard deviation depicts variability, which is an indication that the data is non-stationary.

In Figure 4.10, for stock 000679, the same pattern is depicted; the rolling mean fluctuates over time. The rolling standard deviation also stays fairly constant in comparison, but it is not constant and does capture changes, which means that the series is not stationary in its current form.

Both series are non-stationary, and therefore, differencing or detrending is needed to make them stationary. ARIMA models are only applicable when the data is stationary; therefore, making data stationary is a crucial step in the modeling process. These rolling statistics give a graphical confirmation of the importance of data cleaning before analysis.

The SARIMAX model summaries for the stocks 000679 and 600719 are presented in Figures 4.11 and 4.12, respectively. These summaries extend their focus on the results of the SARIMAX models, the estimates of parameters, the tests used, and the fit metrics.

In Figure 4.11 (Stock 000679), the SARIMAX model was estimated with the parameters that describe the order of the lag, the integration, and the moving average. The model has been checked for the goodness of fit using AIC (-7402.455), BIC (-7391.532), and HQIC (-7398.416). The Ljung-Box Q-test checks the residual's autocorrelation, skewness, and kurtosis and gives information about the residual distribution. The coefficients of the model are also statistically tested, and the result can be seen through the p-value of each coefficient.

```
                               SARIMAX Results
==============================================================================
Dep. Variable:                      y   No. Observations:                 1741
Model:               SARIMAX(0, 1, 0)   Log Likelihood                3703.047
Date:                Fri, 20 Dec 2024   AIC                          -7404.094
Time:                        08:29:15   BIC                          -7398.633
Sample:                             0   HQIC                         -7402.075
                               - 1741
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
sigma2         0.0008   1.75e-05     47.310      0.000       0.001       0.001
===================================================================================
Ljung-Box (L1) (Q):                   0.34   Jarque-Bera (JB):               716.45
Prob(Q):                              0.56   Prob(JB):                         0.00
Heteroskedasticity (H):               0.94   Skew:                            -0.02
Prob(H) (two-sided):                  0.49   Kurtosis:                         6.14
===================================================================================
```

Figure 4.11: SARIMAX Model Summary for Stock 000679

In Figure 4.12 (Stock 600719), the SARIMAX model has a similar structure with adjusted parameters for this stock in particular. AIC (-8953.199), BIC (-8941.735), and HQIC (-8950.486) Values indicate that the model has a good fit. Statistical tests such as the Jarque-Bera test (JB: 350.57) and heteroskedasticity metrics also support the analysis of the residuals' properties. The parameter estimates, for example, ar.L1 and sigma2, have their p-values, which makes them important in the model.

```
                               SARIMAX Results
==============================================================================
Dep. Variable:                      y   No. Observations:                 1941
Model:               SARIMAX(2, 1, 4)   Log Likelihood                4476.195
Date:                Fri, 20 Dec 2024   AIC                          -8938.390
Time:                        08:31:21   BIC                          -8899.397
Sample:                             0   HQIC                         -8924.050
                               - 1941
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.4444      0.134      3.323      0.001       0.182       0.707
ar.L2         -0.5093      0.132     -3.866      0.000      -0.767      -0.251
ma.L1         -0.4785      0.136     -3.522      0.000      -0.745      -0.212
ma.L2          0.5774      0.136      4.248      0.000       0.311       0.844
ma.L3         -0.0207      0.027     -0.762      0.446      -0.074       0.032
ma.L4         -0.0692      0.024     -2.940      0.003      -0.115      -0.023
sigma2         0.0006   1.12e-05     51.714      0.000       0.001       0.001
===================================================================================
Ljung-Box (L1) (Q):                   0.01   Jarque-Bera (JB):              1222.93
Prob(Q):                              0.94   Prob(JB):                         0.00
Heteroskedasticity (H):               0.90   Skew:                            -0.24
Prob(H) (two-sided):                  0.18   Kurtosis:                         6.86
===================================================================================
```

Figure 4.12: SARIMAX Model Summary for Stock 600719

These model summaries are critical for evaluating the suitability of SARIMAX in capturing the underlying patterns of each stock's time series data. The comparative analysis of AIC and BIC values between the two models also aids in understanding the relative model performance for the two stocks.

The ARIMA model was applied to the stock data to evaluate its predictive capability using key performance metrics: MSE, RMSE, MAE, and MAPE. The evaluation results for both stocks, 000679 and 600719, are shown in Table 4.1.

Table 4.1: Performance Metrics of the ARIMA Model

Stock	**Metrics**		
	MSE	**RMSE**	**MAE**
000679.SZ	2.055	1.434	1.059
600719.SS	5.408	2.326	1.912

The relatively high values of RMSE and MSE reflect the model's difficulty in accurately predicting stock price fluctuations. MAE further confirms that the average error remains substantial, indicating ARIMA's shortcomings in capturing non-linear dependencies inherent in financial time-series data.

To better understand the model's performance, the predicted values were plotted against the actual test data as shown in Figure 4.13 and 4.14.

Figure 4.13: ARIMA Predictions for 600719

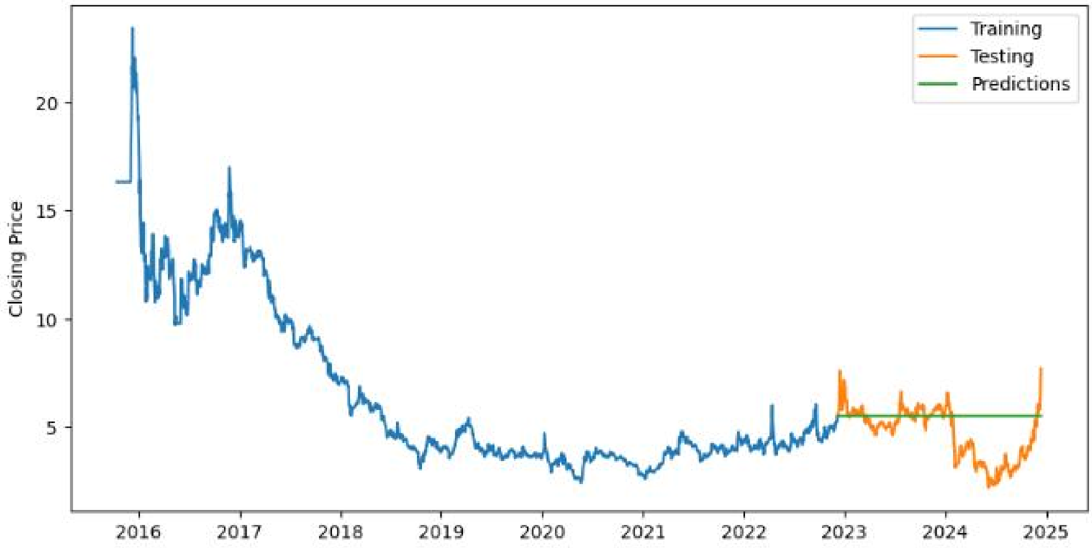

Figure 4.14: ARIMA Predictions for 000679

In Figure 4.13, ARIMA predictions fail to capture sudden market spikes or drops, with a notable lag in trend adjustment. Training and test data are clearly segregated, emphasizing the challenge in generalizing trends. In Figure 4.14, similar issues arise, with predictions showing over-smoothing and an inability to react to rapid market changes. These findings highlight ARIMA's reliance on stationarity assumptions, which limit its applicability in highly volatile stock markets.

