Predicting Company Closing Prices: Exploring Informatics Techniques Across Regions And Industries

Marianthi Bitsika

4th Year Project Report
Computer Science and Management Science
School of Informatics
University of Edinburgh

2024

Abstract

Stock price prediction is an ongoing area of research that involves testing various combinations of indicators, models, and feature selection methods. This paper contributed to the literature by experimenting with novel combinations of Correlation Reduction, Mutual Information, and Recursive Feature Elimination, alongside the application of Haar wavelets to denoise closing prices. To assess the model's robustness, it underwent testing across various regions and industries. This involved analyzing the largest market capitalization stocks from North and South America, as well as Europe. The paper evaluated the pipeline of steps for cultivating the optimal model by comparing the application of ARIMA and LSTM models to both denoised and non-denoised prices, and then assessed the effectiveness of existing feature selection methods through manual input analysis. Through this critical examination of the evolution of a stock prediction model, this study uncovered the power of simplicity. All stocks showed optimal results with only denoised closing prices as inputs with eight out of nine preferring LSTM, and one ARIMA. The incorporation of additional inputs increased the mean squared error (MSE) by up to 100%.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(Marianthi Bitsika)

Acknowledgements

I extend my heartfelt gratitude to my supervisor, Daga Panas, for her invaluable guidance, encouragement, and unwavering support throughout the completion of my project. Her assistance and constructive feedback significantly contributed to refining my research work. Furthermore, I dedicate this paper to my grandparents, who unfortunately are no longer with us. Despite not always fully grasping my topic when I explained it to them, they consistently provided their support, and I am indebted to them.

Contents

Chapter 1

Introduction

Prediction of stock prices involves navigating a complex landscape influenced by a multitude of interconnected factors, including global economic data, unemployment rates, monetary policies, immigration policies, natural disasters, and public health conditions [29]. Stakeholders in the stock market grapple with these complexities in pursuit of higher profits while mitigating risks through diligent market evaluation. However, accurately predicting stock prices faces significant hurdles due to the inherent noise, non-linearity, and chaotic nature of stock markets [66].

Moreover, the efficient market hypothesis (EMH) casts a long shadow over stock prediction research. EMH posits that stock prices reflect all available information and follow a random walk, thereby rendering attempts at consistent out-performance futile [37]. Although some studies aim to identify anomalies or inefficiencies that may allow short-term predictability, the overarching influence of EMH underscores the complexity and uncertainty inherent in reliably forecasting stock prices. Navigating these challenges requires researchers to carefully consider the implications of EMH and to develop predictive models and strategies that account for market efficiency and the intricate interplay of factors shaping stock price dynamics.

Effective feature selection is crucial in model development. Incorporating all available features, including technical, fundamental, macroeconomic, and textual data from the financial market, can result in a complicated and difficult-to-interpret model [16]. This could potentially compromise its performance due to collinearity among multiple variables. In contrast, a well-crafted model equipped with an optimal set of attributes can provide reasonable predictions of stock prices and offer better insights into market dynamics. The selection of variables is crucial, as the attainment of the desired outcome hinges on identifying an optimal combination of features [29]. The significance of feature selection becomes even more apparent as it directly impacts the model's interpretability of features, which influences the predictive accuracy; this underscores the need for a thoughtful and strategic approach to variable inclusion.

To address the challenge of accurately predicting financial data, classical models such as the Auto-Regressive Integrated Moving Average (ARIMA) have been developed [9]. Although ARIMA models are commonly employed in modeling economic and

financial time series, they are limited in capturing nonlinear relationships between variables due to their assumption of constant standard deviation in errors, which may not accurately reflect real-world conditions [31]. However, their simplicity and interpretability remain advantageous, particularly in addressing the complexities of noisy financial data. Meanwhile, advances in computing power have spurred the exploration of machine learning and deep learning techniques, offering new avenues for prediction problems where complex relationships between variables are modeled hierarchically [29]. Among these, Long Short-Term Memory (LSTM), a type of Recurrent Neural Network (RNN), has gained prominence for its ability to capture complex and nonlinear patterns in time series data, making it particularly suitable for forecasting stock prices and other financial indicators [16]. Deep learning methods, including LSTM, have demonstrated success in various fields beyond finance, showcasing their potential for addressing intricate prediction tasks in diverse domains.

Existing models are typically domain-specific, focusing on parameters optimized for industry, market dynamics, or geographic distribution, to maximize model accuracy within their respective scopes. Nevertheless, there remains a gap in the literature regarding the exploration of universal algorithms that operate independently of such contextual constraints.

This study investigated the efficacy of utilizing an LSTM model in conjunction with three distinct feature selection techniques—Pearson's Correlation Coefficient, Mutual Information, and Recursive Feature Elimination—for the prediction of stocks across diverse global markets. Specifically, markets from the United States, Germany, and Brazil were carefully chosen to represent varying economic statuses and geographical distributions. Furthermore, stocks were meticulously selected based on their market capitalization, encompassing a diverse array of industries such as telecommunications, technology, and basic materials. The data inputs and pre-processing methodologies employed in this study have been delineated in Chapter 3. Concurrently, the model integrated technical, fundamental, and macroeconomic variables to ensure a comprehensive range of inputs. In Chapter 4, a detailed exposition of the tested model architectures is provided, along with a comprehensive evaluation spanning industries, economic conditions, and geographic locations. The main goal was to determine the effectiveness of the proposed model in accurately predicting global stock prices, specifically focusing on forecasting stock closing prices five days ahead.

1.1 Contribution

- Evaluated data pre-processing mechanisms, including normalization techniques, forward-filling, and data denoising.
- Analyzed optimal indicator combinations, encompassing basic technical indicators, advanced technical indicators, fundamental, and macroeconomic factors.
- Assessed the impact of feature selection by comparing scenarios with no feature selection to two combinations using three feature selection methods.
- Implemented Bayesian hyperparameter tuning for stock price prediction.

Chapter 2

Literature Review

The Auto-Regressive Integrated Moving Average (ARIMA) model is a traditional yet effective tool for stock prediction. Lakshmi and Radha applied ARIMA to forecast future prices of Adobe stock over a year, using data from 2014 to 2021. Results indicated a 12.2% increase in stock prices from September 2021 to February 2022, followed by a mild decrease of 2.6% in prices from February 2022 to March 2022. Additionally, there was an increase of 21.74% in prices from March 2022 to August 2022. ARIMA proved advantageous and highly efficient for short-term predictions, achieving a remarkably low Mean Absolute Percentage Error (MAPE) of 3.9% [68]. Similarly, Ariyo et al. explored ARIMA models for stock price prediction using data from the New York Stock Exchange (NYSE) and the Nigeria Stock Exchange (NSE), utilizing historical daily stock prices: Open-High-Low-Close (OHLC) from 1995 to 2011, to predict closing prices. They reported a Bayesian information criterion of 5.4 and a relatively small standard error of regression of 3.6 [1], illustrating its successful application across different geographies. However, despite its effectiveness, the emergence of newer technologies such as Support Vector Machines (SVMs) and Artificial Neural Networks (ANNs) have demonstrated superior accuracy in stock prediction. Zhang utilized data from 2015 to 2022 for five major stock indexes (DJIA, S&P 500, NASDAQ, Hang Seng Index, and FTSE 100) to predict stock prices, citing a 49.2% reduction in Mean Squared Error (MSE) and a 29.7% reduction in Root Mean Squared Error (RMSE) when employing LSTM, an ANN, compared to ARIMA [102]. LSTMs superior results were attributed to its capacity to handle non-linear relationships and to selectively forget abnormal data fluctuations. The paper suggested that with increasing data volume and complexity, LSTM models would have more advantages over ARIMA.

Several studies have compared ANNs with other machine learning models in stock prediction tasks to determine the most effective one. Kara et al. specifically compared SVM and ANN models to predict movements in the Istanbul Stock Exchange (ISE) National 100 Index [96]. Their study, spanning from 1997 to 2007, incorporated ten technical indicators that depicted the daily closing price fluctuations in the ISE National 100 Index. Linear scaling was applied to normalize each feature independently, preventing larger value inputs from overshadowing smaller ones and reducing prediction errors. The results showed that ANNs significantly outperformed SVMs, achieving an

accuracy of 75.7% compared to SVMs' 71.5%. The study highlighted the necessity of incorporating macroeconomic indicators to enhance accuracy, as the model struggled to predict Turkey's financial crisis in 2001 due to the absence of non-technical precursors to infer the state of the economy. Similarly, Karmiani et al. conducted a comparative study of LSTM, Backpropagation, SVM, and Kalman filter for nine US technology companies; stock price data was collected from Yahoo Finance, covering the period from January 2009 to October 2018. They concluded that LSTM demonstrated superior prediction accuracy with the smallest variance, achieving a mean accuracy of 68.6% across ten runs [18]. The paper revealed that the choice of algorithm should depend on specific requirements such as accuracy, variance, and speed. LSTM was preferred for high accuracy and low variance, while Backpropagation was suitable for high-speed applications. T-test results suggested that LSTM was more reliable compared to Backpropagation and SVM. Therefore, LSTM was preferred due to its transparent model architecture and superior performance with time series data. This prompted investigations into testing LSTM with wavelet transformations for denoising, feature selection, and diverse data [60].

2.1 Input Analysis

Basic technical indicators in stock prediction, which are open, close, high, low prices, and volume, constitute fundamental metrics widely employed in research. Chen et al. developed an LSTM model to forecast the closing prices of stocks in the Shenzhen and Shanghai (SSE) markets, utilizing data from 1990 to 2010 [47]. Initially, using only the five basic indicators yielded a low accuracy of 20.1%. However, upon incorporating the SSE Index, the accuracy increased to 24.1%, emphasizing the importance of basic technical indicators while also underscoring the requirement for supplementary data to grasp the intricacies of markets. Similarly, Gupta et al. compared the impact of closing price prediction for five US stocks using data comprising OHLC prices from 2012 to 2022 across three model architectures: Multilayer Linear Regression (MLR), Convolutional Neural Network (CNN), and LSTM algorithms [5]. LSTM consistently achieved the lowest MSE score, exhibiting an average decrease in MSE by 96.4% compared to MLR and 32.8% compared to CNN. In a similar study, Yan et al. achieved the highest accuracy of 73.8% in forecasting the closing price of American Airlines using LSTM with only basic indicators, surpassing the performance of Linear Regression (LR) and Decision Trees (DT)[89]. However, the accuracy values of the LSTM models in these studies were hindered by the use of only the basic technical variables, calling for the use of advanced technical indicators.

The utilization of advanced technical indicators has significantly enhanced stock market prediction accuracy. Mittal and Chauhan compared the inclusion of 20 advanced technical indicators to the 5 basic technical indicators in their LSTM model to forecast ITC Limited's closing price in the Indian market, using data from 2010 to 2019 [74]. Their study revealed that the incorporation of advanced technical indicators led to an average decrease of 7.1% in MAPE and a decrease of 76.8% in RMSE across different sliding window sizes. Subsequently, the effectiveness of advanced technical indicators has been extensively evaluated across various models. Agrawal et al. compared SVM,

LR, and LSTM to predict the closing price of three Indian stocks, incorporating only four advanced technical indicators: Relative Strength Indicator (RSI), Moving Average (MA), Stochastic Oscillator (% K), William (% R), and Exponential Moving Average (EMA) [51]. LSTM consistently outperformed SVM and LR with an average accuracy of 59.3% compared to 52.0% and 53.0%, respectively. Thus, incorporating advanced technical indicators in LSTM substantially enhanced accuracy and remains a cornerstone in stock market prediction strategies.

Although fundamental indicators have traditionally been associated with monthly and longer-term predictions [95], recent research has indicated their importance in short-term (daily) predictions [55]. Firstly, fundamental indicators have been heavily implemented for longer-term prediction; their frequency has been shown to better reflect longer timeframes. As presented by Kamble, technical indicators were deemed insufficient for predicting buy and sell decisions for stocks in the NSE and Bombay Stock Exchange (BSE) one year in advance [65]. Thus, Kamble included five fundamental indicators and utilized Random Forest (RF), to achieve an accuracy of 85.8%. Chai et al. similarly benefited from the introduction of fundamental indicators in predicting the movement of the CSI300 index in China one year ahead, by implementing the New Loan/Market Capitalization Ratio to capture the state of the firm in the domestic economy [38]. Their model, an empirical mode decomposition least squares support vector, reached a MAPE of 0.8%. Moreover, Zhou and Qu tested the implementation of fundamental indicators in short-term prediction. They combined technical and fundamental indicators into both their ARIMA and LSTM models to forecast the next day's price for 10 stocks in the S&P 500, using data from 2004 to 2013. [101]. The ARIMA model garnered an MSE score of 0.05 for Google, while the LSTM achieved an average MSE score of 0.04. Similarly, Li et al. compared the impact of integrating fundamental indicators into short-term predictions for stocks on the SSE, using data from 2010–2021 for various models such as Adaboost, Bagging, LSTM, RF, and their proposed architecture: Pearson Correlation Coefficient for feature selection with Broad Learning System (PCC-BLS). PCC-BLS had the lowest error metrics for all tested stocks, emphasizing the significance of feature selection when utilizing a diverse range of indicators and a large dataset [23].

Finally, macroeconomic variables, including exchange rates, commodities, economic performance metrics, and interest rates, play a pivotal role in predictive models for stock prices. Zhong and Enke incorporated multiple exchange rates to predict the daily direction of the S&P 500, using data from 2003–2013. They implemented technical and fundamental indicators along with exchange rates between the US dollar (USD) and four major currencies [91]. Employing PCA for dimensionality reduction into an ANN model, they achieved an average accuracy of 58.1%. Similarly, Saeed and Jamil analyzed the effect of USD fluctuations on the closing price of Fauji Fertilizer stock from the Pakistan Stock Exchange, using OHLC data spanning 2011–2021 [50]. Their objective was to correlate the impact of the USD exchange rate on stock prices and to predict the closing price of the stock. They tested various machine learning models such as LR, ANN, RF, DT, and SVM, and realized that the correlation between USD and the prediction of the stock price varied between models. Specifically, they observed a correlation of 94% for ANN, 95% for RF, 78.53% for SVM, 78% for LR, and 100% for DT. Consequently, DT and RF performed the best on the data, as confirmed by

their lowest MSE, RMSE, and MAE scores. Hence, the USD exchange rate provided insight into modeling volatile markets. Therefore, 96.2% of exchange rate variables implemented in stock prediction models include USD, reflecting its significant influence on global economies, particularly as commodities like oil and metals are priced in USD [20].

Lakshminarayanan and McCrae compared the prediction of a dataset containing only Dow Jones index (DJI) data against a dataset enriched with oil prices, spanning from 2014 to 2018 [72]. When tested on both SVR and LSTM models with moving averages, the inclusion of commodity prices resulted in a decrease in RMSE by 14.7%, in MSE by 27.4%, and in MAE by 18.7% for SVR, while for the LSTM model, it reduced RMSE by 13.0%, MSE by 27.0%, and MAE by 24.4%. The LSTM model outperformed SVR, and the dataset containing commodities demonstrated optimal performance, showcasing the significance of incorporating even a few commodity prices. Furthermore, Dingli and Fournier emphasized the significance of commodities in forecasting both short-term and long-term movements in the stock market. They incorporated currency exchanges, gold and oil prices, technical indicators, and historical prices spanning from 2003 to 2016. Their approach involved employing a CNN model to predict weekly and monthly stock price fluctuations. Their model achieved an accuracy of 65% when forecasting the next month's price direction and 60% for the next week's price direction forecast [4], illustrating a successful application of commodities for both short-term and long-term predictions.

Pan examined the relationship between stock prices and unemployment in 30 countries, both developed and developing [86]. The study revealed that in G7 countries, stock prices predict unemployment. In other developed countries, there exists a bilateral relationship between stock prices and unemployment. However, in developing economies, the relationship is one-way, with unemployment predicting stock prices but not vice versa. Similarly, Tsai et al. applied unemployment and interest rates to predict the quarterly rate of return in Taiwan's electronic industry, using data from 2002 to 2006 [15]. The highest performing model, Bagging, achieved an accuracy of 66.7%. Finally, Tufekci's study on predicting the movement direction of the Istanbul Stock Exchange (ISE) National 100, with LR, SVM, and MLP using data from 2007 to 2012, implemented commodity prices, CPI, exchange rates, and unemployment statistics. The study achieved the highest accuracy with LR, reaching 55.6% without feature selection and the highest accuracy of 60.8% again with LR [63]. This highlighted that the incorporation of numerous macroeconomic indicators would require feature selection to properly distill essential features and reduce manually inputted noise.

2.2 Wavelet Transformation Analysis

Haar wavelets, a mathematical tool employed in signal processing, have received attention for their effectiveness. Wang's study demonstrated the effectiveness of wavelet transformation in forecasting the closing price of the SSE from 1993 to 2009, with the non-denoised model undeperforming all variants of a denoised Backpropagation neural network [44].

Further studies have illustrated the versatility of wavelets in predictive modeling. Wang and Gupta employed the Daubechies wavelet in a feedforward backpropagation neural network to denoise S&P 500 time series data spanning from 1950 to 2010. The denoising process aimed to eliminate white noise by applying various thresholds at each level of decomposition and by reconstructing the denoised signal. Subsequently, the denoised signal was partitioned into matrices of appropriate sizes for the neural network training, validation, and testing phases. However, despite these efforts, the directional efficiency for movement prediction saw only a modest improvement of 2.3% [48]. This marginal enhancement in prediction accuracy might be attributed to several factors, including the sheer volume of the data, the choice of wavelet, and the potentially excessive level of denoising, set at 5, which could have led to information loss. In contrast, Chandar et al. implemented Haar wavelets in a backpropagation neural network to forecast the closing prices of five Indian stocks using OHLC prices from 2010 to 2015 [73]. The application of Haar wavelets yielded a noteworthy average reduction in RMSE by 31.4%, and a significant decrease by 120.7% in MSE across the five stocks. This large improvement in error metrics, compared to the findings of Wang and Gupta, may stem from differences in the choice of wavelet, the range of data examined, and the nature of the data itself, as Chandar et al. analyzed individual stock companies rather than stock indexes. The amplified benefit of Haar wavelets when applied to ANNs has been corroborated by Bao et al., who predicted the price movement of six international stock indexes—CSI 300, Nifty 50, Hang Seng, Nikkei 255, S&P 500, and DJIA—using technical and macroeconomic indicators in LSTM[84]. The denoised LSTM model resulted in an average decrease in MAPE by 29% across the six indexes compared to the non-denoised LSTM. The paper also tested the incorporation of stacked autoencoders into the denoised LSTM, with the objective of extracting deep daily features in an unsupervised manner. This model was found to be optimal, and the advantages were less apparent in less developed markets than in developed markets.

In addition, the integration of Haar wavelets with ARIMA has shown promising results. Shan et al. implemented a maximal overlap discrete wavelet transformation to predict stock prices for the Shanghai and Shenzhen stock markets, utilizing data from 2010 to 2012 [67]. The ARIMA model with denoising outperformed the one without for both stock indexes, achieving a 36.5% reduction in MAPE and a 47.5% reduction in RMSE across the two indexes. The successful incorporation of wavelets into ARIMA was further confirmed by Wang and Guo's study, where the authors compared ARIMA models with discrete wavelet and XGBoost and without any additional pre-processing for ten Chinese companies with data from 2010 to 2015 [99]. The incorporation of wavelets and XGBoost resulted in an average 36% increase in accuracy, illustrating the benefits of Haar wavelet in denoising.

2.3 Feature Selection Analysis

As highlighted by Cai et al., in stock market analysis, numerous factors impact price fluctuations, necessitating meticulous feature selection before employing machine learning models to improve prediction accuracy [87]. Kim et al. emphasized that achieving the optimal set of features would require striking a balance between providing sufficient

information for predictions and avoiding computational complexity and overfitting [97]. Thus, prioritizing impactful features was vital for successful predictions.

Li et al. employed the Pearson Correlation Coefficient (PCC) to identify the most representative features from a pool of 35 variables, utilizing a PCC threshold of 0.5 [24]. Their objective was to predict one-day-ahead closing prices using a Broad Learning System (BLS) model, focusing on four stocks from the Shanghai and Shenzhen stock exchanges. Drawing from 11 years of data (2010-2021), their research showcased that the PCC-BLS methodology outperformed 10 single machine learning techniques, including LSTM, CNN, and GRU. Compared to the second-ranked model, BL, the PCC-BLS approach exhibited a 9% reduction in MSE and an increase in R^2 from 0.945 to 0.950. This study accentuated that even simple feature selection methods could have an impact on predictive accuracy.

Alsubaie et al. utilized three filter feature selection methods—correlation, Gain Ratio, and ReliefF—to identify optimal technical indicators to predict the stock movements of 100 stocks in the Saudi Arabian market [92]; Gain Ratio is a variant of mutual information. These methods were integrated into a trading strategy framework and assessed across SVM, ANN, and Naive Bayes (NB) models. The study revealed that the choice of feature selection method depended on both the model and the variables tested. For instance, ANN preferred ReliefF for accuracy optimization, while correlation minimized costs, and Gain Ratio increased investment returns. This variability underscored the absence of a universal solution, necessitating further investigation. Similarly, Gunduz et al. used Gain Ratio and ReliefF to enhance a Gradient Boosting Machine model for stock price forecasting on the Istanbul exchange. Their study stressed the value of feature selection due to the high dimensionality of financial data. While two stocks favored Gain Ratio, it slightly underperformed for the third stock, highlighting its overall effectiveness for refining trading strategies.[28].

Wrapper filters represent another category of feature selection methods. Yuan et al. employed a Recursive Feature Elimination (RFE) algorithm based on SVM to identify a subset of features from 60 categories, aiming to enhance prediction accuracy for Chinese A-share stocks [90]. The incorporation of RFE resulted in a slight average improvement in SVM model accuracy, reaching 51.78% compared to 51.73% without feature selection; some stocks preferred no feature selection over SVM-RFE. In a comparative analysis, SVM-RFE was compared with RF; the latter emerged as the optimal choice, increasing the Sharpe ratio by 23.5% compared to no feature selection and by 22.5% over SVM-RFE. However, recognizing the limitations of SVM-RFE, Botunac et al. explored the application of RFE with various regressors, including LR, DT, and RF, to predict the closing prices of three US technology stocks; the study implemented RFE into LSTM [33]. Interestingly, each stock demonstrated a preference for a different regressor. Nevertheless, the combination of RFE with an RF regressor emerged as the top performer, outputing an MAE of 0.01547 and an MSE of 0.00041. These studies illustrated the dynamic nature of feature selection methods and the importance of selecting the most suitable approach, tailored to specific datasets and prediction tasks.

Chapter 3

Methodology & Data

3.1 Proposed Approach

To evaluate the optimal combinations of inputs, feature selection methods, and machine learning models, various versions of ARIMA and LSTM models were tested. All trials, except those that exclusively involved the closing price, underwent Algorithm 1. To evaluate the impact of denoising, both non-denoised (ND) and denoised (D) closing prices were compared. Denoised closing prices were utilized with all indicator types: T for Technical indicators, M for Macroeconomic indicators, F for Fundamental indicators, ER for Exchange Rate and COM for commodities; all technical indicators encompasses both basic and advanced technical indicators. Finally, three feature selection methods were tested: C denoting Correlation Removal through PCC, MI indicating Mutual Information, and RFE representing Recursive Feature Elimination. The structure of this section was in line with the sequence of steps outlined in Algorithm 1. The models were executed on Google Collab using the T4 GPU and Python 3.0.