4.3.2 Performance of Standalone LSTM and GRU Models

The standalone LSTM model was evaluated for both stocks, as shown in Figures 4.15 and 4.17. The metrics indicate how effectively LSTM layers capture sequential dependencies within the data. The results also show that increasing the number of LSTM layers generally enhances the model's ability to capture long-term dependencies but may also lead to overfitting if the layers are excessively deep. The visualizations demonstrate that while the predicted values closely follow actual trends, certain deviations occur during sharp price changes.

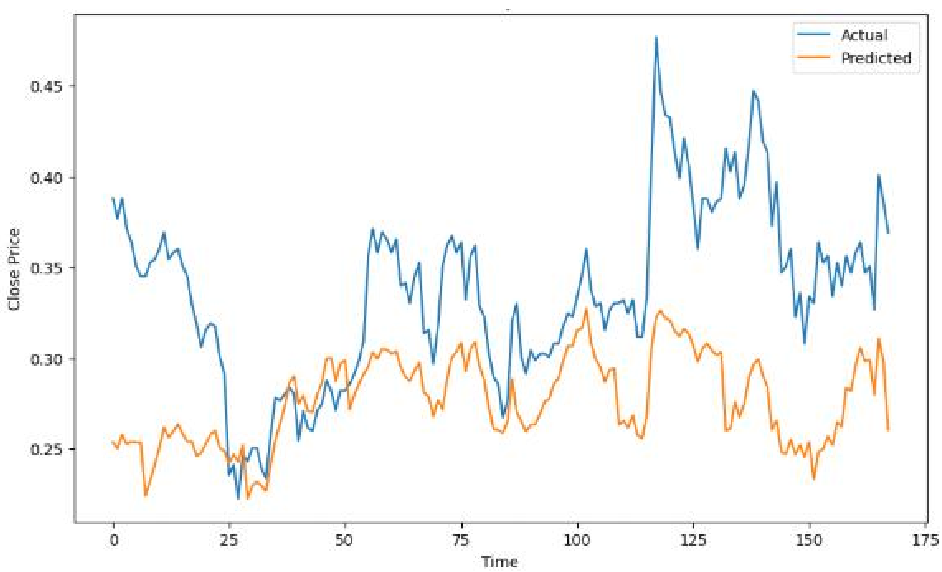

Figure 4.15: Actual vs Predicted Stock Prices for LSTM Model on Stock 000679

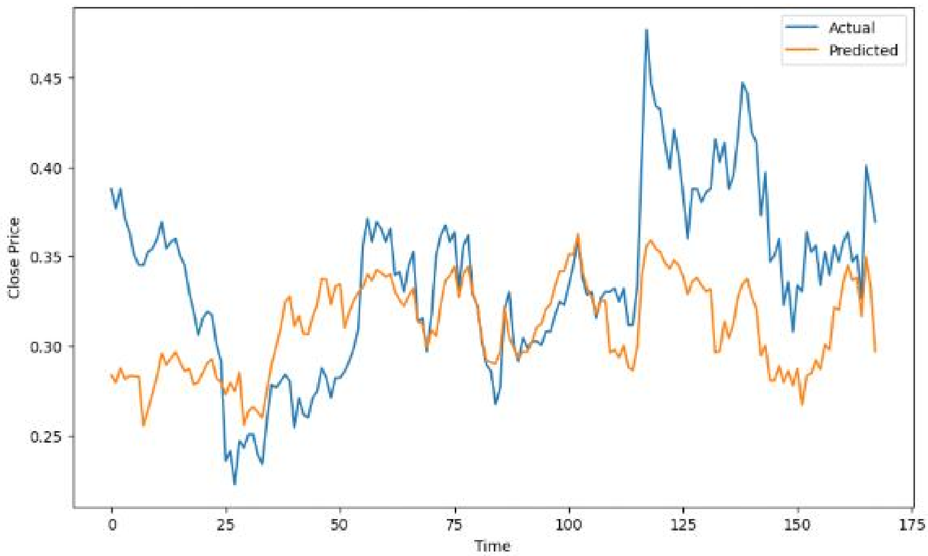

Figure 4.16: Actual vs Predicted Stock Prices for GRU Model on Stock 000679

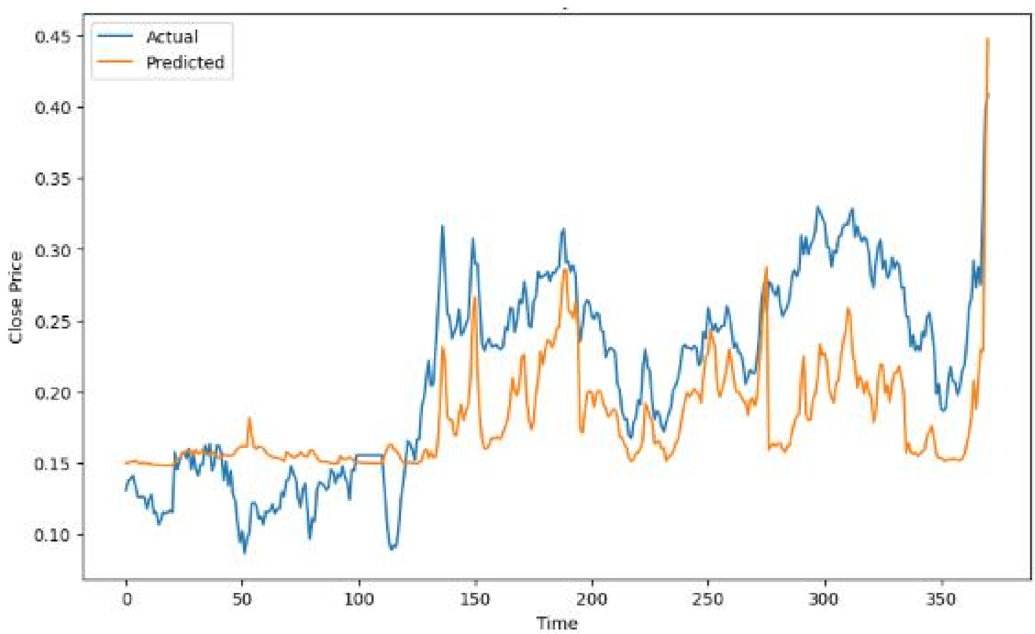

Figure 4.17: Actual vs Predicted Stock Prices for LSTM Model on Stock 600719

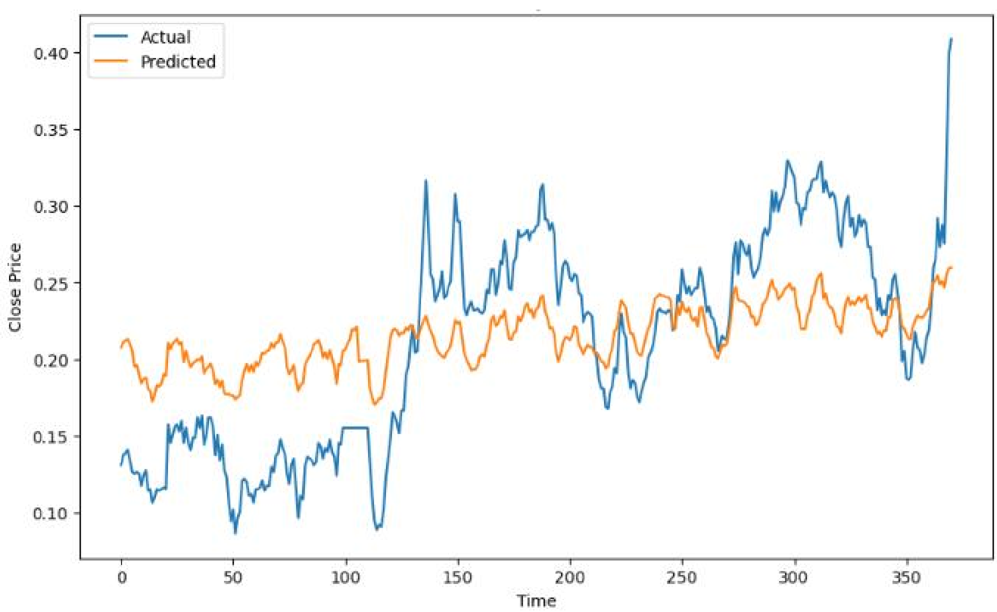

Figure 4.18: Actual vs Predicted Stock Prices for GRU Model on Stock 600719

On the other hand, the GRU standalone model demonstrates better generalization with slightly lower errors than LSTM layers as seen in Table 4.2. However, for stock 600719, the performance marginally deteriorates, suggesting that GRU layers might not always capture the full complexity of fundamental and technical data.

Table 4.2: Performance Results of GRU and LSTM Standalone Models

Evaluation		**Metrics**			
Stock	**Model**	**MSE**	**RMSE**	**MAE**	**MAPE**
600719	GRU	0.0029	**0.0542**	0.0463	0.2835
	LSTM	0.0032	0.0566	0.0474	0.2221
000679	GRU	0.0025	**0.0505**	0.0406	0.1164
	LSTM	0.0030	0.0545	0.0384	0.0842

Comparing the LSTM and GRU models, GRU shows superior performance for both stocks, with lower MSE, RMSE, and MAE values. However, the difference in MAPE suggests that LSTM may better handle percentage-based deviations in certain scenarios. The visualized predictions align well with actual values for both architectures, though GRU predictions appear more consistent during volatile periods.

The analysis highlights that both GRU and LSTM have strengths in modeling temporal dependencies, but GRU offers a simpler and computationally efficient alternative with comparable performance. The choice between GRU and LSTM ultimately depends on the specific stock and the desired trade-off between complexity and predictive accuracy. Future work may explore hybrid configurations combining both architectures to maximize their complementary strengths.

4.4 Performance of Hybrid GRU-LSTM Model

This section analyzes the performance of the proposed GRU-LSTM hybrid system for stock price prediction of incorporating fundamental data such as EPS, profitability and P/E ratio, with the description of technical analysis indicators. Cross-comparisons of prediction results for two stock data sets, 600719 and 000679, are made. The MSE, RMSE, MAE and MAPE are calculated to evaluate the accuracy of the prediction.

Figures 4.19 to 4.22 show the graphical analysis of the actual and predicted stock prices of models with and without fundamental data. In both cases, the results of models that use both fundamental and technical values are closer to the actual stock prices than models that use only technical data. These graphs suggest that the two data types are useful when integrated, because the hybrid approach takes fine details into account versus the other approach.

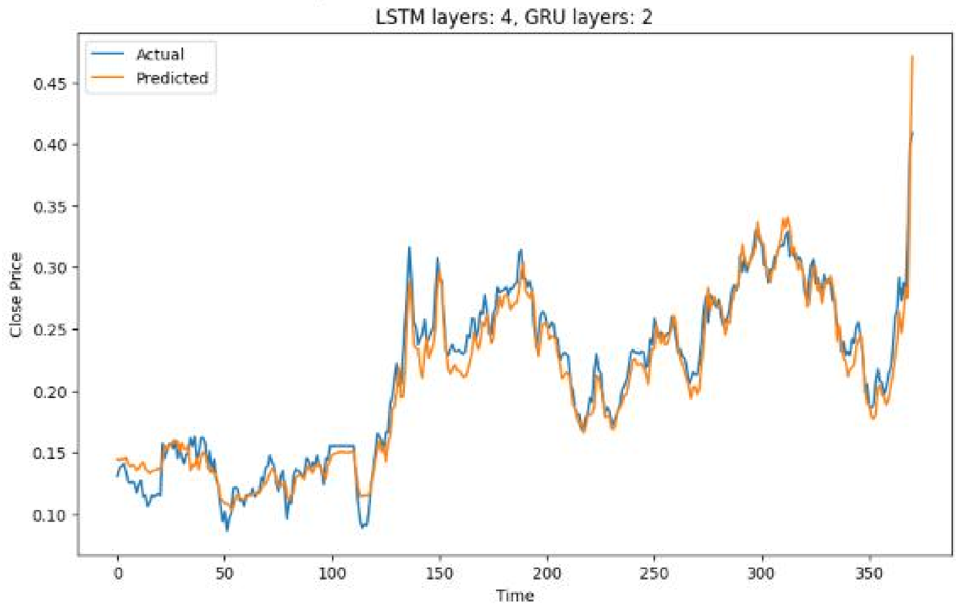

Figure 4.19: Predicted vs Actual Stock Prices for 600719 with Fundamental and Technical Data Integration