Table 3.1: Model Combinations

ARIMA	**LSTM**
ND Close Price	ND Close Price
D Close Price	D Close Price
	Basic T with D Close Price
	All T with D Close Price
	All T + F with D Close Price
	All T + M with D Close Price
	All T + ER with D Close Price
	All T + COM with D Close Price
	All T + M + F with D Close Price
	All T + M +F with C + MI with D Close Price
	All T + M +F with C + MI + RFE with D Close Price

Algorithm 1 Proposed Approach

```
Denoise Close Price
Calculate advanced technical indicators based on denoised close price
Create a complete dataset with technical, fundamental, and macroeconomic indicators, all correctly forward-filled to 5296 days
Calculate the skewness of each column
if skewness is NAN then
    Drop the column
else
    Continue
end if
Normalize Dataset
Feature Selection Testing: No Feature Selection, C+MI, C+MI+RFE
```

3.2 Data Sourcing

The dataset comprised historical data for nine different stocks, involving three stocks from each of the three countries representing North America, Europe, and South America. The country selected to represent each continent was cited to have the most literature in Kumbure et al.'s review of stock market forecasting, which referenced over 138 papers [55]; this ensured the availability of publicly accessible data for sourcing. Furthermore, the selected stocks represented some of the largest market capitalization companies in their respective countries, guaranteeing a robust representation of each market [103]. By opting for the largest market stocks, the selection avoided bias toward any particular industry, enabling a comprehensive portrayal of the global market. This selection encompassed both developing and developed economies, facilitating a comprehensive analysis aimed at understanding the data inputs necessary to create a global stock prediction model. This approach aligned with the objective of identifying the most universally successful combination of inputs.

Table 3.2: Stocks and Their Industries by Country and Continent

Continent	Country	Stock	Industry
Europe	Germany	SAP	Technology
		Siemens	Technology
		Deutsche Telekom (DT)	Telecommunications
North America	USA	Apple	Technology
		Microsoft (MSFT)	Technology
		Alphabet	Technology
South America	Brazil	Ambev	Beverages
		Petroleo Brasiliero (PB)	Energy: Oil and Gas
		Vale	Metals and Mining

While Asia had been initially considered, it was ultimately excluded due to challenges

encountered during data sourcing. Taiwan, the first Asian country included, lacked separate macroeconomic data from China. Furthermore, companies in India, the second most popularly cited Asian developing nation, showed significant gaps in fundamental data, making them unsuitable for inclusion. Consequently, the analysis focused on North America, Europe, and South America to ensure a more comprehensive and reliable dataset.

The dataset spanned 14 years from March 31, 2009, to September 30, 2023, capturing significant market events including two major bear markets—the aftermath of the financial crisis in 2008 and the COVID-19 pandemic in 2020. This timeframe selection ensured a comprehensive representation of market dynamics, encompassing both bullish and bearish market conditions.

To ensure uniformity among variables, monthly and quarterly data were converted to daily using forward filling [29]. This conversion facilitated consistency in the dataset and enabled the inclusion of a wide range of variables for analysis.

The data was divided into training and tested sets with an 80/20 split. Within the training set, a further split was made into training and validation data using the same 80/20 ratio as customary [29, 74, 23, 73]. The validation data was utilized for hyperparameter tuning, while the remaining training data was used to predict values five days in advance.

A comprehensive list of all indicators has been included in Appendix A.1.

3.2.1 Technical Indicators

Technical indicators are mathematical calculations based on historical price, volume, or open interest data aimed at providing information on market trends, momentum, volatility, and volume of a stock [13]. They help analysts make informed decisions by identifying patterns, signals, and potential market reversals. These indicators have become increasingly sophisticated and are now integral to stock price prediction [29]. Thus, technical indicators are essential for stock market prediction and can be used alone to predict prices or in combination with other input categories.

Basic technical indicators, which include the open, close, high, and low prices along with volume, were directly extracted from historical stock market data sourced from Yahoo Finance; they have been defined in A.1. Advanced technical indicators utilize basic indicators through mathematical formulas to predict price movement. According to [61], advanced technical indicators simplified the analysis by summarizing the behavior or trends in the time series, compared to basic technical indicators which are raw prices, making them more appropriate for stock price prediction. They are categorized as follows:

1. **Momentum Indicators**: These indicators measure the speed with which a price was changing over time [29]. Momentum indicators aid in identifying trend-lines and potential market reversals [61] and have been defined in A.2.
2. **Trend Indicators**: Trend indicators focus on the direction and strength of a change in price [13]. They determine whether a stock is in an uptrend, downtrend, or trading sideways. The trend indicators utilized have been outlined in A.3.

3. **Volatility Indicators**: These indicators measure how much a variable, such as price, is changing and fluctuating during a certain time period. They help traders assess the level of risk associated with a particular stock or market [13]. The volatility indicators implemented have been shown in A.4.

In this analysis, the five basic indicators were included along with the 20 advanced technical indicators from Kumbure et al.'s literature review [55]. Advanced indicators were calculated either manually or using existing Python packages such as Finta [64] or Algorithmic Trading with Python [58]. The selected advanced indicators were carefully chosen to encompass all advanced technical indicator categories while avoiding redundancy and excessive overlap. Additional volume indicators were omitted due to their infrequent appearance within the top 30 variables for stock prediction models in Kumbure et al.'s literature survey of advanced technical indicators for stock prediction, which underscored their limited effectiveness [55]. Technical indicators were generated for every Monday to Friday (the days the stock market would be open), with weekend data forward-filled to create a daily dataset.

3.2.2 Fundamental Indicators

Fundamental indicators are financial metrics derived from a company's statements, such as the balance sheet, cash flow statement, and key financial ratios [34]. These indicators offer insight into a company's financial health, performance, and operational efficiency, thereby assisting investors in evaluating investment opportunities.

1. **Balance Sheet:** The balance sheet is a financial statement that provides a snapshot of a company's financial position at a specific point in time. It comprises assets, liabilities, and shareholders' equity [29]. These variables have been outlined in A.6.

2. **Cash Flow Statements:** The cash flow statement tracks the flow of cash into and out of a company over a specific period. It consists of three main sections: operating activities, investing activities, and financing activities [34]; the cash flow statement indicators utilized have been added in A.5 and A.7.

3. **Key Financial Ratios:** Key financial ratios are calculated using data from both the balance sheet and the income statement. These ratios provide insights into a company's financial performance, profitability, and efficiency [76]; the ones implemented in this study have been described in A.8.

Fundamental indicators were provided on a quarterly basis and were sourced from MacroTrends, which utilizes data from Zacks Investment Research Inc. [103].

3.2.3 Macroeconomic Indicators

Macroeconomic indicators are key metrics that provided insight into the overall health and performance of an economy. These indicators play a crucial role in understanding economic trends, informing policy decisions, and evaluating investment opportunities. The following categories of macroeconomic indicators have been implemented in this study:

1. **Commodity Prices:** Commodity prices, containing various goods from precious metals to agricultural products, serve as vital inputs for stock price prediction, reflecting economic conditions and market interconnections [7]. Short-term fluctuations in commodity prices can signal changes in inflation, interest rates, and market sentiment, while longer-term movements often correlate with future economic performance, offering insights into stock market trends. These prices, sourced from the IMF Commodity Data Portal, were accessible on a monthly basis [35]; they have been outlined in A.9.

2. **Exchange Rates:** Exchange rates, representing the value of one currency as compared to another, significantly impact international trade profitability, export-import competitiveness, and multinational corporation valuations. They also serve as indicators of global economic shifts, influencing worldwide stock markets [30]. Since Forex markets were closed on weekends, exchange rates were unavailable during this time, necessitating forward-filling for daily datasets [32]. The exchange rates implemented in this study have been tabulated in A.10.

3. **GDP Statistics:** GDP statistics, measuring a country's total production value, encompass variables related to productivity, trade, income, investment, and international liquidity. These statistics, sourced from the IMF Gross Domestic Product and Components portal, were updated quarterly [36] and described in A.12, A.13.

4. **CPI:** CPI is a vital measure of price changes for goods and services, offering insights into inflation and purchasing power. It is closely tied to interest rates, impacting stock returns, as demonstrated in the findings of Ferson et al. [22]. Data were sourced from the IMF's International Financial Statistics which updated the CPI monthly [36].

5. **Labor Statistics:** Labor statistics provide insights into various aspects of the labor market, including employment levels, unemployment rates, and wages. These data, sourced from the IMF Prices, Production, Labor, and Trade Portal, were accessible on a monthly basis [36]; they have been defined in A.11.

3.3 Wavelet Transformation

Wavelet transformation is a fundamental mathematical framework used to decompose signals into distinct frequency bands [73]. Originating from Fourier analysis, wavelet theory enables the representation of any function as a combination of sine and cosine functions. This decomposition process offers a comprehensive analysis of signals in both time and frequency domains, enabling detailed examination of signal features.

The Discrete Wavelet Transform (DWT) breaks down the original signal $O(t)$ into approximation coefficients $A(t)$ and detail coefficients $D(t)$. These coefficients capture different frequency components of the signal at various scales and translations. $A(t)$ covers the signal's low-frequency components, derived from the low-pass filtered signal, while $D(t)$ encapsulates high-frequency details. This decomposition process offers a comprehensive analysis of signals in both time and frequency domains, enabling detailed examination of signal features.

A mathematical breakdown of wavelets has been shown in Appendix A.2.2.

3.3.1 Application

The Haar wavelet is commonly applied in all non-custom wavelet methods for stock market prediction, as evident from these papers [73, 82, 49]. The Haar wavelet is particularly effective in capturing fluctuations between adjacent observations and addresses the issue of aliasing. As mentioned by Yang and Liang et al., Haar wavelets offered several advantageous properties [100, 49]: first, they exhibited tight support, with significantly sharp drop-off performance. Secondly, the support length was short, which helped reduced computation and data processing time. Finally, they were symmetric, leading to lower distortion rates during signal decomposition and reconstruction.

There exist two threshold functions: soft threshold and hard threshold denoising. The soft thresholding method was chosen for its effectiveness and widespread application in stock prediction literature [100, 49, 73, 82]. In soft thresholding, coefficients with absolute values lower than the threshold λ are set to 0, while the absolute value of non-zero coefficients are subtracted from the threshold. The expression is as follows, where ω represents the coefficient and λ is the threshold [100]:

$$\omega_\lambda = \begin{cases} \text{sign}(\omega) \cdot (|\omega| - \lambda) & \text{if } |\omega| \geq \lambda \\ 0 & \text{otherwise} \end{cases}$$

In contrast, hard thresholding resets coefficients with absolute values lower than the threshold λ to 0 but retains non-zero coefficients. Thus, soft thresholding, being a continuous method, avoided introducing abrupt changes and additional oscillations to the signal, unlike hard thresholding methods.

Next, the level of decomposition was set to 2, as is customary for individual stock companies as in [73]. The threshold value was determined through the variance of the noise of the closing price, following the methodology proposed by Donoho and Johnstone [19]. Algorithm 2 (see Appendix A.2.2) has outlined the pseudocode of the denoising process and the functions implemented.

3.4 Dataset Pre-Processing

To ensure seamless integration of datasets featuring diverse indicator frequencies—quarterly, monthly, and Monday-to-Friday data—a standardization process was undertaken, aligning all datasets to a uniform span of 5296 days through forward filling. This approach, endorsed by Bhandri et al. [29], facilitated consistency and comparability across the dataset. In addition, Alqahtani and Abdelhafez [53] have demonstrated the efficacy of forward filling in predicting stock trends within Saudi Arabian markets, encompassing a spectrum of industries such as energy, banking, telecommunications, basic materials, insurance, and real estate management. Given the congruence of industries between their study and this one, forward-filling was implemented.

Consequently, each indicator experienced a varying amount of forward filling; this led to a wide range, with GDP statistics filled forward 98.92%, commodity data 97.36%, and exchange rate data only 24.77%. These indicators were common for all stocks. International Liquidity, CPI statistics, and Labor statistics were unique to each country and were forward-filled approximately 97.37%, 97.1%, and 97.08%, respectively. Finally, technical indicators exhibited some variance in percentage forward-filled, ranging from 15.82% for Apple to 20.53% for PB, while fundamental indicators were steady at around 98% for each stock. For a breakdown of the percentage, each indicator was forward filled see Tables A.14, A.15.

In response to the widespread presence of forward-filled data, the skewness of each column was computed to discern and rectify columns containing extreme constant values. Consequently, 30 indicators were eliminated for all stocks, as has been detailed in Table A.16 in the Appendix. These indicators included GDP and CPI statistics, identified by Patatoukas's study as having a disconnect with stock market data [62]. This step reinforced data integrity and ensured accurate analysis.

3.4.1 Normalization

Additionally, the 193 indicators were in different units, including prices, ratios, and volume quantities. Consequently, data normalization was vital for an accurate prediction [23]. A study by Nayak, Misra, and Behera on predicting the closing price of the BSE revealed that various model architectures preferred different normalization types [77]. Therefore, four normalization techniques were tested: min-max, robust, z-score, and a hybrid method presented in Kumari et al., which combined the previous three methods, to determine the optimal normalization method for LSTMs [11]. The equations for all normalization methods have been expressed in Appendix A.2.3. These normalization techniques were tested on validation data for the LSTM model with all indicators and no feature selection. The models were optimized using Bayesian Optimization, as discussed in Section 3.9.1.

Min-max normalization, a widely used method, scales each feature's minimum value to 0 and its maximum value to 1, while adjusting all other values to a decimal between 0 and 1. This method is commonly employed in stock prediction due to its simplicity and ability to maintain relative relationships between values as evident in past literature [29, 85, 23]. However, it is sensitive to outliers, which can skew the normalization process and potentially lead to loss of information, particularly if the data range is large.

Z-score normalization is another approach that employs the mean and standard deviation of the data to normalize values, effectively managing outliers and preserving data distribution. While z-score normalization allows for comparison between raw scores from different tests and accounts for both mean value and variability, it has limitations such as assuming a normal distribution and uneven distribution around the origin line if the data is skewed.

While not commonly used in stock market prediction, robust normalization was integrated in the hybrid technique. It handles outliers by excluding them from mean and standard deviation calculations. This approach reduces the impact of extreme values on

scaling, but could overlook potentially valuable data points in volatile stock market data [11].

Kumari and Swarnkar developed a hybrid normalization methodology that combined elements of the aforementioned normalization techniques, as described in Algorithm 3 and in Appendix A.2.3 [11]. This technique achieved comparable accuracy to min-max normalization, demonstrating slightly better or similar accuracy in predicting six indexes: BSESN, NIFTY50, NASDAQ, HAND SSENG, NIKKEI255, and SSE. Therefore, due to its high performance with international indexes, it underwent testing with international stocks.

Overall, min-max normalization emerged as the most effective for LSTM prediction, showing the lowest MSE score across all stocks, as illustrated in Figure 3.1. Thus it was applied to all LSTM models in this study. Alphabet and PB would experience a high MSE score with robust normalization because the removal of outliers would negatively affect the predictive accuracy of volatile stocks. Therefore, this was extended extended to the hybrid method.

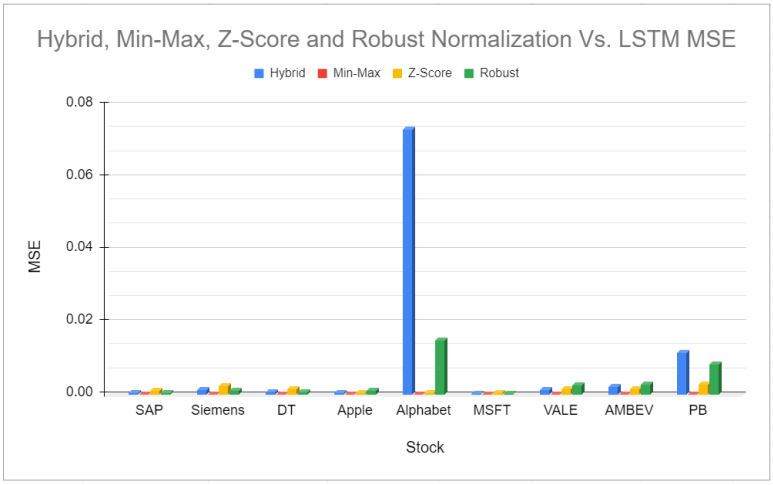


Figure 3.1: Comparison of MSE of LSTM on Validation Data with Different Normalization

3.5 Feature Selection

A combination of feature selection methods was employed, as Tsai and Hsiao's analysis of feature selection methods highlighted that combinations yield higher accuracy by removing a wider range of unrepresentative features [14]. Algorithm 4 illustrates the sequence of feature selection methods in Appendix A.2.4.

3.5.1 Correlation Reduction

The Pearson Correlation Coefficient (C in this study), introduced by Pearson in 1895, served as a fundamental tool for measuring the strength and direction of the relationship between two variables [24]. Its efficacy in eliminating redundancy within datasets has been demonstrated in various studies [24, 29, 27]. Thus, since there were instances of

redundancy pervasive within the dataset (such as commodities segmented by region) and exchange rates covered the same countries, correlation reduction was deemed appropriate. The formula for deriving the C between variables x_i (stock features) and y (stock price) has been shown below:

$$r_{x,y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}, \quad (5) \tag{3.1}$$

The absolute value of $r_{x,y}$ determines the strength of the correlation, ranging from +1 to -1. A value of 1 signifies a perfect positive linear correlation, -1 represents a perfect negative linear correlation, and 0 indicates no linear correlation [24]. A predefined correlation coefficient threshold of 0.80 was implemented, as recommended by Bhandari et al. [29].

3.5.2 Mutual Information

Mutual information (MI), a measure of statistical independence between two random variables, offers a more generalized approach for detecting nonlinear relationships compared to traditional linear correlation measures [88]. Its significance lay in its close association with entropy, rooted in Shannon's entropy theory [8]. Incorporating mutual information-based methods, such as the Forward Selection Minimal-Redundancy-Maximal-Relevance (FSMRMR) and Conditional Mutual Information Maximization (CMIM) methods, as demonstrated by Sun et al. and Chen and Hao, respectively, could offer improved predictive modeling outcomes by capturing more nuanced relationships among features [41, 93].

The mutual information between two random variables X and Y has been mathematically expressed as:

$$I(X;Y) = \sum_{x \in X}\sum_{y \in Y} p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right) \tag{3.2}$$

where $p(x,y)$ is the joint probability distribution of X and Y, and $p(x)$ and $p(y)$ are the marginal probability distributions of X and Y, respectively [88]. This paper selected features with mutual information values greater than 1.0 in relation to the denoised closing price. A hard threshold was chosen for uniformity across all industries and countries, as previous literature did not suggest a specific threshold value.

3.5.3 Recursive Feature Elimination

Recursive Feature Elimination (RFE) is a wrapper feature selection method that utilizes machine learning models to compute the relevance scores of features [26]. Initially, RFE trains a model with the entire feature set and assigned a relevance score to each feature. Subsequently, the feature with the lowest relevance score is removed, and the

model is re-trained to compute new feature relevance scores [40]. This iterative process continues until the desired number of features remains in the feature set.

3.5.3.1 Application

RFE was implemented utilizing a Random Forest Regressor (RF) as it outperformed linear and decision tree regression in Botunac's analysis of the underlying regressors of RFE [33]. The advantage of using the RF as the underlying machine learning model for RFE lay in its robustness and accuracy. RF is an ensemble learning method that combines the predictions of multiple decision trees to produce more accurate and stable predictions [98]. It is less likely to overfit the data, can handle noisy data and outliers, and provides a measure of feature importance. Additionally, RF can handle large datasets, making it suitable for feature selection in this application.

RFE selects the most important number of features, based on a predetermined value of how many features to output. Previous literature did not justify authors' reasoning behind selecting a specific number of features. Therefore, RFE was implemented for three different numbers of features: 25%, 50%, and 75%. The percentage that was correlated with the smallest MSE score was selected. These trials were run on the validation data.

Table 3.3: Optimal Number of Features and Mean Squared Error (MSE)

Metrics	SAP	SIEMENS	DT	APPLE	MSFT	ALPHABET	AMBEV	PB	VALE
Optimal Number of Features (%)	50.00	50.00	25.00	25.00	25.00	75.00	50.00	50.00	25.00
MSE	0.00023	0.00814	0.01169	0.03172	0.03859	0.03887	0.00020	0.00056	0.05731

3.6 Overview of LSTM

LSTM networks have gained prominence in time series prediction tasks within Recurrent Neural Networks (RNNs) due to their ability to address the vanishing gradient problem and retain long-term dependencies [79]. LSTM was selected due to its high performance compared to other approaches, such as SVM and ARIMA, which were the most researched machine learning methods after LSTM [29, 72, 75]. Structurally, LSTM comprises an input layer, a hidden layer, a memory cell, and an output layer, designed to overcome the challenge of retaining long-term dependencies encountered by conventional RNNs due to the vanishing gradient problem [70].

The LSTM cell integrates three crucial sources of information: the current input sequence x_t, the short-term memory from the previous cell h_{t-1}, and the long-term memory from the preceding cell state c_{t-1} at time t. Input gates facilitate the input of data into memory cells, while forget gates selectively manage the retention or deletion of data from the memory cell for subsequent inputs. The forget gate evaluates the information from x_t and h_{t-1}, assigning values between 0 and 1 to selectively retain or discard the information from c_{t-1} [29]. This gate plays a pivotal role in maintaining stability by regulating the flow of information, allowing data to pass through the cell without significant alteration. The output gate determines the output data from the memory cell [79]. This structural design enables the memory cell to accumulate

values over varying time intervals, facilitating effective learning of context-specific temporal dependencies. LSTMs gather information over short or extended periods without the need for explicit activation functions within the recurrent components [12]. Consequently, LSTMs effectively address the vanishing gradient problem by ensuring stable memory cell contents over time, leading to enhanced performance across various sequential learning tasks [12]. For a deeper analysis see Appendix A.3.1.

3.7 Overview of ARIMA

ARIMA, also known as the Box-Jenkins model, is a stochastic time series model widely utilized in financial forecasting due to its simplicity and effectiveness [9]. These linear models, extensively studied by Hamilton and Contreras et al., can be effectively applied to forecast the behavior of economic and financial time series [45, 39]. ARIMA models are renowned for their efficient capability in generating short-term forecasts, sometimes outperforming complex structural models [1]. However, as evidenced in the literature review, since ANNs often outperformed ARIMA, ARIMA has commonly been used as a baseline for comparison against newer time series prediction models, such as LSTM [31].

The ARIMA model building process involves three main steps: model identification, parameter estimation, and diagnostic checking [9]. First, the Autoregressive (AR) component utilizes lagged values of the target variable to regress on its own past values. This involves performing partial autocorrelation to forecast the time series using multiple lagged observations. Second, the Integrated (I) component applies differencing to the data to reduce seasonality, making the series more stationary. Differencing involves subtracting the current value from the previous one to create a new series, which is generally more suitable for modeling. Lastly, the Moving Average (MA) component predicts future values using error terms from previous observations. The parameters of the ARIMA model include: p, the count of lagged observations; d, the degree of differencing; and q, the size or width of the moving average window. A breakdown of the equations involved at each step is shown in Appendix Section A.3.2.

In ARIMA, the future value of a variable is a linear combination of past values and past errors [69, 17], and has been represented by the equation:

$$Y_t = \sum_{i=1}^{p} \phi_i Y_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} \tag{3.3}$$

where Y_t is the actual value at time t, ε_t is the random error at time t, ϕ_i and θ_j are coefficients, and p and q are integers representing the autoregressive and moving average components, respectively [1].

3.8 Error Metrics

Three error metrics were utilized: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Squared Error (MSE). Combinations of these error metrics were found in a variety of papers [29, 93, 23].

RMSE measures the average magnitude of the errors between predicted, $\hat{y}_i$, and actual values, y_i, where n is the number of observations:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.4}$$

MAE measures the average absolute difference between predicted and actual values:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.5}$$

MSE measures the average squared difference between predicted and actual values:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.6}$$

Employing multiple error metrics enabled a thorough evaluation of model performance in stock price prediction. Although RMSE prioritizes accuracy, it can be influenced by outliers. In contrast, MAE provides a balanced assessment, but may have underestimate the impact of larger errors. MSE offers detailed insights, but may have lack intuitive interpretability and be sensitive to outliers. By combining all three metrics, a comprehensive evaluation covering accuracy, precision, and robustness to outliers was achieved, enhancing decision-making in stock market forecasting.

3.9 Hyper-parameter Tuning

The ARIMA parameters underwent optimization using the `auto_arima()` function. This process involved conducting the Augmented Dickey-Fuller test to determine the order of differencing, denoted as d. Subsequently, models were fitted within specified ranges: 0–3 for p and q [78]. Meanwhile, the LSTM model utilized Bayesian Optimization, as elaborated in the following section.

3.9.1 Bayesian Optimization

Bayesian optimization is a powerful technique for optimizing complex, black-box functions commonly encountered in deep learning tasks, offering a principled solution to challenges such as high-dimensional spaces and noisy data. Although its application for hyperparameter tuning in stock analysis is limited, its presence is increasingly noticeable in financial time series data, including indexes [71, 43, 3].