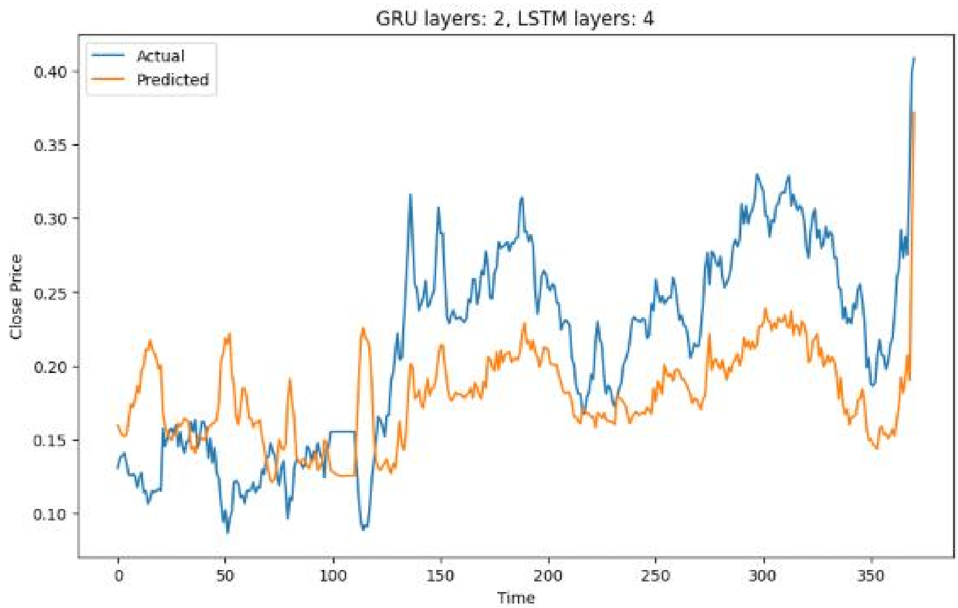

Figure 4.20: Predicted vs Actual Stock Prices for 600719 with Technical Data Only

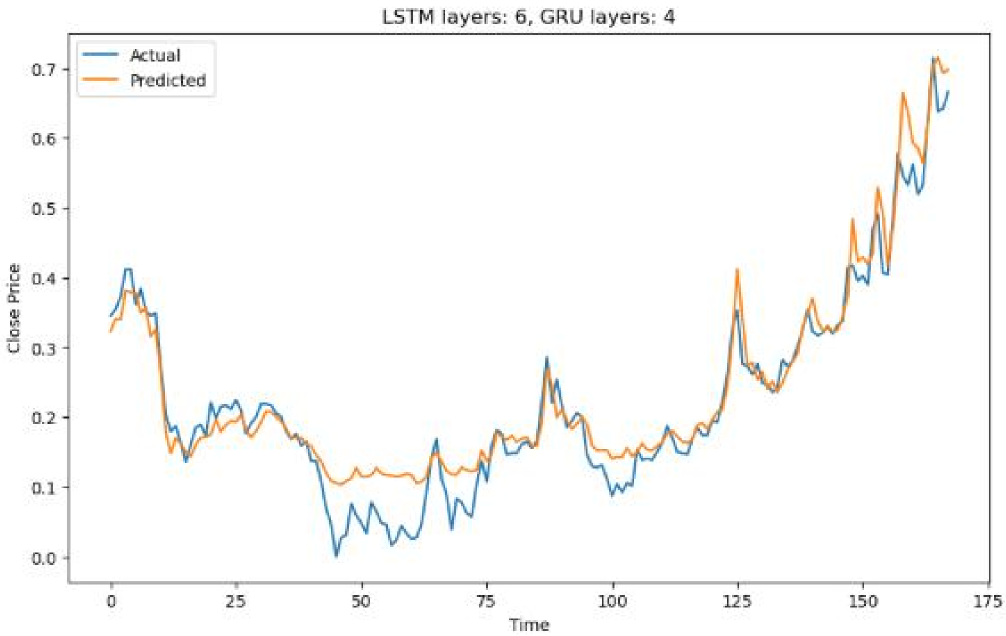

Figure 4.21: Predicted vs Actual Stock Prices for 000679 with Fundamental and Technical Data Integration

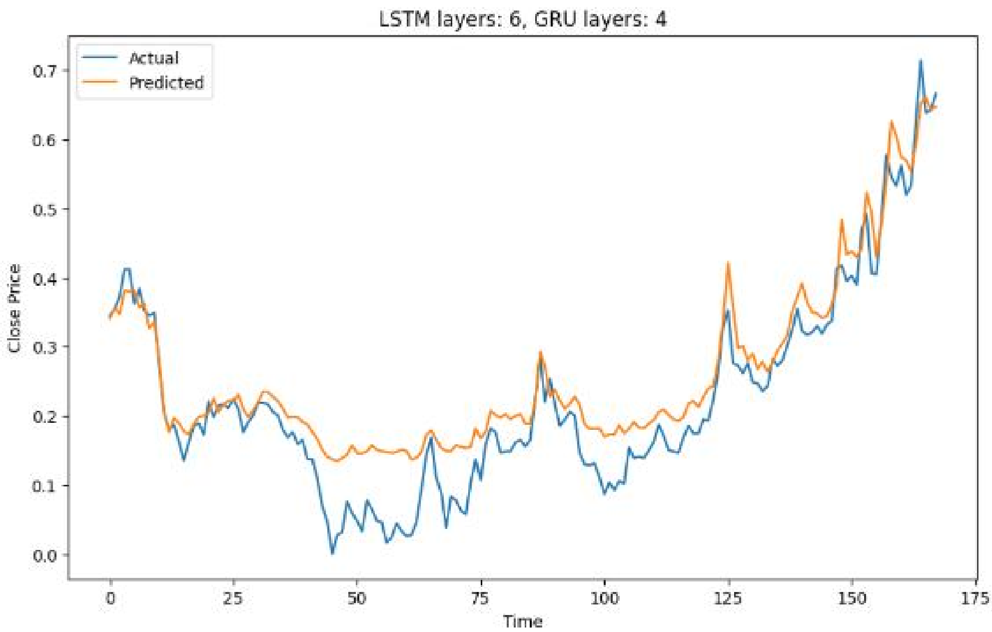

Figure 4.22: Predicted vs Actual Stock Prices for 000679 with Technical Data Only

For 600719, the hybrid model with fundamental and technical data achieved superior performance, as evidenced by an MSE of **0.0002**, RMSE of **0.0141**, MAE of 0.0111, and MAPE of 0.0569. These values indicate minimal deviation between the predicted and actual stock prices. In comparison, the model using only technical data exhibited significantly higher error values, with an MSE of 0.0036 and MAPE of 0.2659, demonstrating the importance of incorporating fundamental data in improving prediction accuracy.

Similarly, for 000679, the inclusion of fundamental data resulted in improved performance, with an MSE of **0.0013**, RMSE of **0.0360**, MAE of **0.0269**, and MAPE of **0.0587**. Models trained solely on technical data displayed slightly higher error metrics, with an MSE of **0.0022** and MAPE of **0.0610**, further validating the contribution of fundamental data in enhancing model efficacy.

Table 4.3: Evaluation Metrics for Models with Fundamental and Technical Data Integration

Evaluation		**Metrics**			
Stock	Data	MSE	RMSE	MAE	MAPE
600719	Fundamental & Technical	**0.0002**	**0.0141**	**0.0111**	**0.0569**
	Technical Only	0.0036	0.0601	0.0540	0.2659
000679	Fundamental & Technical	**0.0013**	**0.0360**	**0.0269**	**0.0587**
	Technical Only	0.0022	0.0474	0.0348	0.0610

The findings show that incorporating basic data is useful in enhancing the accuracy of stock price forecasts. The fundamental data help in understanding the value and condition of a company as a business, in addition to the short-term trends that the technical data gives. The improved performance metrics for both stocks prove that the GRU-LSTM hybrid model has the advantage of using multiple features that help in capturing the movement of the stock prices.

However, the extent of the improvement is different for each dataset, which means that the effect of the fundamental data may be different from one stock to another. For example, firms with stable returns and well-defined financial ratios might get more value from using fundamental data, whereas more risky shares may need more auxiliary variables for further enhancement.

4.5 Effect of GRU and LSTM Layer Configurations

In this section, the author discusses the different structures of the GRU and LSTM layers and how they affected the prediction of the two stocks, 000679 and 600719. To find the best structure for the proposed hybrid model type proposed depending on both GRU and LSTM layers, the previously mentioned configurations were executed according to the number of GRU and LSTM layers defined. To evaluate the model, this study will make use of MSE, RMSE and MAE metrics. The summary of results of the experiments with stock 000679 and 600719 is shown in Tables 4.4 and 4.5 respectively.

Table 4.4: Performance Metrics for Various GRU-LSTM Layer Configurations on Stock 000679

Configuration (GRU-LSTM)	**MSE**	**RMSE**	**MAE**
7 GRU, 9 LSTM	0.0118	0.1086	0.1834
6 GRU, 8 LSTM	0.0016	0.0403	0.0285
5 GRU, 7 LSTM	0.0015	0.0385	0.0301
4 GRU, 6 LSTM	**0.0013**	**0.0360**	**0.0269**
3 GRU, 5 LSTM	0.0026	0.0513	0.0406
2 GRU, 4 LSTM	0.0036	0.0598	0.0466
1 GRU, 3 LSTM	0.0021	0.0455	0.0368

As shown in Table 4.4, the performance metrics of stock 000679 are quite different based on the configuration of the GRU-LSTM layer. The best results were achieved with 4 GRU and 6 LSTM layers: MSE = 0.0013, RMSE = 0.0360, and MAE=0.0269; therefore, such architecture is the most precise and least erroneous. Conversely, the configuration with 7 GRU layers and 9 LSTM layers recorded the highest errors across all metrics (RMSE: 0.1086, MAE: 0.1834 at 0118), which indicate that including more layers can result in overfitting.

Configurations with fewer GRU and LSTM layers, such as 1 GRU and 3 LSTM, also performed well (RMSE: 0.0455, MAE: 0.0368 for 0021), albeit with slightly higher errors compared to the optimal configuration. This means that although the increase in the complexity of the model enhances the possibility of identifying more complex patterns, the over-complication of the model reduces its ability to generalize.

As can be seen in Table 4.5, the best model for stock 600719 was achieved with 2 GRU layers and 4 LSTM layers, and the model has the lowest MSE of 0.0002, RMSE of 0.0141, and MAE of 0.0111. This result shows that the model can reach high accuracy with the right number of layers in the network. On the other hand, the configuration with 7 GRU layers and 9 LSTM layers resulted in relatively higher errors (MSE: 0.0004, RMSE: 0.0201, MAE: 0.0170), which show that with the increase in the number of layers, the performance starts to decline.

The findings point out the trade-off between the model's complexity and the accuracy of the predictions. The configurations with the middle layers, such as 4 GRU and 6 LSTM or 3 GRU and 5 LSTM, were also effective, indicating a stable performance range within this configuration range. As with stock 000679, basic configurations like 1 GRU and 3 LSTM also worked fine but with slightly higher errors.

Table 4.5: Performance Metrics for Various GRU-LSTM Layer Configurations on Stock 600719

Configuration (GRU-LSTM)	**MSE**	**RMSE**	**MAE**
7 GRU, 9 LSTM	0.0004	0.0201	0.0170
6 GRU, 8 LSTM	0.0002	0.0142	0.0109
5 GRU, 7 LSTM	0.0005	0.0220	0.0182
4 GRU, 6 LSTM	0.0005	0.0220	0.0186
3 GRU, 5 LSTM	0.0004	0.0205	0.0170
2 GRU, 4 LSTM	**0.0002**	**0.0141**	**0.0111**
1 GRU, 3 LSTM	0.0008	0.0279	0.0234

The results indicate that the selection of the layer configuration for the hybrid GRU-LSTM models is important. While using many layers makes the model learn the data well, more than needed layers will cause overfitting and high computational costs. The two stocks, 000679 and 600719, were chosen to have the best performance with the moderate layers to prove that there is a trade off between complexity and accuracy.

The results also reveal that the best architecture is not the same for the two stocks, suggesting that the choice of layer configuration may depend on the characteristics of the stock data. It is clear that for any type of hybrid model the selection of the model and the tuning of the model has to be done based on the data set at hand.

4.6 Comparison Between Stocks

In this section, we compare the predictive performance for the two stocks, 600719 and 000679, using various models based on the metrics MSE, RMSE, MAE, and MAPE. The evaluation measures of ARIMA, standalone GRU, standalone LSTM, and hybrid GRU-LSTM are presented in Table 4.6 for different configurations.

Table 4.6: Comparative Performance Metrics for Different Models and Stocks

Evaluation		**Metrics**			
Stock	**Model**	**MSE**	**RMSE**	**MAE**	**MAPE**
600719	ARIMA	5.408	2.326	1.912	**-**
	Standalone GRU	0.0029	0.0542	0.0463	0.2835
	Standalone LSTM	0.0032	0.0566	0.0474	0.2221
	GRU-LSTM with Fundamental & Technical Data	**0.0002**	**0.0141**	**0.0111**	**0.0569**
	GRU-LSTM with Technical Data	0.0036	0.0601	0.0540	0.2659
000679	ARIMA	2.055	1.434	1.059	**-**
	Standalone GRU	0.0025	0.0505	0.0406	0.1164
	Standalone LSTM	0.0030	0.0545	0.0384	0.0842
	GRU-LSTM with Fundamental & Technical Data	**0.0013**	**0.0360**	**0.0269**	**0.0587**
	GRU-LSTM with Technical Data	0.0022	0.0474	0.0348	0.0610

4.6.1 Performance Analysis for Stock 600719

For stock 600719, the hybrid GRU-LSTM model integrating both fundamental and technical data exhibited the best performance across all metrics (MSE: 0.0002, RMSE: 0.0141, MAE: 0.0111, MAPE: 0.0569. This finding supports the use of basic information in conjunction with technical information in the analysis of stock price trends.