At its core, Bayesian optimization leverages Bayes' theorem, defined in Appendix Section A.4. This theorem is utilized to iteratively update a probabilistic model of the objective function based on observed data, to minimize loss or maximize accuracy. This method employs a Gaussian process to model the posterior distribution of the objective function, enabling efficient estimation of uncertainty ($\sigma(x)$) and prediction of function values ($\mu(x)$) at unobserved points. The key steps include selecting the next point for evaluation using an acquisition function, such as that proposed by Lam et al.:

$$a_t(x) = \int_{-\infty}^{y_{\text{optimal}}} N(y|\mu(x), \sigma(x))dy \tag{3.7}$$

which balances exploration of uncertain regions with exploitation of promising areas [57]. Bayesian optimization efficiently tunes hyperparameters for deep learning models by iteratively evaluating, updating, and selecting configurations until reaching a stopping criterion. It treats the training process as a black-box function, mapping it to an objective metric to explore hyperparameter settings and converge towards an optimal solution in high-dimensional parameter space.

Bayesian Optimization was implemented for hyperparameter tuning of epochs, batch size, number of layers, and learning rate in LSTM models. Initially, dropout was included, but resulted in deteriorated output, prompting its removal. The optimization function prioritized minimizing negative MSE. Activation functions were set to ReLU and the optimizer to Adam.

Bayesian optimization has not yet been extensively applied to hyperparameter tuning for stock data; hence, there were no comparable grid searches to reference. Consequently, various combinations of hyperparameter tunings from the literature were incorporated. For instance, epochs ranged from 10 to 100 as suggested in [10]. Moreover, the batch size ranged from 16 to 128, and the learning rate varied from 0.0001 to 0.01, mirroring the application of Bayesian optimization for the stock index data in [71]). The grid search was adjusted for the type of dataframe tested, with the main variable that changed being the number of LSTM units. The best-performing configurations were determined to be: a) between 10 and 50 LSTM units for datasets with one column b) between 10 and 100 LSTM units for datasets with fewer than 10 columns, and c) between 10 and 200 LSTM Units for datasets between 10 and 150 columns. Both single-layer and 2-layer LSTM architectures were considered.

Parameter	**Range**
Epochs	(10, 100)
Batch Size	(16, 128)
LSTM Units for Layer 1	(10, 200)
LSTM Units for Layer 2	(10, 200)
Learning Rate	(0.0001, 0.01)

Table 3.4: Grid for Bayesian Optimization Hyperparameter Tuning

Chapter 4

Results and Discussion

Experiments were conducted on the validation data and only the Overall Section (Section 4.4) used test data. Following the literature review's guidance, experiments proceeded in a sequenced pipeline: first, testing for the optimal model: either ARIMA or LSTM; then, evaluating the preferred data preprocessing method, which was wavelet transformation; and finally, conducting optimal feature selection. Feature selection was done both manually and automatically. Manual analysis compared indicator combinations. The objective was to identify the best inputs and to compare them with automatic feature selection methods: Correlation (C), Mutual Information (MI), and RFE. This evaluation aimed to assess the effectiveness of automatic feature selection and determine if any method—and if so which—could produce the most optimal model. Hyperparameters will not be further analyzed in this paper, focusing instead on models and inputs. Therefore Bayesian optimization was chosen to iterate through the grid and identify optimal choices for LSTM models, while `auto_arima()` was used to optimize all ARIMA models. The sequence of models examined has been outlined in Table 3.1.

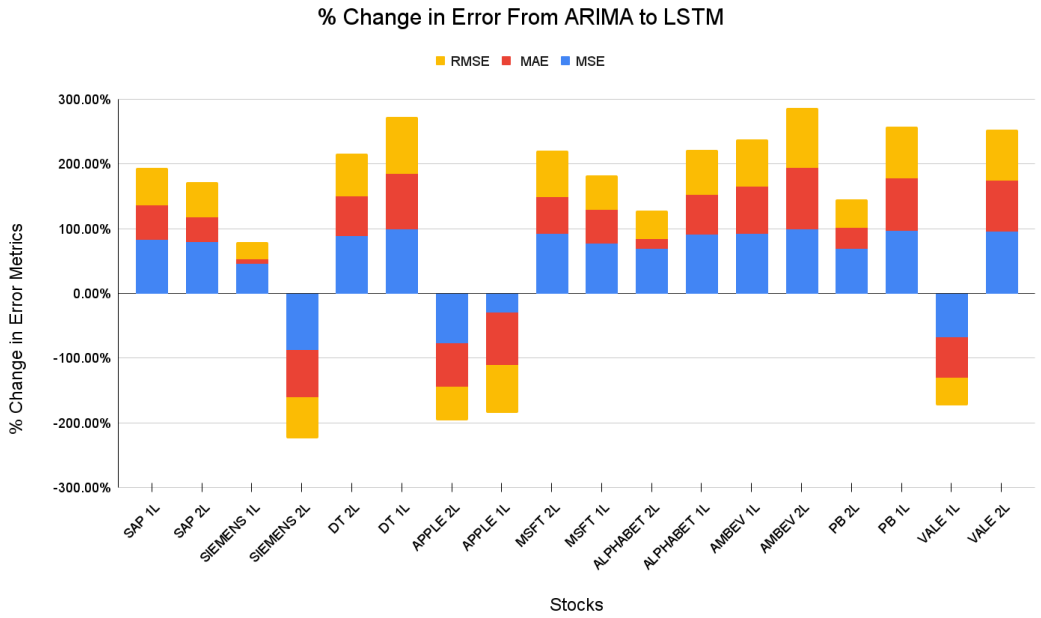
4.1 ARIMA vs LSTM

To compare ARIMA and LSTM models, only the closing prices were used as inputs. Denoised and non-denoised trials were run separately for each model: ARIMA, single layer and 2-layer LSTM, and the trial that resulted in the lowest error metric of each model was selected for each stock. Therefore the optimal model for ARIMA, single-layer and 2-Layer LSTM for each stock was independent of denoising. This choice was made to analyze the models, as the impact of denoising has been assessed in Section 4.2.

As depicted in Figure 4.1, where 1L refers to single layer LSTM and 2L to 2-layer LSTM, LSTM generally outperformed ARIMA regardless of LSTM architecture (single layer or 2-layer LSTM) and of the preference of each optimal model for denoised or non-denoised closing prices, for all stocks except Apple.

ARIMA outperformed LSTM architecture in only 22% of cases. Among these, 50% involved Siemens and Vale, where the selected LSTM architecture underperformed

Figure 4.1:
Decrease in Error Represented by Positive % and Vice Versa



its counterpart. Thus, both single-layer and two-layer LSTMs outperformed ARIMA 77% of the time. Single-layer LSTMs showed positive percentage changes in MSE ranging from 45% to 92%, and in RMSE ranging from 26% to 72% for Siemens and Ambev for both metrics, respectively. As illustrated in Table A.24, 2-layer LSTM models exhibited percentage decreases in MSE ranging from 77% to 99%, and RMSE percentage decreases ranged from 52% to 92% for Microsoft and Ambev for both metrics, respectively.

These results surpassed Zhang's comparison of ARIMA and LSTM models in predicting the closing price of international indexes from 2015 to 2022. Zhang reported percentage decrease in MSE from ARIMA to LSTM ranging from 28% to 68%, and percentage decrease in RMSE ranging from 14% to 44%. While Zhang's study did not specify the LSTM architecture used, this study consistently showed larger percentage increases for both LSTM architectures, suggesting LSTM as more favorable over ARIMA. The discrepancy in the results could be attributed to Zhang's shorter timeframe and the complexity of index data compared to stock-specific data used in this study.

Ambev depicted the largest decrease in MSE between ARIMA and both LSTM layers, suggesting that its closing price was complex and not adequately represented through ARIMA's linear structure. This phenomenon could have stemmed from Ambev's status as a subsidiary within a larger corporate structure—AB InBev, a Belgian beverage and brewing company—distinct from the other eight stocks, which represented parent organizations. Thus, the oversimplification of data through ARIMA's linear characteristics could have overlooked vital information regarding Ambev's parent organization. The correlation between parents and subsidiaries is a developing field. Aufaristama's study

of the relationship of Indonesian companies established a correlation between parent and subsidiary companies[52]. Although variations depend on the country and company, it provided a framework for understanding similar dynamics in other regions, such as Brazil. Even though Aufaristama's findings may not have directly applied to Brazilian stock behavior, they offer insights that could be extrapolated.

Apple stood out as the sole stock wherein ARIMA surpassed both LSTM architectures, leading to a significant increase of 30% in MSE, 81% in MAE, and 74% in RMSE compared to the optimal LSTM architecture, which was a two-layered model. Given its status as the largest market capitalization stock globally at the time of the study, it was anticipated that Apple's closing price would be the most intricate, reflecting numerous macroeconomic variables and global market performance statistics due to EHM dynamics. However, owing to its position as the largest market capitalization stock globally, Apple experienced the least volatility among all technology stocks, which held the most power over the market during the timeframe of this study (see A.21). Due to this reduced volatility, Apple's data aligned better with the stationarity assumptions of ARIMA, resulting in better performance compared to LSTM, which might have overly complicated the modeling process.

In summary, as depicted in A.23, these results suggested that LSTM effectively captured intricate patterns within short-term stock market data better than ARIMA. This contrast between ARIMA's superiority for Apple and LSTM's overall enhanced performance, highlighted the complexity of stock market dynamics, and stressed the importance of assessing the unique characteristics of individual stocks when selecting forecasting models. In conclusion, LSTM's structure was better suited for stock data, given the volatility of closing prices influenced by multiple parameters through EHM.

4.2 Denoised vs Non-Denoised Price

To evaluate the efficacy of the Haar wavelet on ARIMA and LSTM models, separate comparisons were conducted for each model type using only the closing price as input, with and without denoised price data.

In Table 4.1, the outcome of employing a Haar wavelet to denoise the closing prices within the ARIMA model have been presented. For all cases except Ambev, the application of denoised prices led to reduced errors compared to non-denoised prices. The decreases in MSE ranged from 98% for Siemens to 30% for SAP, while in seven out of the total stocks analyzed, MSE decreased by over 60%. In addition, decreases in MAE ranged from 37% to 90% and in RMSE from 17% to 85% for SAP and Siemens, in both error metrics, respectively. These variations might have stemmed from the inherent volatility of the stocks, potentially exerting a more pronounced effect on more volatile stocks. However, 86% of the error metrics which were measured for all stocks except for Ambev, showcased a decrease by over 50%, underscoring the considerable effectiveness of denoising in capturing the chaotic nature of stock data. This illustrated that denoising simplified the data to better fit ARIMA's stationarity assumptions.

These results slightly underperformed Wang and Guo's prediction of the closing price

of ten Chinese stocks when comparing the predictive performance of ARIMA with wavelets and XGBoost implementation [99]. They observed a consistent decrease in MAE and in RMSE errors of 99% for all ten stocks. The discrepancy in their findings, compared to the range observed in this study, could be attributed to the incorporation of XGBoost to reinforce the impact of wavelet transformation. For a detailed breakdown of the application of Haar wavelets for each stock within the ARIMA model, refer to Tables A.17, A.18 and A.25 to A.33.

Table 4.1: % Change in Error From Non-Denoised to Denoised: ARIMA
Decrease in Error Signified Increase in Error Metrics and Vice Versa

	SAP	Siemens	DT	APPLE	MSFT	ALPHABET	AMBEV	PB	VALE
MSE % Change (%)	30.00	98.00	68.00	74.00	78.00	87.00	-57.00	67.00	90.00
MAE % Change (%)	37.00	90.00	61.00	64.00	67.00	75.00	-3.00	63.00	72.00
RMSE % Change (%)	17.00	85.00	44.00	49.00	53.00	64.00	-35.00	43.00	55.00

Only Ambev showed decreased accuracy with denoised prices in ARIMA, with MSE increasing by 57%. It experienced the largest percentage increase when compared between ARIMA and LSTM, reaffirming the complexity of its data as a subsidiary. This suggested that ARIMA's linear characteristics, combined with denoising, might have removed information about its parent organization embedded into the closing price through EHM, which the non-denoised data maintained experiencing better performance.

Table 4.2: % Change in Error From Non-Denoised to Denoised: LSTM
Decrease in Error Signified Increase in Error Metrics and Vice Versa

	SAP		SIEMENS		DT		APPLE		MSFT		ALPHABET		AMBEV		PB		VALE	
	1-L	2-L	1-L	2-L	1-L	2-L	1-L	2-L	1-L	2-L	1-L	2-L	1-L	2-L	1-L	2-L	1-L	2-L
MSE % Change (%)	96	78	98	-35	-1	82	97	-2801	89	-398	57	22	-1739	88	-192	91	-2	68
MAE % Change (%)	86	56	90	-36	-19	62	82	-15	66	-162	36	22	-364	72	-114	81	-3	56
RMSE % Change (%)	80	53	88	-16	-1	58	82	-67	67	-123	35	12	-329	65	-71	69	-1	44

In Table 4.2, the outcomes obtained from denoising the LSTM models, which solely relied on the close price as its input, have been illustrated. Previous literature evaluated the application of wavelets in LSTM, but has not compared the effectiveness of denoising based on the model's architecture. Thus, the impact of denoising on single layer and double layer LSTMs was examined. Overall, denoised prices exhibited superior performance compared to the non-denoised prices, with 8 out of 9 stocks preferring denoised prices, except for Apple. Stocks were classified into three categories: those where denoising proved more effective for both single-layer and 2-layer LSTMs, those where denoising was more beneficial for single-layer LSTMs but not for 2-layer LSTMs, and those where denoising was optimal for 2-layer LSTMs while non-denoised prices were favored for single-layer LSTMs. In none of the stocks, non-denoised prices outperformed denoised for both single layer and 2-layer LSTMs. For all stocks but Apple, denoised prices overall were preferred to non-denoised as it experienced a 2800% increase in MSE for the 2-layer LSTM.

In the first category, denoised models demonstrated superior performance for both single and 2-layer LSTMs, as evidenced by SAP and Alphabet. These companies,

both technology-oriented conglomerates headquartered in Europe and North America, respectively, exhibited a preference for single-layer LSTMs, with SAP experiencing an 18% decrease in MSE and Alphabet a 35% decrease when transitioning from a single layer to a 2-layer LSTM. This highlighted the increased impact of denoising on simpler structures. This trend persisted for the second category which included the other North American and European technology stocks such as Siemens, Microsoft, and Apple. These three companies showed a decrease in error when using Haar wavelets for single-layer LSTMs, while experiencing an increase in error when applied to 2-layer LSTMs. Although Apple generally favored non-denoised prices, its denoised prices exhibited preference towards the single-layer LSTM.

A potential reason for the preference of denoising in single-layer LSTM could have been the relatively lower volatility experienced by technology stocks compared to industries producing material goods. S&P's analysis of volatility trends from 2010 to 2020 indicated that technology firms encountered lower volatility levels when contrasted with stocks from industries involved in material goods production [56]. Therefore, single-layer LSTMs benefited from denoising because of their simpler architecture. In contrast, 2-layer LSTMs, with a deeper architecture enabling them to capture both low-level features and higher-level abstractions, were more robust in modeling non-denoised technology stock data, as the data was less volatile compared to other industries. Consequently, denoising resulted in a loss of information for 2-layer LSTMs, leading to increases in MSE ranging from 2800% to 35% for Apple and Siemens, respectively.

This argumentation could expand to the third category, which was characterized by the model performing better for a 2-layer LSTM than a single layer for denoised prices. This was true for all the commodity and infrastructure companies examined: DT, a telecom company; Ambev, a beverage producer; Vale, a mining corporation; and PB, an oil giant, all of which were more volatile due to their industry compared to technology companies according to [56]. Hence, a denoised single-layer LSTM might have oversimplified the diversity of parameters influencing the close price, as explained in the EMH theory and exhibited with negative MSE percentage changes for the stocks in this category. Thus, the complexity of 2-layer LSTM could have better captured the diverse influencing parameters present in the denoised close price, with decreases in MSE ranging from 91% to 68% from PB and Vale, respectively. The variance could be due to the volatility of each industry as the S&P 500 ranked oil the most volatile [56]. This consensus aligned with Liang et al.'s analysis of the impact LSTM layers have on denoised prices for the S&P stock index, which found 2-Layer LSTMs optimal; although it was an index, the results were comparable as the diversity of sectors in the S&P 500 reflected the industries in this category [49]. Finally, in-depth analysis of the impact of Haar wavelets on LSTM for each stock have been shown in Appendix Section A.5.2.3.

Ambev experienced a roughly 1700% increase in MSE with single-layer LSTM denoising, marking its weakest denoising performance. This increase mirrored Ambev's underperformance with denoised ARIMA, suggesting denoising might have erased crucial parent organization data from closing prices in simpler structures. However, 2-layer LSTMs, capturing both high and low-level details, could have discerned this information, resulting in an 88% reduction in MSE through denoising, thereby enhancing accuracy.

Apple was the only stock which preferred non-denoised prices overall for LSTM. Although it experienced a 97% decrease in MSE with the application of denoising for single-layer LSTM, it endured an approximately 3000% increase in MSE with the application of Haar wavelets for 2-layer LSTM, which was the most effective architecture overall with non-denoised prices. This highlighted Apple's preference for ARIMA over LSTM. Given Apple's limited volatility, denoising removed excessive information, considering its already straightforward data. Hence, 2-Layer LSTM managed Apple's data without denoising, yielding better results, while single-layer LSTM benefited from denoising due to its simpler structure. Similarly, the ARIMA model saw a 74% decrease in MSE with denoising. Nonetheless, ARIMA consistently outperformed LSTM as Apple's limited volatility might have better suited ARIMA's stationary assumptions than LSTM's complex pattern-capturing mechanisms. Acknowledging existing literature advocated for diverse variables, Apple underwent testing with denoised and non-denoised prices in multi-indicator LSTMs.

Ultimately, Haar wavelets proved more successful in both LSTM and ARIMA, with 89% of stocks benefiting through denoising in both models as showcased in A.23. Denoising showed greater effectiveness in ARIMA than in LSTM, with a narrower range of error metrics (-57% to 98% in MSE for ARIMA, compared to -2801% to 98% in LSTM). This highlighted denoising's ability to complement the stationarity assumptions of ARIMA while LSTM was adept at handling more complex data. Thus, the successful application of Haar wavelets on LSTM depended on the industry and the layers in the model.

4.3 Feature Selection

The indicator combinations were tested for LSTM, as ARIMA accepts one variable; they were optimized using Bayesian Optimization. This sequence of analysis in this section aligned with the pipeline outlined in the literature review for optimal feature selection. Each subsection analyzed the application of specific indicator groups or feature selection methods across stocks from various industries and geographies. The discussion within each subsection was independent of the final optimal inputs and feature selection method, which is discussed in the overall section.

4.3.1 Manual Feature Selection: Indicator Analysis

4.3.1.1 Basic vs. Advanced Technical Indicators

Numerous studies have highlighted the effectiveness of advanced technical indicators in predicting short-term stock movements [81]. Thus, basic technical indicators served as a baseline in this analysis, compared to a dataset containing both basic and advanced technical indicators– all technical indicators. Six stocks, namely SAP, Apple, Microsoft, Alphabet, Ambev, and Vale, demonstrated overall superior performance when advanced technical indicators were included, compared to three stocks where basic technical indicators sufficed: Siemens, DT, and PB. Previous literature had not compared the application of basic and all technical indicators across LSTM architecture. For the

performance of each stock on basic and all technical indicators, refer to Section A.5.3.1.

All US stocks, as well as SAP, Ambev, and Vale, exhibited a clear preference for the integration of advanced technical indicators, with decreases in MSE compared to basic indicators ranging from 17% to 87% for Siemens and Vale, respectively. This preference likely could have been attributed to the substantial influence these stocks wielded in global markets, with all US stocks ranking within the top 5 market capitalization stocks globally; Vale is the world's largest producer of iron ore and nickel, and SAP holds the position of the world's leading resource planning software provider. Hence, this underscored the need to employ multiple equations to accurately model the closing price and gain additional insights into its movements and trends.

It should be noted that Apple's denoised and non-denoised prices exhibited a preference for all technical indicators. Moreover, denoised prices outperformed non-denoised prices with a 63% decrease in MSE from non-denoised to denoised because advanced technical indicators relied on the close price, and complexity increased significantly with non-denoised technical indicators. Thus, denoising helped alleviate the chaotic nature of the data, enhancing the performance of advanced technical indicators.

On the contrary, the three stocks that performed better with basic technical indicators—PB, Siemens, and DT—illustrated that complexity did not necessarily equate to better prediction. These firms, while significant players in their respective industries, did not hold the same global influence or complexity as their counterparts who benefited from advanced indicators. Therefore, the introduction of additional indicators to model the closing price might have only served to add noise to the model rather than to improve predictive accuracy with MSE increases ranging from 12% for Siemens to 49% for PB.

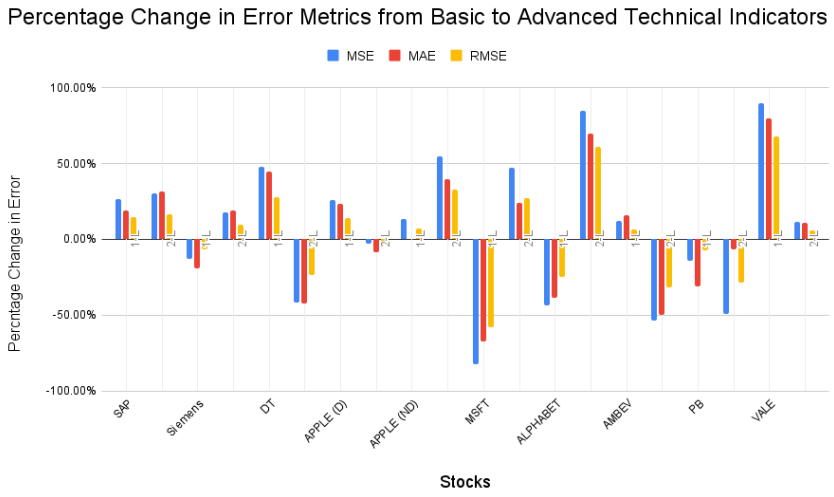
There were once again discrepancies between single and two-layer LSTM models, as illustrated in Table 4.2. Only one model, PB, preferred basic technical indicators for both one and two-layer LSTMs, experiencing a 14.19% increase in MSE when all technical indicators were incorporated for a single-layer LSTM and a 49.28% increase in MSE for the two-layer LSTM. A fairly even split distribution in preference for basic and all technical indicators emerged based on LSTM layers. For single-layer LSTM models, five stocks exhibited a preference for all technical indicators: SAP, DT, Apple, Ambev, and Vale, while four performed better with basic technical indicators: Siemens, Microsoft, Alphabet, and PB. In the case of 2-layer LSTM models, five stocks favored all technical indicators: SAP, Siemens, Microsoft, Alphabet, and Vale, while four preferred basic technical indicators: DT, Apple, Ambev, and PB. This even distribution illustrated that the application of basic or all technical indicators on single layer and 2-Layer LSTM was unique to the characteristics of the stocks and did not follow any patterns.

Overall, the benefits of incorporating advanced technical indicators appeared to be less significant than initially anticipated. This was evidenced by only a 5% average decrease in RMSE for all stocks, in contrast to Mittal and Chauhan's findings, where they reported a 90% decrease in RMSE upon incorporating advanced technical indicators alongside basic ones to predict closing prices of 20 Indian companies [74]. The disparity between these outcomes might have stemmed from the utilization of Haar wavelets in this study, which could have mitigated the effectiveness of advanced technical indicators

in providing insights on the closing price. Although the use of advanced technical indicators led to a reduction in dimensionality, as evidenced by Apple's improved performance with denoised prices compared to non-denoised ones, the overall need for advanced technical indicators could be diminished. This was because denoising enabled the model to accurately interpret the closing price itself, thereby reducing the significance of additional indicators in capturing the different components of EHM reflected in the price. Consequently, the model might have effectively interpreted the closing price itself without the need for supplementary indicators, illustrating some stocks' preference for basic indicators and the low overall increase in accuracy.