The performance of the standalone GRU and LSTM models was also comparable to the proposed ensemble models, with the standalone GRU model having slightly lower MSE and RMSE than the LSTM model. However, both models were less accurate than the hybrid GRU-LSTM model with fundamental and technical features. Also, the errors of the hybrid GRU-LSTM model using only technical data were higher than the model incorporating fundamental data, which underlines the significance of including fundamental variables.

The ARIMA model was found to be the least efficient in all aspects, with high errors suggesting it is not fit for the non-linear and irregular nature of stock prices.

4.6.2 Performance Analysis for Stock 000679

For stock 000679, the hybrid GRU-LSTM model with fundamental and technical data again emerged as the most accurate, with the lowest error metrics (MSE: 0.0360, MAE of 0.0269, and MAPE of 0.0587). In the same manner as stock 600719, the fundamental data improved the performance of the hybrid model.

The standalone GRU and LSTM models had similar accuracy to the GRU model, showing slightly better results in terms of MSE and RMSE. However, neither standalone model was as good as the proposed hybrid GRU-LSTM model. With only the technical data, the error metrics of the hybrid GRU-LSTM model were higher, which is another proof of the effectiveness of incorporating fundamental data.

4.6.3 Discussion

The comparative analysis of the results proves the effectiveness of the proposed hybrid GRU-LSTM model based on fundamental and technical indicators for both stocks. It enables the model to have results from the two models and thus come up with a better model. The results also indicate that the standalone deep learning models as efficient as they are, cannot overcome the proposed hybrid model.

The results also reveal the limitations of linear models, for example, ARIMA, which fail to capture non-linear features of the data. This strengthens the argument about the use of deep learning approaches, especially the hybrid model for stock price prediction. This paper employs the proposed hybrid GRU-LSTM model that combines fundamental and technical data to forecast stock prices. These findings provide evidence for the generalisation of the model and a good foundation for the expansion of the model to financial forecasting.

4.7 Summary

In this chapter, an analysis of the proposed hybrid GRU-LSTM model for stock price prediction is discussed in detail. In this case, the study aims at improving the prediction of the model by including technical indicators such as moving averages (50-day and 200-day), MACD, RSI, stochastic factors, and fundamental indicators such as EPS and P/E. These features were extracted in a very rigorous feature engineering manner to extract trends, momentum, and trading signals from the given dataset. The present study focused on stock 600719 and stock 000679 for a decade, and the performance was assessed based on MSE, RMSE, MAE, and MAPE.

ARIMA, standalone GRU, and standalone LSTM were employed to compare with the proposed models and set the baseline. ARIMA cannot model the non-linear and extremely unstable stock price data and has high forecasting errors, particularly for linear trends. On the other hand, standalone GRU and LSTM models were observed to be better, with GRU outperforming LSTM because it is less complex and has the capacity to handle the sequential dependencies of the time-series data.

In the experiment, the proposed hybrid GRU-LSTM model was superior to the basic models. For stock 600719, the model that incorporates both fundamental and technical data was better than the model that incorporates only technical data with an MSE of 0.0002 against an MSE of 0.0036. Similarly, for stock 000679, the MSE of the hybrid model was 0.0013 when integrated data was used and 0.0022 when only technical indicators were used. This is in agreement with the previous finding that the use of fundamental and technical data improves the accuracy of stock price prediction.

The study also had the objective of comparing the impact of changing the number of layers in both GRU and LSTM to the performance of the model. For stock 000679, the model decomposition of 4 GRU and 6 LSTM layers was the best one with MSE of 0.0013. Similarly, for stock 600719, the best architecture was identified to be 2 GRU and 4 LSTM layers with MSE of 0.0002. Overfitting was observed when the depth of the configurations was set to very high, while a relatively low depth provided good but less precise estimates.

When the two stocks were compared, it was observed that the proposed hybrid GRU-LSTM model outperformed the other three models, which included GRU, LSTM, and ARIMA. The inclusion of the basic variables alongside the technical data improved the capacity of the model to explain the stock price relationship and trends. The study shows that the proposed hybrid GRU-LSTM model is more effective than the other models and can be used to make real-life financial predictions. Therefore, this study reveals that feature engineering and model configuration are crucial for getting the most efficient results in a very unpredictable financial market.

CHAPTER 5

CONCLUSION

5.1 Summary of Findings and Contributions

This study proposed and implemented a hybrid GRU-LSTM deep learning model for predicting stock prices by combining both technical indicators and fundamental data to enhance the predictive accuracy. The results were examined using key performance metrics, including MSE, RMSE, MAE, and MAPE, across two stock datasets: 600719 and 000679.

The GRU-LSTM model with integrated fundamental and technical data achieved superior performance compared with conventional methods. The hybrid model improved by combining the strengths of GRU which is computationally efficient and LSTM which captures long-term dependencies to achieve better generalization capabilities and higher predictive accuracy. More importantly, the addition of fundamental indicators improved the robustness of the model as opposed to using only technical indicators.

Table 5.1 provides an overview of how this systematic approach implemented in this study based on the overall research objectives, deliverables and the corresponding chapters that support achieving them. This alignment ensured a structured methodology, guiding the research process and validating the findings presented in this chapter.

Table 5.1: Alignment of Research Objectives with Deliverables and Thesis Chapters

Research Objective	**Deliverables**	**Chapter Output**
To identify the shortcomings traditional ARIMA and standalone GRU and LSTM stock price prediction model.	- Literature review to analyze the limitations of ARIMA and standalone deep learning models.	2
	- Implementation of ARIMA, GRU, and LSTM models. - Comparative analysis of performance using evaluation metrics.	4
To develop a hybrid model that integrates GRU and LSTM architectures with both fundamental and technical data, leveraging their complementary strengths.	- A detailed methodology describing the GRU-LSTM hybrid model, including integration of fundamental and technical data.	3
	- Comparative analysis of performance using evaluation metrics.	4
To investigate the impact of varying the number of layers in GRU and LSTM architectures on the performance and accuracy of	Experimental results and discussions showing the effect of different GRU-LSTM layer configurations.	4
To evaluate the performance of the enhanced predictive model with evaluation metrics such as MSE, RMSE, MAE and MAPE.	A comprehensive comparison of evaluation metrics across baseline and hybrid models.	4

5.2 Limitations of the Study

Although results of the proposed model are promising, there are some limitations that should be addressed.First, the model heavily relies on the volume and quality of financial data available. So the accuracy of them is totally depending on the variety and reliability of technical and fundamental indicators which is different for each market, company and region respectively. In this study, the selected stocks have very limited fundamental data available, hence using proxy calculations.

Next, the analysis indicates that GRU-LSTM model is complex in terms of computing. The hybrid architecture out-performed baselines in terms of predictive power, however, deep learning models entail enormous computational demands. This may restrict the scalability and practical use of the model for real-time prediction or for resource-constrained applications.

For the sake of feature scope, the research focused mainly on technical and fundamental indicators. Yet, additional external elements (economic news, world developments, market psychology, and global financial crises) were not accounted for. By including stock price movements and external factors that seem to impact stock price movements, the predictive power of the model can be enhanced.