However, since the addition of advanced technical indicators was more effective for 67% of the stocks, they were implemented over basic technical indicators in the remainder of the feature selection analysis. Specific considerations for stocks that preferred basic technical indicators were not made because of time constraints. Theoretically, feature selection would also have removed unhelpful variables, minimizing the impact of incorporating advanced technical indicators throughout the study. Finally, a broader range and greater number of stocks were necessary to reveal patterns in the preference of LSTM architecture towards basic and all technical indicators.

Figure 4.2:
Decrease in Error Signified Increase in Error Metrics and Vice Versa



4.3.1.2 Fundamental and Macroeconomic Indicators

To determine the optimal indicator combination, all technical indicators were combined with fundamental and macroeconomic data to test which combination was best suited to the stock, industry, or region. There was limited literature comparing the impact of these indicators individually on stock market prediction. A complete dataset with all technical, fundamental, and macroeconomic indicators was analyzed in Section 4.3.2, while, in this section, it was investigated if a particular indicator group was stronger. While both single-layer and double-layer LSTM models were tested, the optimal model from the two was selected for analysis. Incidentally, 89% of stocks preferred the 2-layer

LSTM, illustrating that an increase in the number of indicators favored a larger model. The remainder of this section's analysis was based on the optimal LSTM architecture per stock to analyze indicator preferences.

Primarily, all stocks except one—PB—performed better on a dataset comprising both technical and fundamental or macroeconomic data as compared to using just advanced technical indicators. As illustrated in Table 4.3, most stocks experienced an increase in performance with the incorporation of additional indicators. This outcome aligned with expectations, as the addition of diverse indicators increased performance as explained in the Literature Review Section 2.1. For a detailed breakdown of the performance of each stock per indicator dataset, refer to Appendix A.5.3.2.

Table 4.3: % Change in Error From Advanced Technical to Fundamental (F) & Macroeconomic (M): Decrease in Error Signified Increase in Error Metrics and Vice Versa

	SAP		SIEMENS		DT		APPLE		MSFT		ALPHABET		AMBEV		PB		VALE	
	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M
MSE % Change (%)	4.5	8.4	4.3	17.4	10.2	7.1	15.2	-10.7	33.4	33.3	17.3	2.4	-18.5	0.3	-57.8	-47.8	7.3	-13.1
MAE % Change (%)	2.1	1.0	10.1	5.7	6.5	1.1	10.0	-9.7	32.8	29.6	14.3	0.6	-25.0	3.8	-40.0	-26.4	2.5	-25.5
RMSE % Change (%)	2.3	4.3	2.1	9.1	5.2	3.6	7.9	-5.5	18.4	18.4	9.1	1.2	-9.7	0.1	-35.0	-27.7	-3.7	-6.8

As mentioned in Section 4.3.1.1, PB stood out as the only stock that favored basic indicators over all technical indicators in both LSTM architectures. This preference has persisted throughout this analysis, as the inclusion of fundamental indicators resulted in a 57.78% increase in MSE, while the incorporation of macroeconomic indicators increased MSE by 47.77%. Although there appeared to be a slight preference for macroeconomic indicators due to their smaller increase in error, PB's inclination towards basic technical indicators underscored the typical structure within the oil industry. This aligned with Firouzjaee and Khalilian findings that suggested that incorporating commodity prices had limited benefits in assisting the prediction of oil companies [42]. Therefore, while PB was impacted by the prior decision to incorporate advanced technical indicators in this study, this was mitigated by the fact that neither fundamental nor macroeconomic variables were essential for accurately predicting oil companies.

77% of stocks performed better when fundamental indicators were incorporated alongside all technical indicators. The value of fundamental indicators was supported by Beyaz et al., who conducted tests on 140 companies from the S&P 500. They concluded that datasets combining fundamental and technical indicators resulted in a lower RMSE in over 95% of cases compared to using fundamental or technical indicators alone [21]. The difference between these studies may be rooted in the number of stocks tested.

Five stocks, namely DT, both denoised and non-denoised Apple, Microsoft, Alphabet, and Vale, exhibited a preference for fundamental indicators. These stocks, recognized as industry leaders, showcased decreases in MSE spanning from 7.3% for Vale to 33.47% for Microsoft. While macroeconomic indicators such as consumer demand and interest rates were expected to significantly influence these stocks by impacting their revenue streams, operational costs, and investor sentiment in the global market, fundamental indicators took precedence. This observation highlighted that these giants shape the market rather than passively reacting to it. Factors such as market dominance, innovation

leadership, and regulatory influence enabled these corporations to establish industry standards, prices, and trends, all reflected in fundamental indicators. Consequently, these indicators, depicting strong performance metrics like Return on Equity and Return on Assets, served as magnets for investors and could predict price increases. The pivotal role of financial ratios in forecasting stock prices was emphasized by Fatimah and Lubis [83].

Within this subset of stocks, only Apple and Vale demonstrated an increase in error when macroeconomic indicators were included. This illustrated the critical nature of feature selection and the necessity of testing an indicator's impact on a stock, rather than presuming its benefits, as much of the previous literature has done. Furthermore Apple's denoised price consistently outperformed its non-denoised counterpart for all indicator combinations as presented in Tables A.56 and A.57. Thus, the preference for non-denoised closing prices may have been only applicable to the sole-indicator LSTM.

In this analysis, macroeconomic indicators were favored by companies that are not global leaders but rather respond to the market. This trend is observed in companies like SAP, Siemens, Ambev, and PB. This suggests that these stocks, while influential within their respective countries, have less industry-wide impact compared to giants like Apple, Microsoft, and Alphabet. While prominent in their domains, they lack the same level of industry dominance. For instance, SAP dominates resource planning software but lacks the overarching influence seen in the technology sector. Their decrease in MSE ranged from 0.25% to 17.42%, which is less pronounced than that observed among firms favoring fundamental indicators. This might be because these firms were still part of the industry leaders; therefore, their reaction to the market may have been reduced. Additionally, the range of MSE decrease fell below Lakshminarayanan and McCrae's comparison of advanced technical indicators dataset with a dataset involving oil prices, which experienced a 27% decrease in MSE when predicting the closing price of the DJI index with LSTM and incorporating moving averages [72]. This difference may have stemmed from variations in the data between individual stocks and indexes, as well as differences in the model structure. Specifically, Lakshminarayanan and McCrae utilized moving averages for temporal smoothing, whereas this study focuses on denoising implementation. Moving averages smooth data by averaging consecutive data points within a time window, effectively captured trends and reduced noise. In contrast, Haar wavelets decomposed the data into different frequency components, allowing for the detection of abrupt changes and edges in the data, potentially offering a more nuanced representation for LSTM networks. Consequently, the effectiveness of Haar wavelets in denoising may have reduced the necessity of macroeconomic indicators, much like advanced technical indicators, to provide additional information about the closing price, as the model can interpret the close price directly. Moreover, Lakshminarayanan and McCrae only incorporated oil prices, hypothesizing that the broader range of macroeconomic indicators in this analysis might introduce complexities in the data. This reinforced the need for feature selection to extract the most prominent features.

In summary, the addition of fundamental or macroeconomic indicators, alongside all technical indicators, increased accuracy for all but one stock: PB. These datasets were compared against all technical indicators, which were utilized as a baseline, as prior literature supports an increase in accuracy with the incorporation of diverse features.

The preference for either macroeconomic or fundamental indicators appeared to be more closely tied to the global impact of a particular company on its industry rather than industry or location. This observation emerged as a consequence of selecting the largest market capitalization stocks per country, revealing an implicit global hierarchy of stocks. Companies higher on this hierarchy tended to be more influenced by fundamental indicators, reflecting their significant impact on a global scale. Conversely, companies with a more regional focus exhibited a greater influence from market conditions, thereby favoring macroeconomic indicators.

4.3.1.3 Exchange Rates and Commodities

Macroeconomic indicators experienced significant disparities in their forward filling rates, as illustrated in Tables A.14 and A.15. To assess the impact of forward-filling on the data, exchange rates, representing the least forward-filled macroeconomic indicators common to all stocks at 24.77%, and commodities, representing the most forward-filled macroeconomic indicators common to all stocks at 97.36%, were individually tested. The recommendation of forward filling over removing missing data was suggested by Diamond et al., however, their data gaps were less than 10%, raising questions about extensive use of forward-filling [59]. As the focus of this section was on the impact of forward-filling, the LSTM architecture converging to the least error was utilized.

Table 4.4: % Change in Error From Macroeconomic to Exchange Rates (ER) & Commodities (C) : Decrease in Error Siginified Increase in Error Metrics and Vice Versa

	SAP		SIEMENS		DT		APPLE		MSFT		ALPHABET		AMBEV		PB		VALE	
	ER	C	ER	C	ER	C	ER	C	ER	C	ER	C	ER	C	ER	C	ER	C
MSE % Change (%)	-24.8	-19.6	-12.1	-100.7	-55.2	-47.2	-21.0	-23.6	-107.1	-95.9	-35.5	-74.5	-84.8	-35.8	-23.1	8.4	17.9	14.2
MAE % Change (%)	-21.9	-19.4	3.0	-69.6	-50.5	-24.4	-19.4	-16.4	-77.9	-55.6	-33.9	-48.1	-80.2	-35.3	-8.4	9.9	24.0	15.7
RMSE % Change (%)	-11.7	-9.4	-5.9	-41.7	-24.6	-21.4	-1.1	-11.2	-43.9	-39.3	-16.3	-32.2	-36.0	-16.5	-11.0	4.3	9.2	-7.4

Only two stocks exhibited superior performance with commodities or exchange rate datasets compared to macroeconomic indicators. This suggested that a diverse range of indicators was preferable over specialized ones, with increases in MSE ranging from 12.1% for Siemens to 107% for Microsoft with exchange rate data. This finding aligned with existing literature, which typically incorporated various macroeconomic indicators to fully exploit the informational variability that encompassed the dynamics of the global market. However, PB and Vale illustrated a decrease in error metrics, with PB experiencing a decrease in MSE by 8.4% for the commodities dataset and Vale being the only stock to decrease in MSE for both exchange rates and commodities datasets by 17.9% and 14.2%, respectively. PB was the only stock to be negatively influenced by both fundamental and macroeconomic indicators, echoing the oil industry's preference for simplicity. Thus, consistent with Firouzjaee and Khalilian [42], the incorporation of commodities into the all technical indicator dataset reduced MSE by only 11%, compared to the 58% increase in MSE from all macroeconomic indicators. This reinforced the argument from Section 4.3.1.2 that purposeful feature selection for macroeconomic indicators was necessary. Similarly, Vale experienced an increase in MSE of 13% when applying all macroeconomic indicators to the advanced technical dataset, but a decrease of 17.9% and 14.2% for exchange rates and commodity datasets,

respectively. This prompted the hypothesis that firms offering primary products—oil and minerals—were most influenced by exchange rates and commodities, which defined their markets. Hence, purposeful feature selection according to industry was necessary, as all macroeconomic indicators may have unnecessarily added complexity and redundancy to primary products, while a wide range was beneficial for tertiary services.

Secondly, forward-filling rates did not significantly affect prediction accuracy. Commodities -based datasets demonstrated lower error metrics than the exchange rates dataset for five stocks: PB, Ambev, Microsoft, DT, and SAP; this represented the best performing model from both single and 2-layer LSTMs. Exchange rates were favored for the remaining four stocks: Alphabet, Apple, Siemens, and Vale, with Apple preferring exchange rate datasets for both denoised and non-denoised datasets. The division of stocks based on exchange rates and commodities stemmed from the companies' structural organization, with general divisions by industry and country not being applicable. The balanced preference and sub-optimal performance compared to fundamental and macroeconomic datasets suggested that forward-filled data remained a viable method for preparing data for stock price prediction. For a granular analysis of the performance of each stock for both of these datasets, refer to Section A.5.3.3.

4.3.2 Automatic Feature Selection Combinations

Three feature selection approaches had then been compared: including all technical, fundamental, and macroeconomic indicators (no feature selection), feature selection using correlation and mutual information removal (C+MI), and feature selection using correlation and mutual information reduction RFE (C+MI+RFE).The optimal LSTM structure was selected and the number of features selected per stock was determined according to Table 3.3. Table 4.4 indicated a divergence in the preference for feature selection methods: three stocks showed better performance with no feature selection: SAP, Apple, and Ambev, while the remaining six: Siemens, DT, Microsoft, Alphabet, PB, and Vale, exhibited the best results with C+MI+RFE. For a breakdown of the performance of each stock under the three feature selection combinations, refer to A.73 to A.82.

The overall preference towards feature selection was expected and aligned with previous literature as the incorporation of fundamental and macroeconomic indicators to advanced technical indicators led to an overall decrease in error metrics (Table 4.3). Feature selection reduced the dataset size while retaining the most important variables. The benefits were inherent for companies where applied fundamental and macroeconomic indicators increased accuracy: Siemens, DT, Microsoft, and Alphabet, as the best positive features were selected to reduce complexity and noise. However, for PB and Vale, which experienced poorer accuracy due to the incorporation of either dataset, feature selection addressed this problem by choosing variables closer to the optimal set. PB's preference for C+MI+RFE reduced the dataset to 55 values (as shown in A.72), significantly closer to the 5 basic technical indicators that yielded optimal performance. Similarly, Vale experienced a decrease in accuracy with the incorporation of macroeconomic indicators, with a 13.14% increase in MSE; however, C+MI+RFE reduced the presence of macroeconomic indicators from 80% in the macroeconomic-only dataset

to 70% (as seen in A.91), with 20% of the remaining macroeconomic indicators being exchange rates which enhanced Vale's predictive accuracy. This confirmed that feature selection could be effective in extracting the optimal combination of features. The optimal LSTM structure was evenly split among the six stocks that performed better with feature selection. This depended on the optimal number of features preferred by the stock in RFE, with fewer variables preferring single-layer LSTMs, and larger datasets favoring 2-layer LSTMs.

Table 4.5: % Change in Error From No Feature Selection To Feature Selection C+MI (1) and C+MI+RFE (2): Decrease in Error Siginified Increase in Error Metrics and Vice Versa

	SAP		SIEMENS		DT		APPLE		MSFT		ALPHABET		AMBEV		PB		VALE	
	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
MSE % Change (%)	-29	-1	-6	20	4	17	-23	-5	-22	14	29	31	-14	-15	11	18	-11	4
MAE % Change (%)	-24	1	-5	19	5	21	-99	-99	-15	20	17	22	-15	-12	4	13	-10	5
RMSE % Change(%)	-16	-1	-3	11	2	9	-99	-99	-12	7	16	17	-7	-8	1	3	-5	2

SAP, Apple, and Ambev demonstrated superior performance without employing feature selection, experiencing an increase in MSE from 1% for C+MI+RFE for SAP to 23% for C+MI for Apple. These stocks belonged to different industries and geographies, suggesting that their unique characteristics played a crucial role in their performance. Apple and Ambev consistently stood out in this study. It was possible that the denoised Apple prices, with feature selection, omitted information that the LSTM could handle, given Apple's preference for non-denoised prices in LSTM due to its limited volatility. Similarly, Ambev's preference for no feature selection may have stemmed from its need to incorporate a broad set of indicators to assess the health of its international parent company. It should be noted that all stocks that preferred no feature selection were optimal on the 2-layer LSTM suggesting that the increased complexity of the model could handle the task of modeling various parameters, particularly as the closing price was denoised and the complexity of advanced technical indicators was reduced (recall Section 4.3.1.1).

Despite the absence of feature selection, both stocks underperformed their optimal indicator combination. The feature selection methods which were tested might have removed indicators optimal for these specific stocks, such as fundamental indicators for Apple and macroeconomic indicators for Ambev. RFE exhibited a preference toward stocks with fewer forward-filled indicators as the number of features decreased, aiming to capture complexity from a reduced amount of data points. As illustrated in Table A.86, with an optimal RFE percentage of 25% for Apple data, the application of RFE removed fundamental indicators since they were highly forward-filled, which constituted the only positive indicator group for Apple, thereby increasing MSE by 5%. Similarly, for Ambev, RFE removed highly forward-filled macroeconomic data, as shown in Table A.85, with the RFE dataset focusing more on exchange rates rather than commodities. This contradicted Ambev's preference for commodities over exchange rates, as indicated in Section A.5.3.3. This novel analysis scrutinized the impact of RFE and prompted further investigation into the application of forward-filled data with RFE.

The findings from Table 4.5 illustrated that the combination of C+MI+RFE outper-

formed C+MI, even for stocks that were originally optimal without feature selection. Across all stocks considered, implementing C+MI+RFE resulted in MSE reductions ranging from 4% for Vale to 31% for Alphabet. Conversely, using C+MI alone led to MSE increases ranging from 22% for Microsoft to decreases of 27% for Alphabet within the same subset of stocks. These results aligned with the analysis by Tsai and Hsiao, suggesting that combining feature selection methods enhances accuracy by eliminating a broader range of irrelevant features [14]. However, it was noteworthy that the decrease in accuracy observed with C+MI alone had not been previously documented. This discrepancy could be attributed to the overlap in information removal since both methods were filtering techniques, potentially reducing the range of elements removed due to their similar structures. According to the data presented in Table A.72, employing the combination of C+MI reduced the dataset size by approximately 15% on average, with MI further reducing it by an additional 10 stocks on average, resulting in a total decrease of 22%. While this reduction was significant, it might not have been substantial enough, suggesting a preference for either no feature selection or the addition of RFE.

Furthermore, Tables A.76 and A.77 highlighted that Apple's non-denoised closing price exhibited the lowest error metrics with the C+MI approach, whereas its denoised counterpart performed better with no feature selection. Employing the C+MI approach could have proved advantageous for non-denoised prices, as even minor reductions in dataset complexity could yield benefits. Thus, using C+MI for denoised data might have resulted in information loss, as the number of variables removed would not be significant enough to enhance performance, as seen in non-denoised models. This underscored the necessity for further exploration into how denoising could affect feature selection methods.

Table 4.6: % Change in Error From Optimal Indicator Combination to Optimal Level of Feature Selection: Decrease in Error Siginified Increase in Error Metrics and Vice Versa

	SAP	Siemens	DT	Apple	Microsoft	Alphabet	Ambev	PB	Vale	
MSE % Change (%)	-5.46	-2.03	-35.39	-7.00	-22.31	-20.83	0.60	-10.50	-5.11	-12.00
MAE % Change (%)	-5.86	-7.29	-30.54	-13.84	-14.14	-21.46	-9.05	4.63	-13.84	-12.38
RMSE % Change (%)	-2.69	-1.01	-16.36	-3.56	-10.59	-11.02	0.30	-5.39	-2.52	-5.87

All feature selection models exhibited inferior performance compared to the sole indicator combinations outlined in Section A.5.3.2 for all stocks, as depicted in Table 4.6. The table illustrated a range of MSE percentage increases due to feature selection, ranging from 35% for DT to a 0.6% increase in MSE for Ambev. Although Ambev exhibited a decrease in MSE and RMSE, both remained below 1%. However, there was an overall increase in MSE by 9%, suggesting an overall decline in performance with feature selection. This observation raised concerns about previous research that utilized various datasets without assessing the influence of each specific indicator. Three possible explanations for this issue emerged. Firstly, the strong correlation between a stock and the market might have overshadowed the importance of including diverse metrics in the model. This could have introduced noise that could not be effectively filtered out by any feature selection method, necessitating the exclusion of certain datasets altogether. Alternatively, the current feature selection mechanisms might have been insufficiently

robust for such a large and complex dataset. Finally, since the selected feature selection combination had not been previously used in conjunction with Haar wavelets, it was plausible that Haar wavelet denoising alone sufficed, and the incorporation of a diverse set of indicators distracted the model from processing the closing price, which was robust and encompassed all additional variables due to EMH. Thus, the addition of any indicators worsened the performance of the model, rendering the analysis of feature selection moot.

4.4 Overall

To determine the optimal model, all combinations from Table 3.1 were tested on validation data and with LSTM models optimized using the Bayesian optimization grid and ARIMA models utilizing the `auto_arima()`. The pipeline combination with the smallest error metrics was selected as the optimal model (O), while the best indicator combination (BIC) referred to the optimal combination of indicators per stock encompassing both manual and automatic feature selection. The O and BIC models were run on test data utilizing the parameter turnings from the validation data version of the model for that stock. Contrary to previous literature guidelines, the optimal indicator for all stocks was simply the denoised closing price. All stocks, except for Apple, showed optimal performance with denoised closing prices using LSTM, while Apple favored ARIMA. Meanwhile, as mentioned earlier, the Best Indicator Combinations (BIC) were obtained from the manual feature selection section. PB showed a preference for basic technical indicators, while Apple, Alphabet, Microsoft, and Vale preferred fundamental indicators alongside all technical indicators. On the other hand, Ambev, SAP, and Siemens favored a combination of macroeconomic and all technical indicators. For a complete breakdown of the optimal models and architectures, refer to Table A.92, and for the final results, refer to Tables A.93 and A.94.

Table 4.7: % Change in Error From Optimal Model Overall to Optimal Indicator Combination Overall: Decrease in Error Signified Increase in Error Metrics and Vice Versa

	SAP	Siemens	DT	Apple	Microsoft	Alphabet	Ambev	PB	Vale
MSE % Change (%)	-99.04	-99.66	-13.14	-99.93	-99.38	-69.48	-99.99	-99.25	-84.82
MAE % Change (%)	-89.64	-95.00	-18.38	-97.45	-92.76	-47.59	-99.95	-88.84	-66.39
RMSE % Change (%)	-90.20	-49.89	3242.5	-97.37	-92.15	-44.76	-99.15	91.32	-61.04

The addition of extra indicators to the denoised price resulted in an increase in MSE, ranging from 13.14% for DT to 100% for Ambev. A fairly consistent percentage increase was experienced with the addition of extra indicators, with 89% of stocks seeing an increase above 70%; this underscored the effectiveness of Haar wavelets in decomposing the close price for LSTM and ARIMA models to better process. With denoising, the closing price became decipherable by the model without the need for additional indicators, as they were already encompassed in the close price through EMH. Therefore, the addition of any indicator worsened performance by distracting the model from the close price. This supported the third hypothesis regarding the impact of automatic feature selection compared to manual. Furthermore, the preference for three feature selection techniques over no feature selection and two feature selection

techniques suggested the data's inclination towards simplicity by minimizing the number of variables to reach the closing price itself. Apple's preference for ARIMA may be explained by its lower implied volatility compared to other stocks. This characteristic enabled ARIMA, with its stationarity assumptions and linear characteristics, to perform better, as it is better suited to handle simpler data structures. In contrast, LSTM neural networks excelled in capturing more complex patterns, which did not align as well with Apple's data characteristics. Regarding LSTM architecture preferences, three stocks showed optimal performance with a 1-layer LSTM, while five stocks benefited from a 2-layer LSTM. However, for the BIC models, 2-layer LSTMs prevailed, being preferred by 8 out of 9 stocks. This indicated that the increase in data complexity through additional indicators necessitates stronger models.

Previous research had not extensively compared denoised and non-denoised closing prices or highlighted the potential benefits of using denoised closing prices alone. However, Aminimehr et al. compared the application of PCA, RF, and wavelets in LSTM models to predict the closing prices of the S&P 500 using only basic technical indicators. They found wavelets to be optimal on the validation data, with a decrease in MSE of 11% compared to all remaining methods [2]. The difference in the impact of denoising between the two studies could be attributed to the use of basic technical indicators, which worsened performance, and the utilization of stock index data instead of individual stocks.

Tables A.93 and A.94 consistently showed MSE magnitudes one or two orders of magnitude smaller than MAE and RMSE. This rare phenomenon arose from the synergy between time-series data and LSTM architecture, observed across various stock prediction studies such as [2, 94].LSTM models excelled at capturing sequential patterns, enhancing prediction accuracy over time by effectively learning dependencies within sequences, particularly intricate or long-range temporal dependencies like forward-filled data, and outperforming in minimizing squared differences between predicted and actual values.Consequently, this led to lower MSE scores compared to MAE. Thus, MSE scores were predominantly used as a comparison metric throughout this study. Ambev exhibited the smallest error metrics, with an MSE of 8.3×10^{-7}, MAE of 4.7×10^{-4}, and RMSE of 9.1×10^{-4}. Alphabet experienced the largest error metrics, recording an MSE of 2.1×10^{-3}, MAE of 4.1×10^{-2}, and RMSE of 4.6×10^{-2}. Although, both stocks demonstrated optimal performance on an LSTM model using only denoised closing prices, there existed a stark contrast between the two stocks' performance metrics, with a 200% difference in MSE values, a 95% difference in MAE, and a 199% difference in RMSE. This discrepancy underscored the difficulty in creating a global stock prediction model and the necessity for further exploration to achieve error metrics of comparable magnitudes across different stocks and industries.

Exact comparisons were challenging because of the absence of prior coverage of this specific combination of methods, data ranges, and stocks. However, Butunac et al. achieved a MAE of 0.0147 and MSE of 0.0041 when predicting US technology stocks using RFE with a RF regressor, slightly underperforming the average MAE of 0.0147 and MSE of 0.0041 in this study [33]. This highlighted the robustness of the results.

Chapter 5

Conclusion

The best-performing model was an LSTM equipped with only denoised close prices. While the architecture of the model varied depending on the stock, it was observed that 2-layer LSTMs effectively captured the complexity of the data throughout the analysis. This study reaffirmed the superiority of LSTM models over ARIMA and highlighted the significant reduction in error metrics achieved through denoising, thereby enhancing accuracy and performance. Concerning indicator combinations, the company's status within the industry influenced the preferred indicators. Industry leaders favored fundamental data, which shapes the market, while secondary companies exhibited better performance with macroeconomic indicators, as they responded to market trends. Interestingly, combining fundamental and macroeconomic indicators with advanced technical indicators led to a decrease in performance compared to closing prices alone. This issue persisted despite attempts at feature selection, as the inclusion of three categories of indicators introduced excessive noise and diminished performance. Finally, regarding automatic feature selection methods, the combination of C+MI+RFE outperformed C+MI models; this was because employing multiple feature selection methods reduced the dataset size, facilitating the evaluation of the closing price. However, further research on the impact of forward-filled data with RFE should be examined or performed. Ultimately, denoised close prices emerged as the optimal input, corroborating the suspicions of the EHM, with the inclusion of indicators leading to an average increase in MSE by 85%.

5.1 Limitations

Data Quality: The technical indicators were obtained from Yahoo Finance, a common practice in prior literature. However, fundamental and macroeconomic indicators were less accessible and had been infrequently cited in past studies. Instead of amalgamating various publicly available datasets and aligning them through forward filling, the use of data from Bloomberg Terminal could have enhanced the validity and accuracy of the dataset, aligning it more closely with the information analysts typically utilize. Moreover, opting for macroeconomic indicators represented by indexes, such as implementing the Corn Index rather than relying on monthly corn prices and em-

ploying forward filling, could have been more efficient. While forward filling did not significantly affect the data quality, it raised concerns when applied to RFE.

Thresholds: The correlation threshold in Bhandari et al. [29] was set at 0.8. However, exploring lower thresholds like 0.5 could have led to more extensive feature elimination, potentially enhancing the efficiency of the C+MI method. Previous studies did not utilize a mutual information threshold of 1, which might have been too low, contributing to the poor performance of the C+MI combination with RFE. The number of features for RFE was determined by testing 25%, 50%, and 75% of features to minimize MSE. Expanding this range might have refined RFE further, potentially removing outliers that preferred no feature significance. Moreover, the soft-threshold function for the Haar wavelet, derived from Donoho and Johnstone [19], might not have been directly applicable to stock market data. While their time series application was invoked, a threshold more aligned with financial time series could prevent over-denoising, thus avoiding potential model overfitting. Though Section 4.2 highlighted the superiority of the denoised model, reconsidering the chosen function remained pertinent. Similarly, with no previous Bayesian optimization grid search, further testing on different parameters could have further optimized the models.

Level of Analysis: Due to space constraints, this paper has offered only a high-level overview of the model and input combinations, which has limited the depth of analysis. It has not thoroughly examined specific stock performance concerning individual error metrics or the influence of the specific Bayesian optimization hyperparameter tuning on stock performance. Moreover, basic technical indicators were not tested with fundamental or macroeconomic indicators, limiting the completeness of the manual indicator analysis. Furthermore, sensitivity analysis could have been conducted on the timeframe to evaluate the impact of the date range on the methodology. Lastly, with multiple pipeline combinations, there had been a possibility of over-tuning the models for the validation data, which could have limited the robustness of the results.

5.2 Further Work

Data & Feature Selection Methods & Hyperparameter Tuning: Text data represented a significant gap in this study, as it was not integrated into the analysis. Incorporating text-based data, like newspaper headlines and social media posts, could have enhanced market sentiment insights and stock prediction accuracy. Despite attempts, challenges such as limited access to credible sources and language-specific processing limitations impeded effective integration. While Google News was web-scraped, it lacked equal coverage across regions; Bloomberg and *The Wall Street Journal* prevented web-scrapping. Moreover, existing NLP models like FINBERT had been optimized for financial jargon and have been currently only publicly available in English. This emphasized the necessity of language-specific approaches, such as developing equivalent models like Finbert in multiple languages. This would have helped clarify optimal model architectures and input combinations for processing both textual and numerical data, thereby enhancing stock prediction accuracy across diverse geographies and linguistic contexts. Furthermore, this paper examined only three feature selection methods,

indicating there is scope for a more thorough analysis. Additional methods like Linear Discriminant Analysis (LDA), which effectively characterizes or separates multiple classes of a categorical variable, could be explored. Tantisripreecha et al. applied LDA to predict stock price movement, achieving over 90% accuracy for four stocks [80]. Finally, expanding testing to include more grid searches and tuning additional parameters could enhance the analysis. While the ReLU activation function has been commonly used in LSTM models, exploring alternatives like tanh and sigmoid, could offer valuable insights. Similarly, while the Adam optimizer is commonly used, investigating alternatives like RMSProp and SGD could broaden the search space. These combinations of optimizers and activation functions were successfully implemented in Lee's Bayesian optimization for stock index prediction, specifically the S&P 500 [71].

Broader Country and Stock Selection: While the optimal model seemed to be consistent across various geographies and industries, this observation was for the largest stocks by market capitalization. Diversifying stocks beyond the largest market caps by including more countries and stocks would allow for a broader representation of industries. As seen in this paper, different industries preferred different indicators at different analysis stages (e.g., the oil industry preferred basic indicators like PB). This expansion could validate hypotheses proposed here, such as the influence of parent companies on subsidiaries across diverse geographies and industries. Currently, selected countries were those with the most literature available and were influential within their continents. However, diversifying to include less dominant countries could reveal behavioral shifts. For instance, a preference for macroeconomic factors over fundamental ones might emerge in countries more reliant on the market. Henceforth, extending the analysis to nations in Asia, Africa, and Oceania could provide additional insights into global dynamics. Though obtaining proper data for these regions may pose challenges without access to a paid database, platforms like Bloomberg provide information on emerging markets. This broader scope could enable a comprehensive analysis for developing nations, offering insights into optimal models and parameters.

Model: This analysis could be expanded to include additional architectures, like SVMs, the second most cited method for stock prediction [55]. Such an addition could provide another layer of analysis, as only LSTMs were analyzed in this study. This extension might reveal correlations between model architecture and data variable frequency. Furthermore, it could investigate the impact of denoising on SVM, an area still under-researched. Neighborhood Rough Sets represent a recent advancement in stock prediction, surpassing neural networks in performance. For instance, Al-Qaheri et al. achieved 97% accuracy in predicting daily stock movements on the Kuwait Stock Exchange, while Khoza and Marwala achieved an 80.4% accuracy [25, 54]. To improve accuracy, analyzing indicator combinations and data preprocessing techniques for neighborhood rough sets could be crucial.

Ultimately, by incorporating previous explorations—like utilizing text data for broader country and stock selections and exploring more hyperparameters and feature selection methods for various models—a deeper understanding of the intricate relationships among all components of stock price prediction methods could emerge. Insights from these analyses could be instrumental in predicting stock indexes or forecasting stock price movements, aiding decisions on when to buy, hold, or sell.

Bibliography

[1] A. O. Adewumi A. A. Ariyo and C. K. Ayo. "Stock Price Prediction Using the ARIMA Model". In: *2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*. 2014, pp. 106–112. DOI: 10.1109/UKSim.2014.67.

[2] A. Aminimehr, and A. Raoofi, A. Aminimehr and A. Aminimehr. "A Comprehensive Study of Market Prediction from Efficient Market Hypothesis up to Late Intelligent Market Prediction Approaches". In: *Computational Economics* 60.2 (2022), pp. 781–815. DOI: https://doi.org/10.1007/s10614-022-10283-1.

[3] A. Dezhkam, M. T. Manzuri, A. Aghapour, A. Karimi, A. Rabiee and S. M. Shalmani. "A Bayesian-based classifcation framework for financial time series trend prediction". In: *Journal of Supercomputing* 79 (2023), pp. 4622–4659. DOI: 10.1007/s11227-022-04834-4.

[4] A. Dingli and K. S. Fournier. "Financial Time Series Forecasting – A Deep Learning Approach". In: (2017). DOI: 10.18178/ijmlc.2017.7.5.632.

[5] A. Gupta, Akansha, K. Joshi, M. Patel and V. Pratap. "Stock Prices Prediction Using Machine Learning". In: *2023 2nd International Conference for Innovation in Technology* (2023). DOI: 979-8-3503-2092-3/23/$31.0.

[6] A. H. Victoria and G. Maragatham. "Automatic Tuning of Hyperparameters using Bayesian Optimization". In: 12 (2020), pp. 217–223. DOI: https://doi.org/10.1007/s12530-020-09345-2.

[7] A. J. Black, O. Klinkowska, D. G. McMillan and F. J. McMillan. "Forecasting Stock Returns: Fo Commofity Prices Help?" In: (2014). DOI: https://dx.doi.org/10.2139/ssrn.2467095.

[8] A. Kraskov, H. Stogbauer and P. Grassberger. "Estimating mutual information". In: *Phys Rev E* 69.6 Pt 2 (2004), p. 066138. DOI: 10.1103/PhysRevE.69.066138.

[9] D. Vaghela A. Mahadik and A. Mhaisgawali. "Stock Price Prediction using LSTM and ARIMA". In: *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*. 2021, pp. 1594–1601. DOI: 10.1109/ICESC51422.2021.9532655.

[10] A. Moghar and M. Hamiche. "Stock Market Prediction Using LSTM Recurrent Neural Network". In: *Procedia Computer Science* 170 (2020), pp. 1168–1173. DOI: https://doi.org/10.1016/j.procs.2020.03.049.

[11] B. Kumari and T. Swarnkar. "Stock movement prediction using hybrid normalization technique and artificial neural network". In: 8 (2021), pp. 1336–1350. DOI: http://dx.doi.org/10.19101/IJATEE.2021.874387.

[12] B. S. Pramod and P. M. Mallikarjuna Shastry. "Stock Price Prediction Using LSTM". In: *Test Engineering and Management* 83 (2021), pp. 5246–5251.

[13] Ben Lobel. *Technical Indicators Defined and Explained*. https://www.dailyfx.com/education/technical-analysis-tools/technical-indicators.html. Accessed February 18, 2024. 2024.

[14] C.-F. Tsai and Y.-C. Hsiao. "Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches". In: *Decision Support Systems* 50.1 (2010), pp. 258–269. DOI: https://doi.org/10.1016/j.dss.2010.08.028. URL: https://www.sciencedirect.com/science/article/pii/S0167923610001521.

[15] C.-F. Tsai, Y.-C. Lin, D. C. Yen and Y.-M. Chen. "Predicting stock returns by classifier ensembles". In: *Applied Soft Computing* 11.2 (2011), pp. 2452–2459. ISSN: 1568-4946. DOI: 10.1016/j.asoc.2010.10.001. URL: https://doi.org/10.1016/j.asoc.2010.10.001.

[16] C.-R. Ko and H.-T. Chang. "LSTM-based sentiment analysis for stock price forecast". In: *PeerJ Computer Science* 7 (2021). Ed. by F. Xia. DOI: https://doi.org/10.7717/peerj-cs.408.

[17] C.A Sims. "Macroeconomics and reality". In: *Econometrica* 48.1 (1980), pp. 1–48.

[18] D. Karmiani, R. Kazi, A. Nambisan, A. Shah and V. Kamble. "Comparison of Predictive Algorithms: Backpropagation, SVM, LSTM and Kalman Filter for Stock Market". In: (2019), pp. 228–234. DOI: 10.1109/AICAI.2019.8701258.

[19] D. L. Donoho and I. M. Johnstone. "Ideal Spatial Adaptation by Wavelet Shrinkage". In: *Biometrika* 81.3 (1994), pp. 425–455. DOI: https://doi.org/10.2307/2337118.

[20] D. M. Rees. *Commodity prices and the US Dollar*. Tech. rep. Bank for International Settlements, 2023.

[21] E. Beyaz, F. Tekiner, X.-J. Zeng and J. Keane. "Comparing Technical and Fundamental Indicators in Stock Price Forecasting". In: 2018, pp. 1607–1613. DOI: 10.1109/HPCC/SmartCity/DSS.2018.00262.

[22] W. E. Ferson and C. R. Harvey. "The Variation of Economic Risk Premiums". In: *Journal of Political Economy* 99.2 (1991), pp. 385–415. URL: http://www.jstor.org/stable/2937686 (visited on 03/31/2024).

[23] G. Li, A. Zhang, Q. Zhang , D. Wu and C. Zhan. "Pearson correlation coefficient-based performance enhancement of Broad Learning System for stock price prediction". In: *IEEE Trans Circuits Syst II* (2022). Early Access. DOI: 10.1109/TCSII.2022.3160266.

[24] G. Li, A. Zhang, Q. Zhang, D. Wu and C. Zhan. "Pearson Correlation Coefficient-Based Performance Enhancement of Broad Learning System for Stock Price Prediction". In: *IEEE Transactions on Circuits and Systems II: Express Briefs* 69.5 (2022). DOI: 10.1109/TCSII.2022.3160266.

[25] H. Al-Qaheri, S. Zamoon, A. L. Hassanien, and A. Abraham. "Rough Set Generating Prediction Rules for Stock Price Movement". In: *2008 Second*

UKSIM European Symposium on Computer Modeling and Simulation. 2008, pp. 111–116. DOI: 10.1109/EMS.2008.89.

[26] H. Gunduz. "An efficient stock market prediction model using hybrid reduction method based on variational autoencoders and recursive feature elimination". In: 7 (2021). DOI: https://doi.org/10.1186/s40854-021-00243-3.

[27] H. Gunduz, Y. Yaslan and Z. Cataltepe. "Intraday prediction of Borsa Istanbul using convolutional neural networks and feature correlations". In: *Knowledge-Based Systems* 137 (2017), pp. 138–148. DOI: 10.1016/j.knosys.2017.09.023.

[28] H. Gündüz,Z. Çataltepe and Y. Yaslan. "Stock daily return prediction using expanded features and feature selection". In: *Turkish Journal of Electrical Engineering and Computer Sciences* 25.6 (2017), Article 32. DOI: 10.3906/elk-1704-256. URL: https://doi.org/10.3906/elk-1704-256.

[29] H. N. Bhandari, B. Rimal, N. R. Pokhrel, R. Rimal, K. R. Dahal, and R. K. C. Khatri. "Predicting stock market index using LSTM". In: *Machine Learning with Application* 9 (2022). DOI: https://doi.org/10.1016/j.mlwa.2022.100320.

[30] H. T. Wong. "The Impact of Real Exchange Rates on Real Stock Prices". In: 27 (2022), pp. 262–276. DOI: 10.1108/JEFAS-03-2021-001.

[31] H. Wu, S. Chen and Y. Ding. "Comparison of ARIMA and LSTM for Stock Price Prediction ". In: *Financial Engineering and Risk Management* 6.1 (2023). DOI: 10.23977/ferm.2023.060101.

[32] *Historical Exchange Rates*. https://www.ofx.com/en-gb/forex-news/historical-exchange-rates/. Accessed February 18, 2024. 2024.

[33] I. Botunac, A. Panjkota and M. Matetic. "The Effect of Feature Selection on the Performance of Long Short-Therm Memory Neural Network in Stock Market Predictions". In: 2020. ISBN: 9783902734297. DOI: 10.2507/31st.daaam.proceedings.081.

[34] I. K. Nti, A. F. Adekoya and B. A. Weyori. "A systematic review of fundamental and technical analysis of stock market predictions". In: (2019), pp. 3007–3057. DOI: https://doi.org/10.1007/s10462-019-09754-z.

[35] International Monetary Fund. *Commodity Data Portal*. https://www.imf.org/en/Research/commodity-prices/. Accessed February 18, 2024. 2024.

[36] International Monetary Fund. *National Accounts and Price Statistics*. https://www.imf.org/en/Data5. Accessed February 18, 2024. 2024.

[37] Investopedia.com. *Efficient Market Hypothesis: Definition and Critique*. https://www.investopedia.com/terms/e/efficientmarkethypothesis.asp. Accessed:2023-10-13.

[38] J. Chai, J. Du, K. Lai and Y. Lee. "A hybrid least square support vector machine model with parameters optimization for stock forecasting". In: *Mathematical Problems in Engineering* (2015). DOI: 10.1155/2015/231394.

[39] J. Conteras, R. Espinola, F. J. Nogales and A. J. Conejo. "Arima models to predict next-day electricity prices". In: *IEEE Transactions on Power Systems* 18.3 (2003), pp. 1014–1020.

[40] J. Shen and M. O. Shafiq. "Short-term stock market price trend prediction using a comprehensive deep learning system". In: 7 (2020). DOI: https://doi.org/10.1186/s40537020003336.

[41] J. Sun, K. Xiao, C. Liu, W. Zhou and H. Xiong. "Exploiting intra-day patterns for market shock prediction: a machine learning approach". In: *Expert Syst Appl* 127 (2019), pp. 272–281. DOI: https://doi.org/10.1016/j.eswa.2019.03.006.

[42] J. T. Firouzjaee, and P. Khalilian. "The Interpretability of LSTM Models for Predicting Oil Company Stocks: Impact of Correlated Features". In: *International Journal of Energy Research* 2024 (2024), pp. 1–18. DOI: https://doi.org/10.1155/2024/5526692.

[43] J. Wu, X.-Y. Chen, H. Zhang, L.-D. Xiong, H. Lei and S.-H. Deng. "Hyperparameter optimization for machine learning models based on Bayesian optimization". In: *Journal of Electronic Science and Technology* 17 (2019), pp. 26–40. DOI: 10.11989/JEST.1674-862X.80904120.

[44] J.-Z. Wang, J.-J. Wang, Z.-G. Zhang and S.-P. Guo. "Forecasting stock indices with back propagation neural network". In: *Expert Systems with Applications* 38.11 (2011), pp. 14346–55. DOI: https://doi.org/10.1016/j.eswa.2011.04.222.

[45] J.D. Hamilton. "A new approach to the economic analysis of nonstationary time series and the business cycle". In: *Econometrica* 57.2 (1989), pp. 357–384.

[46] K. Cabello-Solorzano, I. Ortigosa de Araujo, M. Pena, L. Correia and A. J. Tallon-Ballesteros. "The Imapct of Data Normaliztion on the Accuracy of the Machine Learning Algorithms: A Comparative Analysis". In: (2023), pp. 344–353.

[47] K. Chen, Y. Zhou, and F. Dai. "A LSTM-based method for stock returns prediction: A case study of China stock market". In: Oct. 2015, pp. 2823–2824. DOI: 10.1109/BigData.2015.7364089.

[48] L. Wang and S. Gupta. "Neural Networks and Wavelet De-Noising for Stock Trading and Prediction". In: *Intelligent Systems Reference Library*. 2013, pp. 229–247. DOI: 10.1007/978-3-642-33439-9_11. URL: https://doi.org/10.1007/978-3-642-33439-9_11.

[49] L. Xiang, Z. Ge, L. Sun, M. He and H. Chen. "LSTM with Wavelet Transform Based Data Preprocessing for Stock Price Prediction". In: *Mathematical Problems in Engineering* 2019 (2019), pp. 1–8. DOI: https://doi.org/10.1155/2019/1340174.

[50] M. A. Saeed and A. Jamil. "Stock Price Prediction in Response to US Dollar Exchange Rate Using Machine Learning Techniques". In: Mar. 2023, pp. 281–290. ISBN: 978-3-031-27098-7. DOI: 10.1007/978-3-031-27099-4_22.

[51] M. Agrawal, A. Khan and P. Kumar. "Stock Price Prediction using Technical Indicators: A Predictive Model using Optimal Deep Learning". In: *International Journal of Recent Technology and Engineering* 8.2 (2019), pp. 2297–2305. DOI: 10.35940/ijrte.b3048.078219. URL: https://doi.org/10.35940/ijrte.b3048.078219.

[52] M. Aufaristama. *The Stock Price Relationship between Holding Companies and Subsidiaries: A Case study of Indonesia Multiholding Companies*. 2023. DOI: 10.48550/arXiv.2303.07244.

[53] M. G. Alqahtani and H. A. Abdelhafez. "STOCK MARKET PREDICITION USING STATISTICAL DEEP LEARNING TECHNIQUES". In: *Journal of Theoretical and Applied Information Technology* 101.23 (2023).

[54] M. Khoza, and T. Marwala. "A rough set theory based predictive model for stock prices". In: *2011 IEEE 12th International Symposium on Computational Intelligence and Informatics (CINTI)*. 2011, pp. 57–62. DOI: 10.1109/CINTI.2011.6108571.

[55] M. M. Kumbure, C. Lohrmann, P. Luukka, and J. Porras. "Machine learning techniques and data for stock market forecasting: A literature review". In: *Expert Systems with Applications* 197 (2 2022). DOI: https://doi.org/10.1016/j.eswa.2022.116659.

[56] M. Moran. *Performance and Volatility for Sectors in the 2010s*. https://www.spglobal.com/en/research-insights/articles/performance-and-volatility-for-sectors-in-the-2010s. Jan. 2020.

[57] A. Mohan. *Bayesian Optimization and Hyperparameter Tuning*. https://towardsdatascience.com/bayesian-optimization-and-hyperparameter-tuning-6a22f14cb9fa. Accessed: February 18, 2024. 2021.

[58] N. Adithyan. *Algorithmic Trading with Python*. https://github.com/Nikhil-Adithyan/Algorithmic-Trading-with-Python/tree/main. Accessed February 18, 2024. 2024.

[59] N. Diamond and J. Wright. "IMPACT OF INTERMARKET DATA ON STOCK MARKET PRE, journal=WORCESTER POLYTECHNIC INSTITUTE". In: (2022).

[60] O. B. Sezer and M. U. Gudelek and A. M. Ozbayoglu. "Financial time series forecasting with deep learning: A systematic literature review: 2005–2019". In: *Applied Soft Computing* 90 (2020), p. 106181. ISSN: 1568-4946. DOI: 10.1016/j.asoc.2020.106181. URL: https://doi.org/10.1016/j.asoc.2020.106181.

[61] O. Bustos and A. Pomares-Quimbaya. "Stock market movement forecast: A systematic review". In: *Expert Systems with Applications* 156 (2020). DOI: https://doi.org/10.1016/j.eswa.2020.113464.

[62] P. N. Patatoukas. "Stock market Returns and GDP News". In: 36 (2020). DOI: 10.1177/0148558X20913418.

[63] P. Tufekci. "Predicting the Direction of Movement for Stock Price Index Using Machine Learning Methods". In: (2016), pp. 477–492. DOI: https://doi.org/10.1007/978-3-319-29504-6_45.

[64] peerchemist. *Financial Technical Analysis*. https://pypi.org/project/finta/. Accessed February 18, 2024. 2024.

[65] R. A. Kamble. "Short and long term stock trend prediction using decision tree". In: *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*. 2017, pp. 1371–1375. DOI: 10.1109/ICCONS.2017.8250694.

[66] R. G. Ahangar, M. Yahyazadehfar and H. Pournaghshbad. "The comparison of methods artificial neural network with linear regression using specific variables

for prediction stock price in Tehran stock exchange". In: *International Journal of Computer Science and Information Security* 7 (2010).

[67] R. Shan, H. Dai, J. Zhao and W. Liu. "Forecasting Study of Shanghai's and Shenzhen's Stock Markets Using a Hybrid Forecast Method". In: *Communications in Statistics - Simulation and Computation* 44 (2015), pp. 1066–1077. URL: https://api.semanticscholar.org/CorpusID:33137396.