Lastly, the study's evaluation scope is limited to a mere two stocks, 600719 and 000679, which might not reflect overall market conditions. By testing the model on a small dataset, the findings become less generalizable to other stocks, sectors or world financial markets. Wider evaluation across a range of stocks would give a greater feel for how the model performs.

Addressing these limitations in future work will further enhance the reliability, efficiency, and applicability of the hybrid GRU-LSTM model for stock price prediction.

5.3 Future Works

Taking into consideration the limitations underlined in this study, the following are potential avenues for future work to augment the proposed GRU-LSTM hybrid model:

Firstly, enhancing the model with sentiment analysis can make a big difference to the performance of the model. Adding market sentiment features, for instance, obtained from online news articles, company earnings reports, or social media posts, can supplement the technical and fundamental features already provided. This helps the model to better predict price movements based on external perceptions by using sentiment analysis to gain insights into investor behavior and market reactions.

Secondly, the inclusion of actual fundamental data such as EPS, P/E ratio, D/E ratio and P/B ratio is to be further experimented. The selection of popular stocks in the US market could have solved the problem of data unavailability. Since data for Chinese stocks is very limited on the Yahoo Finance API.

Thirdly, the model generalizability would benefit from testing on larger datasets. Gradually increasing the diversity of the data in terms of both stocks and stock characteristics will show whether or not the hybrid GRU-LSTM model achieves consistent performance across sampling space in a linear manner and will conclusively show its generalization ability.

Furthermore, more advanced techniques for optimization can also be applied to optimize the hyperparameters of the models. Using methods like Bayesian optimization or grid search may identify the best configurations for the GRU and LSTM layers, allowing faster training and therefore, prediction accuracy.

Finally, a real-time prediction framework needs to be devised for real-life applications. Such a real-time system would also help investors, traders, and analysts with dynamic data processing and accurate predictions of stock price fluctuations.

By addressing these future directions, the study can further improve the hybrid model's robustness, scalability, and real-world applicability, solidifying its contribution to stock price prediction research.

REFERENCES

Aggarwal, T., Sharma, N., & Aggarwal, N. (2022). Gunshot Detection and Classification Using a Convolution-GRU Based Approach. In *Advances in intelligent systems and computing* (pp. 95–107). https://doi.org/10.1007/978-981-19-4182-5_8

Al-Kahtani, M. S., Mehmood, Z., Sadad, T., Zada, I., Ali, G., & ElAffendi, M. (2023). Intrusion Detection in the Internet of Things Using Fusion of GRU-LSTM Deep Learning Model. *Intelligent Automation & Soft Computing*, *37*(2), 2279–2290. https://doi.org/10.32604/iasc.2023.037673

Bathla, G. (2020). Stock Price prediction using LSTM and SVR. *N 2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC) (Pp. 211-214). IEEE.*, 211–214. https://doi.org/10.1109/pdgc50313.2020.9315800

Bibi, I., Akhunzada, A., Malik, J., Iqbal, J., Musaddiq, A., & Kim, S. (2020). A Dynamic DL-Driven Architecture to Combat Sophisticated Android Malware. *IEEE Access*, *8*, 129600–129612. https://doi.org/10.1109/access.2020.3009819

Bisarya, U., Parekh, V., & Bhattacharjee, S. (2022). Stock Price Prediction Using Corporation Network and LSTM. *2022 2nd International Conference on Intelligent Technologies (CONIT)*, 1–6. https://doi.org/10.1109/conit55038.2022.9848403

Blockchain and Trustworthy Systems. (2024). [Conference-proceeding; eBook]. In J. Chen, B. Wen, & T. Chen (Eds.), *5th International Conference, BlockSys 2023* (Vol. 1897). https://doi.org/10.1007/978-981-99-8104-5

Bohn, T. A. (2017). Improving Long Term Stock Market Prediction with Text Analysis. *(Master's Thesis, the University of Western Ontario (Canada)).* https://ir.lib.uwo.ca/cgi/viewcontent.cgi?article=6267&context=etd

Cai, C., Ma, Q., & Lv, S. (2012). Research on Support Vector Regression in the Stock Market Forecasting. In *Advances in intelligent and soft computing* (pp. 607–612). https://doi.org/10.1007/978-3-642-28655-1_96

Cao, M., Li, V. O. K., & Chan, V. W. S. (2020). A CNN-LSTM Model for Traffic Speed Prediction. *In 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring) (Pp. 1-5). IEEE.* https://doi.org/10.1109/vtc2020-spring48590.2020.9129440

Chen, D., Yan, X., Liu, X., Li, S., Wang, L., & Tian, X. (2020). A Multiscale-Grid-Based Stacked Bidirectional GRU Neural Network Model for Predicting Traffic Speeds of Urban Expressways. *IEEE Access*, *9*, 1321–1337. https://doi.org/10.1109/access.2020.3034551

Donahue, J., Hendricks, L. A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Darrell, T., & Saenko, K. (2015). Long-term recurrent convolutional networks for visual recognition and description. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2625-2634.*, 2625–2634. https://doi.org/10.1109/cvpr.2015.7298878

Fischer, T., & Krauss, C. (2017). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, *270*(2), 654–669. https://doi.org/10.1016/j.ejor.2017.11.054

Goel, H., & Shyamala, L. (2024). Enhancing Investment Decision-Making: A Comprehensive Approach through Fundamental, Technical and Sentiment Analysis with Machine Learning and Deep Learning Algorithms. *In SCOPE: Vol. 2024 10th International Conference on Communication and Signal Processing (ICCSP) (Pp. 1655–1655).* https://doi.org/10.1109/iccsp60870.2024.10544298

Greff, K., Srivastava, R. K., Koutnik, J., Steunebrink, B. R., & Schmidhuber, J. (2016). LSTM: A Search Space Odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), 2222–2232. https://doi.org/10.1109/tnnls.2016.2582924

Hamayel, M. J., & Owda, A. Y. (2021). A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms. *AI*, *2*(4), 477–496. https://doi.org/10.3390/ai2040030

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Huang, Y., Capretz, L. F., & Ho, D. (2021). Machine Learning for Stock Prediction Based on Fundamental Analysis. *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, 01–10. https://doi.org/10.1109/ssci50451.2021.9660134

Islam, & Hossain, E. (2020). Foreign exchange currency rate prediction using a GRU-LSTM hybrid network. *Soft Computing Letters*, *3*, 100009. https://doi.org/10.1016/j.socl.2020.100009

Lipton, Z.C. (2015). A Critical Review of Recurrent Neural Networks for Sequence Learning. *ArXiv, abs/1506.00019*.

Liu, J. (2024). Navigating the Financial Landscape: The Power and Limitations of the ARIMA Model. *Highlights in Science Engineering and Technology*, *88*, 747–752. https://doi.org/10.54097/9zf6kd91

Liu, Y., Wang, Z., & Zheng, B. (2019). Application of Regularized GRU-LSTM Model in Stock Price Prediction. *In 2019 IEEE 5th International Conference on Computer and Communications (ICCC) (Pp. 1886-1890). IEEE.*, 1886–1890. https://doi.org/10.1109/iccc47050.2019.9064035

Lu, W., Li, J., Li, Y., Sun, A., & Wang, J. (2020). A CNN-LSTM-Based Model to Forecast Stock Prices. *Complexity*, *2020*, 1–10. https://doi.org/10.1155/2020/6622927

Lu, Z. (2024). Analysis of the Different Models for Stock Price Prediction. *Highlights in Science Engineering and Technology*, *88*, 130–135. https://doi.org/10.54097/6zq6mp91

Ma, M., & Mao, Z. (2020). Deep-Convolution-Based LSTM Network for Remaining Useful Life Prediction. *IEEE Transactions on Industrial Informatics*, *17*(3), 1658–1667. https://doi.org/10.1109/tii.2020.2991796

Minhaj, N., Ahmed, R., Khalique, I. A., & Imran, M. (2022). A Comparative Research of Stock Price Prediction of Selected Stock Indexes and the Stock Market by Using Arima Model. *Global Economics Science*, 1–19. https://doi.org/10.37256/ges.4120231426

Mondal, P., Shit, L., & Goswami, S. (2014). Study of Effectiveness of Time Series Modeling (Arima) in Forecasting Stock Prices. *International Journal of Computer Science Engineering and Applications*, *4*(2), 13–29. https://doi.org/10.5121/ijcsea.2014.4202

Navastara, D. A., Hidiya, F. R. A., & Wijaya, A. Y. (2023). Prediction of Indonesian Stock Price Using Combination of CNN and BiLSTM Model. *In 2023 International Conference on Information Technology and Computing (ICITCOM) (Pp. 307-312). IEEE.*, 307–312. https://doi.org/10.1109/icitcom60176.2023.10442941

Prakash, S., Jalal, A. S., & Pathak, P. (2023). Forecasting COVID-19 Pandemic using Prophet, LSTM, hybrid GRU-LSTM, CNN-LSTM, Bi-LSTM and Stacked-LSTM for India. *2021 5th International Conference on Information*

Systems and Computer Networks (ISCON). https://doi.org/10.1109/iscon57294.2023.10112065

Qin, A., & Boicu, M. (2023). A Fundamental Analysis of Stock Returns using Machine Learning Algorithms. *In 2023 IEEE MIT Undergraduate Research Technology Conference (URTC) (Pp. 1-5). IEEE.*, 1–5. https://doi.org/10.1109/urtc60662.2023.10534934

Saheed, Y. K., & Raji, M. A. (2022). Effectiveness of Deep Learning Long Short-Term Memory Network for Stock Price Prediction on Graphics Processing Unit. *2022 International Conference on Decision Aid Sciences and Applications (DASA)*, 1665–1671. https://doi.org/10.1109/dasa54658.2022.9765181

Salimath, S., Chatterjee, T., Mathai, T., Kamble, P., & Kolhekar, M. (2022). Indian Stock Movement Prediction with Global Indices and Twitter Sentiment using Machine Learning. https://doi.org/10.1109/csi54720.2022.9924056

Sandhya, P., Bandi, R., & Himabindu, D. D. (2022). Stock Price Prediction using Recurrent Neural Network and LSTM. *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)*, 1723–1728. https://doi.org/10.1109/iccmc53470.2022.9753764

Sarma, A., Singh, S., Jiang, H., Zhang, R., Kandemir, M. T., & Das, C. R. (2021). Structured in Space, Randomized in Time: Leveraging Dropout in RNNs for Efficient Training. *arXiv (Cornell University)*, *34*. https://arxiv.org/abs/2106.12089

Shi, Y. (2023). Research on the Stock Price Prediction Using Machine Learning. *Advances in Economics Management and Political Sciences*, *22*(1), 174–179. https://doi.org/10.54254/2754-1169/22/20230307

Si, Y., Nadarajah, S., Zhang, Z., & Xu, C. (2024). Modeling opening price spread of Shanghai Composite Index based on ARIMA-GRU/LSTM hybrid model. *PLoS ONE*, *19*(3), e0299164. https://doi.org/10.1371/journal.pone.0299164

Trivedi, N. D. V., & Patel, N. P. S. (2022). An Analysis of GRU-LSTM Hybrid Deep Learning Models for Stock Price Prediction. *International Journal of Scientific Research in Science Engineering and Technology*, 47–51. https://doi.org/10.32628/ijsrset229264

WAHYUDI, S. T. (2017). The ARIMA Model for the Indonesia Stock Price. *International Journal of Economics & Management*, *11*.

Xue, T., Qin, S., & Fu, Q. (2023). Stock Market's Price Movement Prediction with Multi-branch LSTM and Technical Analysis. *In 2023 International Conference on Neuromorphic Computing (ICNC) (Pp. 338-344). IEEE.* https://doi.org/10.1109/icnc59488.2023.10462810

Yang, L., Li, Y., Wang, J., & Sherratt, R. S. (2020). Sentiment Analysis for E-Commerce Product Reviews in Chinese Based on Sentiment Lexicon and Deep Learning. *IEEE Access*, *8*, 23522–23530. https://doi.org/10.1109/access.2020.2969854

Zafar, N., Haq, I. U., Chughtai, J., & Shafiq, O. (2022a). Applying Hybrid Lstm-Gru Model Based on Heterogeneous Data Sources for Traffic Speed Prediction in Urban Areas. *Sensors*, *22*(9), 3348. https://doi.org/10.3390/s22093348

Zafar, N., Haq, I. U., Chughtai, J., & Shafiq, O. (2022b). Applying Hybrid Lstm-Gru Model Based on Heterogeneous Data Sources for Traffic Speed Prediction in Urban Areas. *Sensors*, *22*(9), 3348. https://doi.org/10.3390/s22093348

Zeng, C., Ma, C., Wang, K., & Cui, Z. (2022). Parking Occupancy Prediction Method Based on Multi Factors and Stacked GRU-LSTM. *IEEE Access*, *10*, 47361–47370. https://doi.org/10.1109/access.2022.3171330

Zhang, S., Luo, J., Wang, S., & Liu, F. (2023). Oil price forecasting: A hybrid GRU neural network based on decomposition–reconstruction methods. *Expert Systems With Applications*, *218*, 119617. https://doi.org/10.1016/j.eswa.2023.119617

Zhu, H., Chen, S., Lu, W., Chen, K., Feng, Y., Xie, Z., Zhang, Z., Li, L., Ou, J., & Chen, G. (2022). Study on the influence of meteorological factors on influenza in different regions and predictions based on an LSTM algorithm. *BMC Public Health*, *22*(1). https://doi.org/10.1186/s12889-022-14299-y