[68] R. V. Lakshmi and S. Radha. "Time series forecasting for the adobe software company's stock prices using ARIMA model". In: *Journal of Physics: Conference Series* 2115 (2021), p. 012044.

[69] R.S.Tsay. *Analysis of Financial Time Series*. Vol. 543. John Wiley & Sons, 2005.

[70] S. Hochreiter. "The vanishing gradient problem during learning recurrent neural nets and problem solutions". In: 6 (1998), pp. 107–116. DOI: https://doi.org/10.1142/S0218488598000094.

[71] S. I. Lee. *Hyperparameter Optimization for Forecasting Stock Returns*. Tech. rep. Deep Allocation Technologies, 2020.

[72] S. K. Lakshminarayanan and J. McCrae. "A Comparative Study of SVM and LSTM Deep Learning Algorithms for Stock Market Prediction". In: *Irish Conference on Artificial Intelligence and Cognitive Science*. 2019. URL: https://api.semanticscholar.org/CorpusID:212411636.

[73] S. Kumar Chandar, M. Sumathi and S.N. Sivanandam. "Prediction of Stock market Price Using Hybrid of Wavelet Transform and Artificial Neural Network". In: 9 (2016). DOI: 10.17485/ijst/2016/v9i8/87905.

[74] S. Mittal and A. Chauhan. "A RNN-LSTM-Based Predictive Modelling Framework for Stock Market Prediction Using Technical Indicators". In: *International Journal of Rough Sets and Data Analysis* 7 (Jan. 2021), pp. 1–13. DOI: 10.4018/IJRSDA.288521.

[75] S. Siami-Namini, N. Tavakoli, and A. S. Namin. "A Comparison of ARIMA and LSTM in Forecasting Time Series". In: *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. 2018, pp. 1394–1401. DOI: 10.1109/ICMLA.2018.00227.

[76] S. Zhong And D. B. Hitchcock. *SP 500 Stock Price Prediction Using Technical, Fundamental and Text Data*. Tech. rep. University of South Carolina, 2021.

[77] S.C. Nayak, B.B. Misra and H. S. Behera. "Impact of Data Normalization on Stock Index Forecasting". In: *International Journal of Computer Information Systems and Industrial Management Applications* 6 (2014), pp. 257–269.

[78] T. G. Smith. *pmdarima*. https://alkaline-ml.com/pmdarima/modules/generated/pmdarima.arima.auto_arima.html.

[79] T. Swathi, N. Kasiviswanath and A. Ananda Rao. "An optimal deep learning-based LSTM for stock price prediction using twitter sentiment analysis". In: *Applied Intelligence* 52 (Mar. 2022). DOI: 10.1007/s10489-022-03175-2.

[80] T. Tantisripreecha and N. Soonthomphisaj. "Stock Market Movement Prediction using LDA-Online Learning Model". In: *2018 19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*. 2018, pp. 135–139. DOI: 10.1109/SNPD.2018.8441038.

[81] T. W. A. khairi, R. M. Zaki and W.A. Mahmood. "Stock Price Prediction using Technical, Fundamental and News based Approach". In: *2019 2nd Scientific Conference of Computer Sciences (SCCS)*. 2019, pp. 177–181. DOI: 10.1109/SCCS.2019.8852599.

[82] T.-J. Hsieh, H.-F. Hsio and W.-C. Yah. "Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm". In: 11 (2011), pp. 2510–2525. DOI: https://doi.org/10.1016/j.asoc.2010.09.007.

[83] " The impact of fundamental elements on stock prices of subsidiary pulp paper manufacturing firms listed on the indonesia stock exchange ". In: *Zero* 6.2 (2023), pp. 118–118. DOI: 10.30829/zero.v6i2.14623.

[84] W. Bao, J. Yue and Y. Rao. "A deep learning framework for financial time series using stacked autoencoders and long-short term memory". In: *PLoS ONE* 12.7 (2017), e0180944. DOI: 10.1371/journal.pone.0180944. URL: https://doi.org/10.1371/journal.pone.0180944.

[85] W. Huang. *Enhancing Stock Market Prediction Through LSTM Modeling and Analysis*. Tech. rep. School of Software, South China Normal University, 2023.

[86] W.-F. Pan. "Does the stock market really cause unemployment? A cross-country analysis". In: 44 (2018), pp. 34–43. DOI: https://doi.org/10.1016/j.najef.2017.11.002.

[87] X. Cai, S. Hu and X. Lin. "Feature extraction using restricted Boltzmann machine for stock price prediction". In: *IEEE CSAE*. 2012, pp. 80–83.

[88] X. Guo, H. Zhang and T. Tian. "Development of stock correlation networks using mutual information and financial big data". In: *PLoS One* 13.4 (2018), e0195941. DOI: 10.1371/journal.pone.0195941.

[89] X. Yan, Q. Cai, S. Zhang and T. Yu. "Exploring Machine Learning in Stock Prediction Using LSTM, Binary Tree, and Linear Regression Algorithms". In: *International Core Journal Engineering* 7 (2023). DOI: 10.6919/ICJE.202103_7(3).0049.

[90] X. Yuan, J. Yuan, T. Jiang, and Q.U. Ain. "Integrated Long-Term Stock Selection Models Based on Feature Selection and Machine Learning Algorithms for China Stock Market". In: *IEEE Access* 8 (2020), pp. 22672–22685. DOI: https://doi.org/10.1109/access.2020.2969293.

[91] X.Zhong and D. Enke. "A comprehensive cluster and classification mining procedure for daily stock market return forecasting". In: (2017). DOI: https://doi.org/10.1016/j.neucom.2017.06.010.

[92] Y. Alsubaie, K. E. Hindi and H. Alsalman. "Cost-sensitive prediction of stock price direction: selection of technical indicators". In: *IEEE Access* 7 (2019), pp. 146876–146892.

[93] Y. Chen and Y. Hao. "A feature weighted support vector machine and K-nearest neighbor algorithm for stock market indices prediction". In: *Expert Syst Appl* 80 (2017), pp. 340–355. DOI: https://doi.org/10.1016/j.eswa.2017.02.044.

[94] Y. Gür. "Stock Price Forecasting Using Machine Learning and Deep Learning Algorithms: A Case Study for the Aviation Industry". In: *Fırat Üniversitesi Müh. Bil. Dergisi Araştırma Makalesi* 36.1 (2024), pp. 25–34. DOI: https:

//doi.org/10.35234/fumbd.1357613. URL: https://dergipark.org.tr/en/download/article-file/3396115.

[95] Y. Huang, L.F. Capretz and D. Ho. "Machine Learning for Stock Prediction Based on Fundamental Analysis". In: *IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE Press, 2021. DOI: 10.1109/SSCI50451.2021.9660134. URL: https://ieeexplore.ieee.org/document/9660134.

[96] Y. Kara, M. A. Boyacioglu and O. K. Baykan. "Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange". In: *Expert Systems with Applications* 38.5 (2011), pp. 5311–5319. DOI: https://doi.org/10.1016/j.eswa.2010.10.027.

[97] Y. Kim. "Toward a successful CRM: variable selection, sampling, and ensemble". In: *Decision Support Systems* 41.2 (2006), pp. 542–553. DOI: https://doi.org/10.1016/j.dss.2004.09.008.

[98] Y. Wang. *What are the Advantages and Disadvantages of Random Forest?* https://www.rebellionresearch.com/what-are-the-advantages-and-disadvantages-of-random-forest. Accessed: February 18, 2024. 2023.

[99] Y. Wang and Y. Guo. "Forecasting method of stock market volatility in time series data based on mixed model of ARIMA and XGBoost". In: *China Communications* 17.3 (2020), pp. 205–221. DOI: 10.23919/jcc.2020.03.017. URL: https://doi.org/10.23919/jcc.2020.03.017.

[100] Y. Yang. *Forecasting Apple Stock Closed Prices by LR and LSTM with Discrete Wavelet Transformation*. Tech. rep. Proceedings of the 2022 2nd International Conference on Economic Development and Business Culture, 2022, pp. 935–943. DOI: https://doi.org/10.2991/978-94-6463-036-7_138.

[101] Z. B. Zou and Z. Qu. "Using LSTM in Stock prediction and Quantitative Trading". In: 2020. URL: https://api.semanticscholar.org/CorpusID:215776654.

[102] Z. Zhang. "Comparison of LSTM and ARIMA in Price Forecasting: Evidence from Five Indexes". In: *Advances in economics, business and management research* (Oct. 2023), pp. 40–46. DOI: https://doi.org/10.2991/978-94-6463-268-2_6.

[103] Zacks Investment Research Inc. *MacroTrends*. https://www.macrotrends.net/stocks/stock-screener. Accessed February 18, 2024. 2024.

Appendix A

Appendix

A.1 Indicators

Table A.1: Basic Technical Indicators

Variable	**Definition**
Open price	The price of a stock at the beginning of a trading session.
Close price	The price of a stock at the end of a trading session.
High price	The highest price at which a stock trades during a particular period.
Low price	The lowest price at which a stock trades during a particular period.
Volume	The total number of shares traded during a given period, typically a trading session.

Table A.2: Momentum Advanced Technical Indicators

Variable	Definition
Commodity Channel Index (CCI)	A momentum-based oscillator used to identify cyclical trends in stock prices and detect overbought or oversold conditions.
Moving Average Convergence/Divergence (MACD) Indicator	A trend-following momentum indicator that shows the relationship between two moving averages of a security's price.
Williams (%R) Indicator	A momentum indicator used to identify overbought or oversold conditions in a stock by measuring the level of the stock's closing price relative to its high-low range over a specific period.
Disparity Index	A technical indicator that measures the percentage difference between the current price and a chosen moving average.
Relative Strength Index (RSI)	A momentum oscillator that measures the speed and change of price movements. It is used to identify overbought or oversold conditions in a stock.
Stochastic Oscillator (%K)	A momentum indicator that compares a security's closing price to its price range over a specific period.
Stochastic Oscillator (%D)	A smoothed version of the %K oscillator, often used as a signal line to generate buy or sell signals.

Table A.3: Trend Advanced Technical Indicators

Variable	Definition
Exponential Moving Average (EMA)	A type of moving average that gives more weight to recent price data, making it more responsive to recent price changes than the Simple Moving Average.
Simple Moving Average (SMA)	A calculation used to analyze data points by creating a series of averages of different subsets of the full data set.
Average Directional Index (ADX)	A technical analysis indicator used to measure the strength of a trend, regardless of its direction.
Directional Movement Index (DMI)	A technical indicator that measures the strength of a price movement in a specific direction over a given period, helping traders identify strong trends and potential entry points.
Positive DMI	Indicates bullish strength in the market.
Negative DMI	Indicates bearish strength in the market.

Table A.4: Volatility Advanced Technical Indicators

Variable	**Definition**
Bollinger Bands	A technical analysis tool consisting of three lines - an upper band, a lower band, and a rolling mean - used to measure volatility and identify potential overbought or oversold conditions in a market.
Upper Band	The upper boundary of the Bollinger Bands, typically set at two standard deviations above the rolling mean.
Lower Band	The lower boundary of the Bollinger Bands, usually set at two standard deviations below the rolling mean.
Rolling Mean	The middle line of the Bollinger Bands, representing the average price over a specified period.

Table A.5: Income Statement Fundamental Indicators

Metric	Description
Revenue	The total amount of money generated by a company from its business activities, typically from sales of goods or services.
Research and Development Expenses	The costs incurred by a company to develop new products or improve existing ones.
SG&A Expenses (Selling, General, and Administrative Expenses)	The operating expenses incurred by a company not directly associated with production, such as marketing, salaries, and office rent.
Operating Expenses	The total expenses incurred by a company to maintain its operations, including both production and non-production costs.
Gross Profit	The difference between revenue and the cost of goods sold, representing the profit before deducting operating expenses.
Cost of Goods Sold (COGS)	The direct costs associated with producing goods sold by a company.
Operating Income	The profit earned from a company's normal business operations, calculated by subtracting operating expenses from gross profit.
Total non-Operating Income/Expense	The income or expense generated from activities not related to a company's core business operations, such as investments or interest income.
Pre-Tax Income	The total income earned by a company before accounting for taxes.
Income Taxes	The taxes a company owes to the government based on its taxable income.
Income After Taxes	The net income remaining after deducting income taxes from pre-tax income.
Income from Continuous Operations	The income generated from a company's ongoing business activities, excluding one-time gains or losses.
Net Income	The total profit remaining after deducting all expenses, including taxes, interest, and depreciation, from total revenue.
EBITDA (Earnings Before Interest, Taxes, Depreciation, and Amortization)	A measure of a company's operating performance, calculated by adding back interest, taxes, depreciation, and amortization to net income.
EBIT (Earnings Before Interest and Taxes)	Similar to EBITDA but excludes depreciation and amortization.
Basic Shares Outstanding	The total number of shares issued by a company, excluding any dilutive effects such as stock options or convertible securities.
Shares Outstanding	The total number of shares issued by a company, including any dilutive effects.
Basic Earnings Per Share	The portion of a company's profit allocated to each outstanding share of common stock, calculated by dividing net income by the number of basic shares outstanding.
Earnings Per Share (EPS)	Similar to Basic EPS but may include adjustments for dilution.

Table A.6: Balance Sheet Fundamental Indicators

Variable	**Definition**
Cash on Hand	The amount of cash and cash equivalents readily available to a company.
Receivables	The money owed to a company by its customers for goods or services provided on credit.
Other Current Assets	Assets that are expected to be converted into cash or used up within one year, excluding cash, receivables, and inventory.
Total Current Assets	The total value of a company's short-term assets, including cash, receivables, inventory, and other current assets.
Long-term Investments	Investments held by a company for more than one year, typically consisting of stocks, bonds, or real estate.
Goodwill and Intangible Assets	The value of non-physical assets such as trademarks, patents, and brand recognition.
Other Long-Term Assets	Long-term assets other than goodwill and intangible assets, such as property, plant, and equipment.
Total Long-Term Assets	The total value of a company's long-term assets, including long-term investments, goodwill, and other long-term assets.
Total Assets	The sum of a company's total current assets and total long-term assets, representing the value of all assets owned by the company.
Total Current Liabilities	The total amount of money a company owes to creditors for obligations due within one year.
Long-Term Debt	The portion of a company's debt that is due beyond one year.
Other Non-Current Liabilities	Non-current liabilities other than long-term debt, such as deferred tax liabilities or pension obligations.
Total Long-Term Liabilities	The sum of long-term debt and other non-current liabilities, representing the total amount of money owed by a company beyond one year.
Total Liabilities	The sum of a company's total current liabilities and total long-term liabilities, representing the total amount of money owed by the company.
Common Stock Net	The value of common stock issued by a company, net of any treasury stock.
Retained Earnings	The cumulative net income retained by a company after paying dividends to shareholders.
Other Shareholders Equity	Equity other than common stock and retained earnings, such as additional paid-in capital or accumulated other comprehensive income.
Shareholders Equity	The difference between a company's total assets and total liabilities, representing the net worth of the company attributable to its shareholders.
Total Liabilities and Shareholders Equity	The sum of total liabilities and shareholders equity, representing the company's total financing.

Table A.7: Cash Flow Fundamental Indicators

Variable	**Definition**
Net Income Loss	The difference between a company's total revenue and total expenses over a specific period, indicating profitability or loss.
Total Depreciation and Amortization	The total amount of depreciation and amortization expenses incurred by a company over a specific period.
Total Non-Cash Items	The total value of non-cash transactions recorded in a company's financial statements, such as stock-based compensation or changes in fair value of assets.
Total Change in Assets/Liabilities	The net change in a company's assets and liabilities over a specific period, reflecting changes in operating activities, investments, and financing.
Cash Flow From Operating Activities	The net cash generated or used by a company's normal business operations, including revenue, expenses, and changes in working capital.
Net Acquisitions/Divestitures	The net cash spent or received by a company from acquiring or divesting assets or subsidiaries.
Net Change In Investments - Total	The net change in a company's investment portfolio over a specific period.
Cash Flow From Investing Activities	The net cash generated or used by a company's investing activities, including purchases and sales of investments, property, plant, and equipment.
Total Common And Preferred Stock Dividends Paid	The total amount of cash paid by a company to its shareholders as dividends on both common and preferred stock.
Cash Flow From Financial Activities	The net cash generated or used by a company's financing activities, including issuance or repurchase of equity and debt securities.
Net Cash Flow	The total net increase or decrease in a company's cash and cash equivalents over a specific period.

Table A.8: Key Financial Ratios Fundamental Indicators

Variable	Definition
Current Ratio	A measure of a company's liquidity, calculated by dividing its current assets by its current liabilities.
Long-term Debt / Capital	A measure of a company's leverage, calculated by dividing its long-term debt by its total capital (long-term debt plus shareholders' equity).
Debt/Equity Ratio	A measure of a company's financial leverage, calculated by dividing its total debt by its total shareholders' equity.
Gross Margin	The percentage of revenue remaining after deducting the cost of goods sold, representing the profitability of a company's core business activities.
Operating Margin	The percentage of revenue remaining after deducting operating expenses, representing the profitability of a company's normal business operations.
EBIT Margin	The percentage of revenue remaining after deducting operating expenses and non-operating expenses, but before interest and taxes, from total revenue.
Pre-Tax Profit Margin	The percentage of revenue remaining after deducting all expenses except taxes, from total revenue.
Receivable Turnover	A financial ratio that measures a company's ability to collect its accounts receivable during a specific period, calculated by dividing total credit sales by the average accounts receivable.
Days Sales In Receivables	The average number of days it takes for a company to collect payment from its customers after making a sale, calculated by dividing 365 days by the receivable turnover ratio.
ROE (Return On Equity)	A financial ratio that measures a company's profitability relative to its shareholders' equity, calculated by dividing net income by shareholders' equity.
Return On Tangible Equity	A variation of ROE that excludes intangible assets from the equity calculation, providing a more conservative measure of a company's profitability.
ROA (Return On Assets)	A financial ratio that measures a company's profitability relative to its total assets, calculated by dividing net income by total assets.
ROI (Return On Investment)	A financial metric that evaluates the efficiency or profitability of an investment relative to its cost, calculated by dividing the net profit from the investment by the initial cost of the investment.
Operating Cash Flow Per Share	A measure of a company's ability to generate cash from its core business operations per outstanding share of common stock, calculated by dividing operating cash flow by the number of shares outstanding.

Table A.9: Commodities Indicators

Crude oil, average	Crude oil, Brent	Crude oil, Dubai	Crude oil, WTI
Coal, Australian	Coal, South African	Natural gas, US	Natural gas, Europe
Liquefied natural gas, Japan	Natural gas index	Cocoa	Coffee, Arabica
Coffee, Robusta	Tea, avg 3 auctions	Tea, Colombo	Tea, Kolkata
Tea, Mombasa	Coconut oil	Groundnuts	Fish meal
Groundnut oil	Palm oil	Palm kernel oil	Soybeans
Soybean oil	Soybean meal	Rapeseed oil	Sunflower oil
Barley	Maize	Sorghum	Rice, Thai 5%
Rice, Thai 25%	Rice, Thai A.1	Rice, Vietnamese 5%	Wheat, US SRW
Wheat, US HRW	Banana, Europe	Banana, US	Orange
Beef	Chicken	Lamb	Shrimps, Mexican
Sugar, EU	Sugar, US	Sugar, world	Tobacco, US import u.v.
Logs, Cameroon	Logs, Malaysian	Sawnwood, Cameroon	Sawnwood, Malaysian
Plywood	Cotton, A Index	Rubber, TSR20	Rubber, RSS3
Phosphate rock	DAP	TSP	Urea
Potassium chloride	Aluminum	Iron ore, cfr spot	Copper
Lead	Tin	Nickel	Zinc
Gold	Platinum	Silver	

Table A.10: Exchange Rate Indicators

USD/REAL	USD/RUPEE	USD/EURO
USD/GBP	GBP/USD	USD/JPY
USD/CNY	RUPEE/REAL	RUPEE/EURO

Table A.11: Labor Macroeconomic Indicators

Variable	**Definition**
Consumer Price Index (CPI)	A measure of the average change over time in the prices paid by urban consumers for a basket of consumer goods and services, representing inflation and reflecting changes in purchasing power.
Prices, Consumer Price Index, All items, Index	An index that measures the average price level of a basket of consumer goods and services, representing inflation for all items.
Economic Activity, Industrial Production, Index	An index that measures the volume of production of industrial goods over time, representing economic activity and industrial output.

Table A.12: GDP Macroeconomic Indicators: Part 1

Variable	Definition
Household Consumption Expenditure, incl. NPISHs, Real, Seasonally Adjusted, Domestic Currency	The total expenditure by households, including non-profit institutions serving households (NPISHs), adjusted for inflation and seasonal variations, measured in domestic currency.
Gross Domestic Product, Real, Undjusted, Domestic Currency	The total monetary value of all goods and services produced within a country's borders, adjusted for inflation but not seasonally adjusted, measured in domestic currency.
Exports of Goods and Services, Nominal, Undjusted, Domestic Currency	The total value of goods and services sold to foreign countries, not adjusted for inflation and not seasonally adjusted, measured in domestic currency.
Change in Inventories, Nominal, Undjusted, Domestic Currency	The difference in the value of inventories held by firms between two points in time, not adjusted for inflation and not seasonally adjusted, measured in domestic currency.
Household Consumption Expenditure, incl. NPISHs, Nominal, Seasonally Adjusted, Domestic Currency	The total expenditure by households, including non-profit institutions serving households (NPISHs), adjusted for seasonal variations but not inflation, measured in domestic currency.
Imports of Goods and Services, Real, Seasonally Adjusted, Domestic Currency	The total value of goods and services imported from foreign countries, adjusted for inflation and seasonal variations, measured in domestic currency.
Government Final Consumption Expenditure, Real, Seasonally adjusted, Domestic Currency	The total expenditure by the government on final goods and services, adjusted for inflation and seasonal variations, measured in domestic currency.
Imports of Goods and Services, Real, Undjusted, Domestic Currency	The total value of goods and services imported from foreign countries, not adjusted for inflation and not seasonally adjusted, measured in domestic currency.
Imports of Goods and Services, Nominal, Seasonally Adjusted, Domestic Currency	The total value of goods and services imported from foreign countries, adjusted for seasonal variations but not inflation, measured in domestic currency.
Gross Fixed Capital Formation, Nominal, Seasonally Adjusted, Domestic Currency	The total investment in fixed assets such as machinery, buildings, and infrastructure, adjusted for seasonal variations but not inflation, measured in domestic currency.
Change in Inventories, Nominal, Seasonally Adjusted, Domestic Currency	The difference in the value of inventories held by firms between two points in time, adjusted for seasonal variations but not inflation, measured in domestic currency.
Exports of Goods and Services, Real, Seasonally Adjusted, Domestic Currency	The total value of goods and services sold to foreign countries, adjusted for inflation and seasonal variations, measured in domestic currency.

Table A.13: GDP Macroeconomic Indicators: Part 2

Variable	Definition
Gross Domestic Product, Deflator, Seasonally Adjusted	A measure of the change in prices for all goods and services produced domestically, adjusted for seasonal variations.
Household Consumption Expenditure, incl. NPISHs, Real, Undjusted, Domestic Currency	The total expenditure by households, including non-profit institutions serving households (NPISHs), adjusted for inflation but not seasonally adjusted, measured in domestic currency.
Household Consumption Expenditure, incl. NPISHs, Nominal, Undjusted, Domestic Currency	The total expenditure by households, including non-profit institutions serving households (NPISHs), not adjusted for inflation and not seasonally adjusted, measured in domestic currency.
Imports of Goods and Services, Nominal, Undjusted, Domestic Currency	The total value of goods and services imported from foreign countries, not adjusted for inflation and not seasonally adjusted, measured in domestic currency.
Gross Fixed Capital Formation, Real, Seasonally Adjusted, Domestic Currency	The total investment in fixed assets such as machinery, buildings, and infrastructure, adjusted for inflation and seasonal variations, measured in domestic currency.
Gross Domestic Product, Nominal, Seasonally Adjusted, Domestic Currency	The total monetary value of all goods and services produced within a country's borders, adjusted for inflation and seasonal variations, measured in domestic currency.
Gross Domestic Product, Real, Seasonally Adjusted, Domestic Currency	The total monetary value of all goods and services produced within a country's borders, adjusted for inflation and seasonal variations, measured in domestic currency.
Exports of Goods and Services, Nominal, Seasonally Adjusted, Domestic Currency	The total value of goods and services sold to foreign countries, adjusted for seasonal variations but not inflation, measured in domestic currency.
Government Final Consumption Expenditure, Nominal, Seasonally adjusted, Domestic Currency	The total expenditure by the government on final goods and services, not adjusted for inflation and not seasonally adjusted, measured in domestic currency.
Gross Domestic Product, Nominal, Undjusted, Domestic Currency	The total monetary value of all goods and services produced within a country's borders, not adjusted for inflation and not seasonally adjusted, measured in domestic currency.
Exports of Goods and Services, Real, Undjusted, Domestic Currency	The total value of goods and services sold to foreign countries, adjusted for inflation but not seasonally adjusted, measured in domestic currency.

A.2 Data Preparation

A.2.1 Forward Filling & Skewness

Table A.14: Percentage of Country Based Indicators Forward Filled Data

Country	International Liquidity	CPI	Labor
Germany	97.13%	97.77%	97.70%
Brazil	97.24%	96.73%	97.20%
USA	97.80%	96.81%	96.84%

Table A.15: Percentage of Stock Based Indicators Filled Data

Stock	**Technical Indicators**	**Fundamental Indicators**	**Total Dataset**
SAP	17.59%	98.92%	78.67%
Siemens	17.92%	98.95%	78.73%
Deutsche Telekom	19.31%	98.92%	78.97%
Apple	15.82%	98.89%	78.16%
Microsoft	17.17%	98.89%	78.41%
Alphabet	15.97%	98.90%	78.19%
Ambev	20.51%	98.89%	79.06%
Petroleo Brasiliero	20.53%	98.98%	79.08%
Vale	18.57%	98.94%	78.72%

Table A.16: Indicators Removed due to NaN Skewness

CPI
Household Consumption Expenditure, incl. NPISHs, Real, Seasonally Adjusted, Domestic Currency
Gross Domestic Product, Real, Undjusted, Domestic Currency
Exports of Goods and Services, Nominal, Undjusted, Domestic Currency
Change in Inventories, Nominal, Undjusted, Domestic Currency
Household Consumption Expenditure, incl. NPISHs, Nominal, Seasonally Adjusted, Domestic Currency
Imports of Goods and Services, Real, Seasonally Adjusted, Domestic Currency
Government Final Consumption Expenditure, Real, Seasonally adjusted, Domestic Currency
Imports of Goods and Services, Real, Undjusted, Domestic Currency
Imports of Goods and Services, Nominal, Seasonally Adjusted, Domestic Currency
Gross Fixed Capital Formation, Nominal, Seasonally Adjusted, Domestic Currency
Change in Inventories, Nominal, Seasonally Adjusted, Domestic Currency
Exports of Goods and Services, Real, Seasonally Adjusted, Domestic Currency
Household Consumption Expenditure, incl. NPISHs, Real, Undjusted, Domestic Currency
Household Consumption Expenditure, incl. NPISHs, Nominal, Undjusted, Domestic Currency
Imports of Goods and Services, Nominal, Undjusted, Domestic Currency
Gross Fixed Capital Formation, Real, Seasonally Adjusted, Domestic Currency
Gross Domestic Product, Nominal, Seasonally Adjusted, Domestic Currency
Gross Domestic Product, Real, Seasonally Adjusted, Domestic Currency
Exports of Goods and Services, Nominal, Seasonally Adjusted, Domestic Currency
Government Final Consumption Expenditure, Nominal, Seasonally adjusted, Domestic Currency
Gross Domestic Product, Nominal, Undjusted, Domestic Currency
Exports of Goods and Services, Real, Undjusted, Domestic Currency
International Reserves, Official Reserve Assets, SDRs, US Dollars
International Liquidity, Total Reserves excluding Gold, US Dollars
International Liquidity, Total Reserves excluding Gold, Foreign Exchange, USD
International Liquidity, Gold Holdings, National Valuation, US Dollars
International Reserves, Official Reserve Assets, IMF Reserve Position, USD
Barley
Sorghum
Rice, Viet Namese 5%
Wheat, US SRW

A.2.2 Wavelet Transformation

A wavelet, denoted as $\psi(t)$, is essentially a function of time t that adheres to a fundamental rule known as the wavelet admissibility condition [82]:

$$C_\phi = \int_0^\infty \frac{|\psi(f)|}{f}\, df \tag{A.1}$$

Here, $\psi(f)$ represents the Fourier transform, a function of frequency f, of $\psi(t)$.

The decomposition achieved through wavelets segregates the signal into smooth coefficients a and detail coefficients d, represented by equations (A.2) and (A.3)[73]:

$$A_{ij} = \int o(t)\Phi_{ij}(t)\, dt \tag{A.2}$$

$$D_{ij} = \int o(t)\Psi_{ij}(t)\, dt \tag{A.3}$$

Here, Φ and Ψ denote the father and mother wavelets, while j and k represent scaling and translation parameters. The father wavelet approximates the approximation coefficients, whereas the mother wavelet approximates the detailed coefficients. The expressions for Φ and Ψ are defined as [82]:

$$\Phi_{ij}(t) = 2^{-\frac{i}{2}} \cdot \Phi\left(2^{-i}j - j\right) \tag{A.4}$$

$$\Psi_{ij}(t) = 2^{-\frac{i}{2}} \cdot \Psi_{(2^{-i}j-j)} \tag{A.5}$$

These wavelets fulfill the admissibility conditions given by equations (A.6) and (A.7):

$$\int \Phi(t)\, dt = 1 \tag{A.6}$$

$$\int \Psi(t)\, dt = 0 \tag{A.7}$$

Time series data, denoted as $o(t)$, serves as the input in wavelet analysis, expressed as a series of projections onto father and mother wavelets indexed by both k, with $k = \{0,1,2,...\}$, and by $a = 2^j$, with $j = \{1,2,3,...J\}$. To ensure accurate analysis of discretely sampled data, establishing a lattice for computational purposes is essential. The expansion coefficients are derived from these projections [82], where $j = (1,2...J)$:

$$a_{i,j} = \int \Phi_{i,j}(t) \cdot f(t)\, dt \tag{A.8}$$

$$d_{i,j} = \int \Psi_{i,j}(t) \cdot f(t)\, dt \tag{A.9}$$

The orthogonal wavelet representation of the original signal $o(t)$ is defined as [73]:

$$o(t) = \sum_j a_{ij} \cdot \Phi_{ij} + \sum_j d_{ij} \cdot \Psi_{ij} + \sum_j d_{i-1,j} \cdot \Psi_{i-1,j} + \ldots + \sum_j d_{1,j} \cdot \Psi_{1,j} \tag{A.10}$$

Algorithm 2 Wavelet-Based Signal Denoising

1: Let signal be the input signal.
2: Perform 2-level wavelet decomposition using a specific wavelet function, resulting in detail coefficients coeffs.
3: Extract detail coefficients at level 1 (finest scale), denoted as $d1$.
4: Estimate noise variance σ^2 from the finest wavelet coefficients $d1$ using the median absolute deviation: $\sigma = \frac{\text{median}(|d1|)}{0.6745}$
5: Determine noise threshold λ: $\lambda = \sigma\sqrt{2\ln(N)}$, where N is the length of the signal.
6: Perform soft thresholding using the estimated threshold λ on $d1$: thresholded_d1 = soft_threshold$(d1, \lambda)$
7: Construct the list of detail coefficients in the correct format: detail_coefficients = $[\text{thresholded\_d1}] + [None] \times (\text{len}(\text{coeffs}) - 2)$
8: Reconstruct the denoised signal using the inverse wavelet transform: denoised_signal = inverse_wavelet_transform$([\text{coeffs}[0]] + \text{detail_coefficients})$
9: Trim the denoised signal to match the length of the original signal: denoised_signal = denoised_signal$[: \text{len}(\text{signal})]$
10: Update the original signal with the denoised signal: signal = denoised_signal

A.2.3 Normalization

Min-Max normalization uses the formula

$$X_{\text{norm}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \tag{A.11}$$

where X is the original value, X_{min} is the minimum value in the dataset, X_{max} is the maximum value in the dataset, and X_{norm} is the normalized value [46].

Z-score normalization employs the formula

$$Z = \frac{X - \mu}{\sigma} \tag{A.12}$$

where X is the original value, μ is the mean of the dataset, σ is the standard deviation of the dataset, and Z is the z-score normalized value [46].

The robust normalization formula is given by:

$$X_{\text{norm}} = \frac{X - Q_1}{Q_3 - Q_1} \tag{A.13}$$

where X is the original value, Q_1 is the first quartile (25th percentile) of the dataset, Q_3 is the third quartile (75th percentile) of the dataset, and X_{norm} is the robust normalized value.

Hybrid normalization is defined by the following algorithm:

Algorithm 3 Hybrid Normalization Algorithm

1: Compute minimum and maximum values for each feature
2: Apply min-max normalization to each feature
3: Compute mean and standard deviation of the dataset
4: Apply z-score normalization to each feature
5: Compute median and interquartile range for each feature
6: Apply robust normalization to each feature
7: Compute average of normalized values for each feature
8: Transform normalized values using a logarithmic function

A.2.4 Feature Selection

Algorithm 4 LSTM Model Post Hyperparameter Tuning

1: Remove correlated columns from the processed dataset using a threshold of 0.8.
2: Drop columns with Mutual Information less than 1.0.
3: Apply Recursive Feature Elimination (RFE) for the optimal number of features with the minimum Mean Squared Error (MSE) score.
4: Apply the feature-selected dataset to LSTM with hyper-tuned parameters.
5: Calculate error metrics: RMSE, MSE and MAE

A.3 Models

A.3.1 LSTM

For a given input sequence $\{x_1, x_2, ..., x_n\}$, $x_t \in \mathbb{R}^{k\times 1}$ represents the input sequence at time t. The memory cell c_t updates information using three gates: the input gate i_t, the forget gate f_t, and the change gate $\tilde{c}_t$. The hidden state h_t undergoes updates through the output gate o_t and the memory cell c_t. At time t, the respective gates and layers compute the following functions:

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i), \tag{A.14}$$
$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f), \tag{A.15}$$
$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o), \tag{A.16}$$
$$\tilde{c}_t = \tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c), \tag{A.17}$$
$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tilde{c}_t, \tag{A.18}$$
$$h_t = o_t \otimes \tanh(c_t), \tag{A.19}$$

where σ and tanh represent the sigmoid and hyperbolic tangent functions respectively. The operator $\otimes$ denotes the element-wise product, and $W \in \mathbb{R}^{d\times k}$, $W_h \in \mathbb{R}^{d\times d}$ are weight matrices, while $b \in \mathbb{R}^{d\times 1}$ represents bias vectors. Additionally, n, k, and d denote the sequence length, the number of features, and the hidden size respectively [29].

A.3.2 ARIMA

1. Autoregressive (AR):

$$Y_t = B_0 + B_1 \cdot Y_{\text{lag1}} + B_2 \cdot Y_{\text{lag2}} + \ldots + B_n \cdot Y_{\text{lag}n} \tag{A.20}$$

where Y is a linear function of its past n values, B_0, B_1, etc. are respective weights, and n is the number of lags.

2. Integrated (I):

$$Y_{\text{forward1}} - Y_t = B_0 + B_1 \cdot (Y_t - Y_{\text{lag1}}) + B_2 \cdot (Y_{\text{lag1}} - Y_{\text{lag2}}) + \ldots \tag{A.21}$$

where Y is a linear function of the past changes in Y.

3. Moving Average (MA):

$$Y_t = B_0 + B_1 \cdot E_{\text{lag1}} + B_2 \cdot E_{\text{lag2}} + \ldots + B_n \cdot E_{\text{lag}n} \tag{A.22}$$

where E represents the error residual.

A.4 Hyperparameter Tuning: Bayestian Optimization

Bayes' theorem is [6]:

$$P(Z|Y) = \frac{P(Y|Z)P(Z)}{P(Y)} \tag{A.23}$$

where $P(Z|Y)$ is the posterior probability of Z given Y, $P(Y|Z)$ is the likelihood of Y given Z, $P(Z)$ is the prior probability of Z, and $P(Y)$ is the marginal probability of Y.

A.5 Results

A.5.1 Denoised vs. Non-Denoised

Table A.17: Error Metric Changes of Haar Wavelets in ARIMA

Error	SAP	Siemens	DT	APPLE	MSFT	ALPHABET	AMBEV	PB	VALE	Average
MSE % Change (%)	30	98	68	74	78	87	-57	67	80	58
MAE % Change (%)	37	90	61	64	67	75	-3	63	72	58
RMSE % Change (%)	17	85	44	49	53	64	-35	43	55	42

Table A.18: ARIMA: Denoised vs Non-Denoised

Stock	Denoised	MSE	MAE	RMSE	Optimal
SAP	YES	4.26151×10^{-5}	0.004128687	0.006528027	DENOISED
	NO	6.1122×10^{-5}	0.00659378	0.007818054	
SIEMENS	YES	1.6999×10^{-6}	0.000824596	0.001303801	DENOISED
	NO	7.55672×10^{-5}	0.008087598	0.008692938	
DT	YES	1.55834×10^{-5}	0.002496667	0.003947578	DENOISED
	NO	4.91003×10^{-5}	0.006456252	0.007007158	
APPLE	YES	8.31514×10^{-7}	0.00057672	0.000911874	DENOISED
	NO	3.1836×10^{-6}	0.001581253	0.001784264	
ALPHABET	YES	1.76385×10^{-5}	0.002656198	0.004199818	DENOISED
	NO	0.00013465	0.010659882	0.011603887	
AMBEV	YES	0.000297703	0.010912426	0.017254061	NON-DENOISED
	NO	0.000127153	0.010536973	0.011276207	
PB	YES	2.80488×10^{-5}	0.003349556	0.005296114	DENOISED
	NO	8.59594×10^{-5}	0.009103696	0.00927143	
VALE	YES	1.0305×10^{-5}	0.002030269	0.003210137	DENOISED
	NO	5.14664×10^{-5}	0.007137682	0.007174007	

Table A.19: Denoising Closing Prices with Haar Wavelets in LSTM

Stock	Denoised	Architecture	MSE	MAE	RMSE	Optimal
SAP	YES	1-Layer	7.476E-06	0.00192	0.00273	DENOISED
	NO	2-Layer	8.683E-06	0.00258	0.00294	
SIEMENS	YES	1-Layer	0.00019	0.01355	0.01390	DENOISED
	NO	2-Layer	3.892E-05	0.00594	0.00623	
DT	YES	1-Layer	9.226E-07	0.00076	0.00096	DENOISED
	NO	2-Layer	1.794E-05	0.00413	0.00423	
APPLE	YES	1-Layer	6.936E-05	0.00817	0.00832	NON-DENOISED
	NO	2-Layer	1.328E-05	0.00302	0.00364	
MSFT	YES	1-Layer	1.830E-06	0.00116	0.00135	DENOISED
	NO	2-Layer	2.113E-07	0.00035	0.00045	
ALPHABET	YES	1-Layer	1.804E-06	0.00097	0.00134	DENOISED
	NO	2-Layer	1.180E-06	0.00093	0.00108	
AMBEV	YES	1-Layer	3.594E-06	0.00176	0.00189	DENOISED
	NO	2-Layer	3.422E-05	0.00348	0.00585	
PB	YES	1-Layer	0.00011	0.00964	0.01052	DENOISED
	NO	2-Layer	1.180E-06	0.00303	0.00349	
VALE	YES	1-Layer	1.171E-06	0.00100	0.00108	DENOISED
	NO	2-Layer	1.576E-05	0.00298	0.00397	

Table A.20: Impact of Haar Wavelets on LSTM Models

	1-LAYER	2-LAYER
MSE % Change (%)	31	32
MAE % Change (%)	27	35
RMSE % Change (%)	26	25

Table A.21: Implied Volatility for Various Stocks

Stock	Implied Volatility
SAP	20.8
SIEMENS	30.33
DT	16.67
APPLE	19.5
ALPHABET	29.7
MSFT	21.1
AMBEV	29.6
PB	30.6
VALE	27.8

Table A.22: Overall Preference between Denoised and Non-Denoised Prices

	SAP	SIEMENS	DT	APPLE	MSFT	ALPHABET	AMBEV	PB	VALE
LSTM	Denoised	Denoised	Denoised	Not-Denoised	Denoised	Denoised	Denoised	Denoised	Denoised
ARIMA	Denoised	Denoised	Denoised	Denoised	Denoised	Denoised	Not-Denoised	Denoised	Denoised

A.5.2 ARIMA vs LSTM

A.5.2.1 Overall

Table A.23: Preferred Models and Architectures

Company	Preferred Model	Architecture
SAP	LSTM	1-Layer
SIEMENS	LSTM	1-Layer
DT	LSTM	2-Layer
APPLE	ARIMA	-
MSFT	LSTM	1-Layer
ALPHABET	LSTM	2-Layer
AMBEV	LSTM	2-Layer
PB	LSTM	2-Layer
VALE	LSTM	2-Layer

Table A.24: % Change in Error from ARIMA to LSTM

Stock	Layers	MSE	MAE	RMSE
SAP	1-Layer	82.47	53.37	58.13
	2-Layer	79.62	37.29	54.86
SIEMENS	1-Layer	45.72	7.33	26.33
	2-Layer	-87.20	-72.77	-64.22
DT	1-Layer	88.42	60.92	65.97
	2-Layer	98.64	85.79	88.35
APPLE	1-Layer	-76.87	-67.28	-51.90
	2-Layer	-29.58	-81.02	-73.92
MSFT	1-Layer	91.68	57.45	71.15
	2-Layer	77.53	51.91	52.60
ALPHABET	1-Layer	68.74	14.62	44.09
	2-Layer	90.65	61.07	69.42
AMBEV	1-Layer	92.38	72.95	72.40
	2-Layer	99.38	94.43	92.15
PB	1-Layer	69.41	31.31	44.69
	2-Layer	96.06	81.67	80.15
VALE	1-Layer	-67.10	-62.94	-42.64
	2-Layer	95.52	78.46	78.82

A.5.2.2 ARIMA

Table A.25: SAP ARIMA: Denoised Price vs. Without Denoised Price

	With Denoised Price	**Without Denoised Price**
MSE	0.00004261513146	0.00006112196931
MAE	0.004128686545	0.006593780329
RMSE	0.006528026613	0.007818054062

Table A.26: Siemens ARIMA: Denoised Price vs. Without Denoised Price

	With Denoised Price	**Without Denoised Price**
MSE	0.000001699895933	0.00007556716641
MAE	0.0008245958849	0.008087598462
RMSE	0.001303800573	0.008692937732

Table A.27: DT ARIMA: Denoised Price vs. Without Denoised Price

	With Denoised Price	**Without Denoised Price**
MSE	0.00001558337118	0.00004910026184
MAE	0.002496667473	0.006456251932
RMSE	0.003947577888	0.0070071579

Table A.28: Apple ARIMA: Denoised Price vs. Without Denoised Price

	With Denoised Price	**Without Denoised Price**
MSE	0.0000008315142624	0.000003183599716
MAE	0.0005767197803	0.001581252963
RMSE	0.0009118740387	0.001784264475

Table A.29: Microsoft ARIMA: Denoised Price vs. Without Denoised Price

	With Denoised Price	**Without Denoised Price**
MSE	0.00001407975777	0.00006446368867
MAE	0.002373163102	0.007125409045
RMSE	0.003752300331	0.008028928239

Table A.30: Alphabet ARIMA: Denoised Price vs. Without Denoised Price

	With Denoised Price	**Without Denoised Price**
MSE	0.00001763847122	0.000134650199
MAE	0.002656198127	0.01065988151
RMSE	0.004199817999	0.01160388724

Table A.31: Ambev ARIMA: Denoised Price vs. Without Denoised Price

	With Denoised Price	**Without Denoised Price**
MSE	0.0002977026074	0.0001271528429
MAE	0.01091242608	0.01053697281
RMSE	0.01725406061	0.01127620694

Table A.32: PB ARIMA: Denoised Price vs. Without Denoised Price

	With Denoised Price	**Without Denoised Price**
MSE	0.00002804882151	0.00008595942211
MAE	0.003349556479	0.009103695961
RMSE	0.005296113812	0.009271430424

Table A.33: Vale ARIMA: Denoised Price vs. Without Denoised Price

	With Denoised Price	**Without Denoised Price**
MSE	0.00001030497774	0.00005146638051
MAE	0.002030268725	0.007137682355
RMSE	0.003210136717	0.007174007284

A.5.2.3 LSTM

Table A.34: SAP LSTM: Denoised Price vs. Without Denoised Price

Parameters	With Denoised Price		Without Denoised Price	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,50)	46	39	21	39
Batch Size: (16,64)	23	30	42	30
LSTM Units Layer 1: (1,50)	30	11	35	18
LSTM Units Layer 2: (1,50)	-	40	-	42
Learning Rate (0.0001,0.1)	0.005029455877	0.006221568681	0.00593541315	0.006221568681
MSE	0.000007472556345	0.000008683910672	0.0001933074919	0.0000389210897
MAE	0.001925248136	0.002588927096	0.0135586313	0.005947172706
RMSE	0.002733597693	0.002946847582	0.01390350646	0.006238676919

Table A.35: Siemens LSTM: Denoised Price vs. Without Denoised Price

Parameters	With Denoised Price		Without Denoised Price	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,50)	36	40	23	48
Batch Size: (16,64)	19	38	21	38
LSTM Units Layer 1: (1,50)	28	22	24	46
LSTM Units Layer 2: (1,50)	-	30	-	29
Learning Rate (0.0001,0.1)	0.006250304902	0.00672125918	0.005138704969	0.001563332653
MSE	0.000000922699387	0.00001794784374	0.00006936904126	0.00001328098588
MAE	0.0007641805097	0.004132330449	0.008174782154	0.003027955223
RMSE	0.0009605724267	0.004236489554	0.008328807914	0.003644308697

Table A.36: DT LSTM: Denoised Price vs. Without Denoised Price

Parameters	With Denoised Price		Without Denoised Price	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,50)	23	40	25	48
Batch Size: (16,64)	27	38	41	39
LSTM Units Layer 1: (1,50)	24	22	47	22
LSTM Units Layer 2: (1,50)	-	30	-	10
Learning Rate (0.0001,0.1)	0.0099080534	0.00672125918	0.004786233648	0.004228092936
MSE	0.000001830653356	0.000000211327985	0.000001804944617	0.000001180825495
MAE	0.001164981662	0.0003547530002	0.0009756539284	0.0009371116908
RMSE	0.001353016392	0.0004597042365	0.001343482273	0.001086657948

Table A.37: Apple LSTM: Denoised Price vs. Without Denoised Price

Parameters	With Denoised Price		Without Denoised Price	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,50)	23	16	23	47
Batch Size: (16,64)	27	22	27	24
LSTM Units Layer 1: (1,50)	12	25	19	15
LSTM Units Layer 2: (1,50)	-	18	-	47
Learning Rate (0.0001,0.1)	0.0099080534	0.006650851645	0.0099080534	0.006591254917
MSE	0.000003594495754	0.0000342286395	0.0001107753769	0.000001180825495
MAE	0.001762475756	0.003481994735	0.009649592986	0.003039173014
RMSE	0.001895915545	0.00585052472	0.01052498821	0.003496953128

Table A.38: Microsoft LSTM: Denoised Price vs. Without Denoised Price

Parameters	With Denoised Price		Without Denoised Price	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,50)	20	43	42	43
Batch Size: (16,64)	43	54	61	54
LSTM Units Layer 1: (1,50)	7	49	48	49
LSTM Units Layer 2: (1,50)	-	29	-	33
Learning Rate (0.0001,0.1)	0.009984890063	0.008170553829	0.004864652618	0.008170553829
MSE	0.000001171706545	0.00001576277204	0.00001082752189	0.000003163903187
MAE	0.001009794992	0.002987978567	0.002955112189	0.001141309096
RMSE	0.001082453946	0.003970235766	0.003290520003	0.001778736402

Table A.39: Alphabet LSTM: Denoised Price vs. Without Denoised Price

Parameters	With Denoised Price		Without Denoised Price	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,50)	36	42	20	34
Batch Size: (16,64)	19	50	43	22
LSTM Units Layer 1: (1,50)	14	50	15	26
LSTM Units Layer 2: (1,50)	-	49	-	39
Learning Rate (0.0001,0.1)	0.006250304902	0.01	0.009984890063	0.004901859378
MSE	0.000005513826854	0.000001649168729	0.00001288571748	0.000002124397674
MAE	0.002267760536	0.001033931379	0.003561130934	0.001324813827
RMSE	0.002348153925	0.001284199645	0.003589668157	0.001457531363

Table A.40: Ambev LSTM: Denoised Price vs. Without Denoised Price

Parameters	With Denoised Price		Without Denoised Price	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,50)	46	47	19	34
Batch Size: (16,64)	23	24	57	22
LSTM Units Layer 1: (1,50)	30	7	11	26
LSTM Units Layer 2: (1,50)	-	46	-	39
Learning Rate (0.0001,0.1)	0.005029455877	0.006591254917	0.006979919024	0.004901859378
MSE	0.0001783314905	0.0000007838948355	0.000009687102888	0.000006404980504
MAE	0.01321143057	0.0005873900616	0.002850207436	0.002131479142
RMSE	0.01335408142	0.0008853783572	0.003112411105	0.002530806295

Table A.41: PB LSTM: Denoised Price vs. Without Denoised Price

Parameters	With Denoised Price		Without Denoised Price	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,50)	46	40	28	36
Batch Size: (16,64)	23	38	48	25
LSTM Units Layer 1: (1,50)	30	22	36	40
LSTM Units Layer 2: (1,50)	-	30	-	27
Learning Rate (0.0001,0.1)	0.005029455877	0.00672125918	0.007705654516	0.008039514753
MSE	0.00002513255569	0.000001105063599	0.000008579516083	0.00001176421152
MAE	0.004921907238	0.0006139866353	0.002300875411	0.003240803879
RMSE	0.005013238044	0.001051220053	0.002929081099	0.003429899637

Table A.42: Vale LSTM: Denoised Price vs. Without Denoised Price

Parameters	With Denoised Price		Without Denoised Price	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,50)	46	40	33	47
Batch Size: (16,64)	23	38	37	24
LSTM Units Layer 1: (1,50)	30	22	48	15
LSTM Units Layer 2: (1,50)	-	30	-	47
Learning Rate (0.0001,0.1)	0.005029455877	0.00672125918	0.004670400526	0.006591254917
MSE	0.00003199016889	0.0000004621437913	0.00003132233982	0.000001459124029
MAE	0.005633601664	0.0004372209117	0.005478112895	0.0009961971507
RMSE	0.005655985227	0.0006798115851	0.005596636474	0.001207942064

A.5.3 Indicator Combinations

A.5.3.1 Basic vs. All Technical

Table A.43: SAP: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	91	95	43	46
Batch Size: (16,128)	32	35	103	52
LSTM Units Layer 1: (1,100)	63	22	34	95
LSTM Units Layer 2: (1,100)	-	93	-	92
Learning Rate (0.0001,0.1)	0.005029455877	0.006591254917	0.00398405268	0.003257767396
MSE	0.0002254634706	0.000203345342	0.0001650429598	0.0001420915395
MAE	0.01082536746	0.01019815727	0.008761841742	0.006969222497
RMSE	0.01501544107	0.01425992083	0.01284690468	0.01192021558

Table A.44: Siemens: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	63	79	19	88
Batch Size: (16,128)	65	68	66	44
LSTM Units Layer 1: (1,100)	-	49	72	26
LSTM Units Layer 2: (1,100)	-	63	-	94
Learning Rate (0.0001,0.1)	0.004670400526	0.00672125918	0.008190478609	0.006048513342
MSE	0.00009028712353	0.0001486165381	0.0001036844487	0.0001219277893
MAE	0.005470248929	0.009694304238	0.006741628784	0.007836494697
RMSE	0.009501953669	0.01219083829	0.0101825561	0.01104209171

Table A.45: DT: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	52	84	79	85
Batch Size: (16,128)	92	105	82	104
LSTM Units Layer 1: (1,100)	-	70	14	99
LSTM Units Layer 2: (1,100)	-	58	-	61
Learning Rate (0.0001,0.1)	0.007705654516	0.003808171911	0.009172798213	0.008170553829
MSE	0.0001578797285	0.00007649484618	0.00008242899913	0.0001311539764
MAE	0.01012087842	0.005030370244	0.005601004223	0.008735279581
RMSE	0.01256502003	0.008746133213	0.009079041752	0.01145224766

Table A.46: Apple: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	83	46	34	28
Batch Size: (16,128)	80	52	79	91
LSTM Units Layer 1: (1,100)	12	95	22	26
LSTM Units Layer 2: (1,100)	-	92	-	79
Learning Rate (0.0001,0.1)	0.004271046986	0.003257767396	0.009984890063	0.005469378971
MSE	0.00004821229808	0.00004082907144	0.00003579098896	0.00004194732434
MAE	0.004823389274	0.003778628917	0.003683492585	0.004134577725
RMSE	0.006943507621	0.006389763019	0.005982557059	0.006476675408

Table A.47: Apple Non-Denoised: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	22	13	85	85
Batch Size: (16,128)	73	107	104	104
LSTM Units Layer 1: (1,100)	79	95	99	99
LSTM Units Layer 2: (1,100)	-	82	-	61
Learning Rate (0.0001,0.1)	0.003398378072	0.009439391405	0.008170553829	0.008170553829
MSE	0.00006498798877	0.0001734881327	0.00005630024251	0.00007873762156
MAE	0.00484207503	0.00966631587	0.004837993353	0.005801111991
RMSE	0.008061512809	0.01317148939	0.007503348753	0.008873422201

Table A.48: Microsoft: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	40	25	52	25
Batch Size: (16,128)	42	84	92	84
LSTM Units Layer 1: (1,100)	31	79	70	79
LSTM Units Layer 2: (1,100)	-	36	-	36
Learning Rate (0.0001,0.1)	0.0099080534	0.004416984292	0.007705654516	0.004416984292
MSE	0.00005436644393	0.00008158082469	0.0003080169581	0.00004305919098
MAE	0.005256529143	0.006339357534	0.01617445417	0.004810292154
RMSE	0.007373360423	0.009032210399	0.01755041191	0.006561950242

Table A.49: Alphabet: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	82	13	82	71
Batch Size: (16,128)	122	82	122	82
LSTM Units Layer 1: (1,100)	97	97	97	87
LSTM Units Layer 2: (1,100)	-	89	-	53
Learning Rate (0.0001,0.1)	0.004864652618	0.01	0.004864652618	0.001856761574
MSE	0.0000344430974	0.0002030986947	0.00006099679156	0.00003117379234
MAE	0.00394826763	0.01237070315	0.006435775063	0.003705556875
RMSE	0.005868824192	0.01425126993	0.007810044274	0.005583349563

Table A.50: Ambev: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	40	46	34	88
Batch Size: (16,128)	42	52	68	92
LSTM Units Layer 1: (1,100)	31	95	82	91
LSTM Units Layer 2: (1,100)	-	92	-	74
Learning Rate (0.0001,0.1)	0.0099080534	0.003257767396	0.003952690677	0.01
MSE	0.0001304507995	0.0001261203105	0.0001144661013	0.0002726815636
MAE	0.007417773121	0.007097851213	0.006234683662	0.01415606997
RMSE	0.01142150601	0.01123032994	0.01069888318	0.01651307251

Table A.51: PB: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	70	65	25	72
Batch Size: (16,128)	25	31	84	107
LSTM Units Layer 1: (1,100)	34	46	79	34
LSTM Units Layer 2: (1,100)	-	75	-	99
Learning Rate (0.0001,0.1)	0.006250304902	0.004901859378	0.004416984292	0.005696325473
MSE	0.0001732749857	0.0000869310047	0.0002019334075	0.000171388856
MAE	0.006126555483	0.006946749122	0.008910487109	0.007433343245
RMSE	0.01316339567	0.009323679783	0.01421032749	0.01309155667

Table A.52: Vale: Basic vs All Technical

Parameters	Basic		Technical	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	52	72	70	46
Batch Size: (16,128)	32	107	25	52
LSTM Units Layer 1: (1,100)	31	34	34	95
LSTM Units Layer 2: (1,100)	-	99	-	92
Learning Rate (0.0001,0.1)	0.008087004258	0.005696325473	0.006250304902	0.003257767396
MSE	0.0005895283449	0.00007027973476	0.00006170765508	0.00006203748217
MAE	0.02112309119	0.005142458737	0.004299691969	0.004583949268
RMSE	0.0242802048	0.008383300946	0.007855422018	0.007876387635

A.5.3.2 Fundamental vs Macroeconomic

Table A.53: SAP: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	53	72	63	95
Batch Size: (16,128)	84	48	33	35
LSTM Units Layer 1: (1,200)	148	40	63	35
LSTM Units Layer 2: (1,200)	-	79	-	93
Learning Rate (0.0001,0.1)	0.004109767655	0.003410835991	0.006416323073	0.006591254917
MSE	0.0001486448824	0.0001357077654	0.0001729001006	0.0001301187644
MAE	0.007885773287	0.007115500045	0.009083799323	0.006900108173
RMSE	0.01219200075	0.0116493676	0.01314914828	0.01140696122

Table A.54: Siemens: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	82	26	83	85
Batch Size: (16,128)	122	76	118	104
LSTM Units Layer 1: (1,200)	193	38	189	99
LSTM Units Layer 2: (1,200)	-	139	-	61
Learning Rate (0.0001,0.1)	0.004864652618	0.00187551336	0.0004373632061	0.008170553829
MSE	0.00010815376	0.0001180580591	0.00008562769987	0.00009998656652
MAE	0.007424649273	0.007449958171	0.006357381261	0.006624377401
RMSE	0.0103997	0.01086545255	0.009253523646	0.009999328303

Table A.55: DT: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	91	84	20	79
Batch Size: (16,128)	32	105	68	68
LSTM Units Layer 1: (1,200)	122	99	91	93
LSTM Units Layer 2: (1,200)	-	58	-	123
Learning Rate (0.0001,0.1)	0.005029455877	0.003808171911	0.000832179885	0.00672125918
MSE	0.00009578152463	0.00007403838517	0.0001653786208	0.00007661034543
MAE	0.006921946726	0.005237031073	0.009816220779	0.005541886442
RMSE	0.009786803596	0.00860455607	0.01285996193	0.008752733598

Table A.56: Apple: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	83	46	82	72
Batch Size: (16,128)	118	52	122	107
LSTM Units Layer 1: (1,200)	189	189	97	34
LSTM Units Layer 2: (1,200)	-	184	-	99
Learning Rate (0.0001,0.1)	0.0004373632061	0.003257767396	0.004864652618	0.005696325473
MSE	0.00003037041063	0.00003122097906	0.0000400828757	0.00006067920972
MAE	0.00355905956	0.003314423136	0.004052825196	0.006144814798
RMSE	0.005510935549	0.005587573629	0.00633110383	0.007789686112

Table A.57: Apple Non-Denoised: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	46	46	52	72
Batch Size: (16,128)	52	52	92	107
LSTM Units Layer 1: (1,200)	189	189	70	34
LSTM Units Layer 2: (1,200)	-	184	-	99
Learning Rate (0.0001,0.1)	0.003257767396	0.003257767396	0.007705654516	0.005696325473
MSE	0.00008229705151	0.00008318944066	0.0001223137771	0.00005880482755
MAE	0.007101952318	0.006502626895	0.009348385852	0.004974122716
RMSE	0.009071772237	0.00912082456	0.01105955592	0.007668430579

Table A.58: Microsoft: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	22	96	52	46
Batch Size: (16,128)	50	67	92	52
LSTM Units Layer 1: (1,200)	90	182	70	189
LSTM Units Layer 2: (1,200)	-	104	-	84
Learning Rate (0.0001,0.1)	0.005907364435	0.001563332653	0.007705654516	0.003257767396
MSE	0.0000438936717	0.0000286471216	0.00003680814546	0.00002870274165
MAE	0.004952238502	0.003232820617	0.00409633605	0.003385060752
RMSE	0.006625229936	0.00535230059	0.006066971688	0.005357493971

Table A.59: Alphabet: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	91	96	19	100
Batch Size: (16,128)	32	67	66	71
LSTM Units Layer 1: (1,200)	63	91	72	58
LSTM Units Layer 2: (1,200)	-	54	-	100
Learning Rate (0.0001,0.1)	0.005029455877	0.001563332653	0.008190478609	0.01
MSE	0.000228203778	0.00002577705572	0.0000563152189	0.00003041861251
MAE	0.01427304896	0.003176135218	0.005999675724	0.003726890995
RMSE	0.01510641513	0.005077110962	0.007504346667	0.005515307109

Table A.60: Ambev: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	83	97	82	79
Batch Size: (16,128)	80	67	122	68
LSTM Units Layer 1: (1,200)	12	70	97	93
LSTM Units Layer 2: (1,200)	-	13	-	123
Learning Rate (0.0001,0.1)	0.004271046986	0.004228092936	0.004864652618	0.00672125918
MSE	0.0001743636205	0.0001404311164	0.0001320326584	0.0001141807798
MAE	0.009750218324	0.008311707887	0.008053371678	0.006472238803
RMSE	0.01320468177	0.01185036355	0.01149054648	0.01068554069

Table A.61: PB: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	19	96	23	87
Batch Size: (16,128)	66	67	49	107
LSTM Units Layer 1: (1,200)	72	182	47	97
LSTM Units Layer 2: (1,200)	-	104	-	59
Learning Rate (0.0001,0.1)	0.008190478609	0.001563332653	0.000322535413	0.001538240544
MSE	0.0002489988965	0.0002059188178	0.0002203336426	0.000166441997
MAE	0.01080260525	0.01020244736	0.009873076252	0.008351527834
RMSE	0.01577969887	0.0143498717	0.0148436398	0.01290124014

Table A.62: Vale: Fundamental vs Macroeconomic

Parameters	Technical + Fundamental		Technical + Macroeconomic	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	19	96	23	87
Batch Size: (16,128)	66	67	49	107
LSTM Units Layer 1: (1,200)	72	182	47	97
LSTM Units Layer 2: (1,200)	-	104	-	59
Learning Rate (0.0001,0.1)	0.008190478609	0.001563332653	0.000322535413	0.001538240544
MSE	0.0002489988965	0.0002059188178	0.0002203336426	0.000166441997
MAE	0.01080260525	0.01020244736	0.009873076252	0.008351527834
RMSE	0.01577969887	0.0143498717	0.0148436398	0.01290124014

A.5.3.3 Exchange Rates vs Commodities

Table A.63: SAP: Exchange Rates vs Commodities

Parameters	Technical + Exchange Rates		Technical + Commodities	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	41	76	34	99
Batch Size: (16,128)	44	49	79	121
LSTM Units Layer 1: (1,200)	57	49	36	191
LSTM Units Layer 2: (1,200)	-	162	-	97
Learning Rate (0.0001,0.1)	0.009509746227	0.006221568681	0.009984890063	0.008553955247
MSE	0.00017594848	0.0001624395097	0.0001556641712	0.0004533019513
MAE	0.009200337851	0.008413955942	0.00824495211	0.01863918512
RMSE	0.01326455728	0.01274517594	0.01247654484	0.02129088893

Table A.64: Siemens: Exchange Rates vs Commodities

Parameters	Technical + Exchange Rates		Technical + Commodities	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	19	79	52	85
Batch Size: (16,128)	66	49	92	104
LSTM Units Layer 1: (1,200)	142	40	137	198
LSTM Units Layer 2: (1,200)	-	154	-	119
Learning Rate (0.0001,0.1)	0.008190478609	0.005683094956	0.007705654516	0.008170553829
MSE	0.0001314932882	0.00009604135452	0.0002620210918	0.0001720690262
MAE	0.008402873659	0.00616533373	0.01377030226	0.01076896589
RMSE	0.01146705229	0.009800069108	0.01618706557	0.01311750839

Table A.65: DT: Exchange Rates vs Commodities

Parameters	Technical + Exchange Rates		Technical + Commodities	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	25	47	25	14
Batch Size: (16,128)	92	93	92	110
LSTM Units Layer 1: (1,200)	101	194	101	175
LSTM Units Layer 2: (1,200)	-	184	-	138
Learning Rate (0.0001,0.1)	0.008660386322	0.0005078625771	0.008660386322	0.002116466822
MSE	0.0004025856728	0.0001188514793	0.0001717035089	0.0001128326186
MAE	0.01851182791	0.008343194855	0.009541390813	0.006894690794
RMSE	0.02006453769	0.01090190255	0.01310356855	0.01062226994

Table A.66: Apple: Exchange Rates vs Commodities

Parameters	Technical + Exchange Rates		Technical + Commodities	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	82	46	27	14
Batch Size: (16,128)	122	52	75	20
LSTM Units Layer 1: (1,200)	193	189	149	16
LSTM Units Layer 2: (1,200)	-	84	-	53
Learning Rate (0.0001,0.1)	0.004864652618	0.003257767396	0.007137677013	0.001366812065
MSE	0.00003538859808	0.00003167040319	0.04533885625	0.00004954204006
MAE	0.003726355945	0.003072021225	0.176465075	0.004718679492
RMSE	0.005948831657	0.005627646327	0.2129292283	0.007038610663

Table A.67: Microsoft: Exchange Rates vs Commodities

Parameters	Technical + Exchange Rates		Technical + Commodities	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	85	42	63	12
Batch Size: (16,128)	104	23	65	32
LSTM Units Layer 1: (1,200)	198	99	193	55
LSTM Units Layer 2: (1,200)	-	44	-	74
Learning Rate (0.0001,0.1)	0.008170553829	0.003678891514	0.004670400526	0.004260222473
MSE	0.00005944549255	0.001414285506	0.00009003919391	0.00005619190192
MAE	0.006016923273	0.0362708985	0.007825870423	0.005264472219
RMSE	0.007710090308	0.03760698747	0.009488898456	0.007496125794

Table A.68: Alphabet: Exchange Rates vs Commodities

Parameters	Technical + Exchange Rates		Technical + Commodities	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**
Epochs: (10,100)	40	96	84	14
Batch Size: (16,128)	42	67	104	109
LSTM Units Layer 1: (1,200)	55	182	198	192
LSTM Units Layer 2: (1,200)	-	104	-	163
Learning Rate (0.0001,0.1)	0.0099080534	0.001563332653	0.008170553829	0.005788801438
MSE	0.00004736855486	0.0000411442181	0.00007232743293	0.00005311502743
MAE	0.005384891025	0.004992676932	0.0071410389	0.005519603062
RMSE	0.006882481737	0.006414375893	0.008504553658	0.007288005724

Table A.69: Ambev: Exchange Rates vs Commodities

Parameters	Technical + Exchange Rates		Technical + Commodities	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	26	49	23	72
Batch Size: (16,128)	76	49	49	107
LSTM Units Layer 1: (1,200)	38	162	89	62
LSTM Units Layer 2: (1,200)	-	-	-	199
Learning Rate (0.0001,0.1)	0.00187551336	0.006221568681	0.000322535413	0.005696325473
MSE	0.0004827573976	0.0002110150184	0.0001550848807	0.0001961984262
MAE	0.01983066161	0.01166555084	0.008757569929	0.01123948948
RMSE	0.02197174089	0.01452635599	0.01245330802	0.01400708486

Table A.70: PB: Exchange Rates vs Commodities

Parameters	Technical + Exchange Rates		Technical + Commodities	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	21	96	63	71
Batch Size: (16,128)	21	67	65	82
LSTM Units Layer 1: (1,200)	53	182	193	174
LSTM Units Layer 2: (1,200)	-	104	-	101
Learning Rate (0.0001,0.1)	0.0004304932786	0.001563332653	0.004670400526	0.001856761574
MSE	0.0002049076053	0.0003092494792	0.000333023233	0.0001525004583
MAE	0.009054667802	0.0145408286	0.01429086253	0.00747803419
RMSE	0.01431459414	0.01758549059	0.01824892416	0.01234910759

Table A.71: Vale: Exchange Rates vs Commodities

Parameters	Technical + Exchange Rates		Technical + Commodities	
	Single Layer	Two Layers	Single Layer	Two Layers
Epochs: (10,100)	46	46	82	87
Batch Size: (16,128)	52	52	122	31
LSTM Units Layer 1: (1,200)	189	189	193	31
LSTM Units Layer 2: (1,200)	-	184	-	182
Learning Rate (0.0001,0.1)	0.003257767396	0.003257767396	0.004864652618	0.004450424797
MSE	0.00005826611698	0.00006255989239	0.00006405232299	0.00006089058849
MAE	0.004392818797	0.004637506876	0.005107072766	0.004711928092
RMSE	0.007633224547	0.007909481171	0.008003269519	0.007803242178

A.5.4 Feature Selection

Table A.72: Decrease in Dataset Size by Feature Selection Method

Method	SAP	SIEMENS	DT	APPLE	MSFT	ALPHABET	AMBEV	PB	VALE
C	125	125	125	122	122	123	121	122	118
MI	115	113	115	112	111	112	108	110	109
RFE	57	56	28	28	27	84	54	55	27

Table A.73: SAP: Feature Selection

Parameters	**No**		**C + MI**		**C + MI + RFE (0.5)**	
	Single Layer	**Two Layers**	**Single Layer**	**Two Layers**	**Single Layer**	**Two Layers**
Epochs (10,100)	70	94	70	85	83	96
Batch Size (16,128)	25	18	25	104	118	67
LSTM Units Layer 1 (1,200)	62	87	20	49	189	91
LSTM Units Layer 2 (1,200)	-	46	-	33	-	54
Learning Rate (0.0001,0.1)	0.00625	0.00532	0.00625	0.00817	0.00817	0.00156
MSE	0.00016	0.00014	0.00034	0.00019	0.00024	0.00014
MAE	0.00875	0.00733	0.01367	0.00965	0.01186	0.00730
RMSE	0.01269	0.01171	0.01856	0.01390	0.01558	0.01175

Table A.74: Siemens: Feature Selection

Parameters	**No**		**Correlation + MI**		**Correlation + MI + RFE (0.5)**	
	Single Layer	Two Layers	Single Layer	Two Layers	Single Layer	Two Layers
Epochs (10,100)	85	85	82	36	25	95
Batch Size (16,128)	104	104	122	50	84	35
LSTM Units Layer 1 (1,200)	99	99	97	96	79	22
LSTM Units Layer 2 (1,200)	-	61	-	40	-	93
Learning Rate (0.0001,0.1)	0.00817	0.00817	0.00486	0.01	0.00442	0.00659
MSE	0.000109	0.000180	0.000116	0.000162	0.000232	0.000087
MAE	0.00727	0.01102	0.00762	0.00957	0.01198	0.00587
RMSE	0.01046	0.01343	0.01079	0.01274	0.01522	0.00935

Table A.75: DT: Feature Selection

Parameters	**No**		**Correlation + MI**		**Correlation + MI + RFE (0.25)**	
	Single Layer	Two Layers	Single Layer	Two Layers	Single Layer	Two Layers
Epochs (10,100)	91	47	59	13	20	25
Batch Size (16,128)	32	93	60	107	68	84
LSTM Units Layer 1 (1,200)	122	97	99	95	91	79
LSTM Units Layer 2 (1,200)	-	94	-	82	-	36
Learning Rate (0.0001,0.1)	0.00503	0.00051	0.00806	0.00944	0.00083	0.00442
MSE	0.000121	0.000129	0.000117	0.000255	0.0001	0.000127
MAE	0.00829	0.00851	0.00791	0.01174	0.00657	0.00782
RMSE	0.011	0.01135	0.01081	0.01597	0.01001	0.01128

Table A.76: Apple: Feature Selection

Parameters	**No**		**Correlation + MI**		**Correlation + MI + RFE (0.25)**	
	Single Layer	Two Layers	Single Layer	Two Layers	Single Layer	Two Layers
Epochs (10,100)	91	51	80	30	19	97
Batch Size (16,128)	32	46	28	77	66	69
LSTM Units Layer 1 (1,200)	122	24	48	37	72	38
LSTM Units Layer 2 (1,200)	-	181	-	25	-	11
Learning Rate (0.0001,0.1)	0.00503	0.00127	0.00521	0.00593	0.00819	0.00423
MSE	0.0000402	0.0000327	0.0000424	0.0000466	0.0000401	0.0000345
MAE	0.00422	0.00358	0.00415	0.00447	0.00402	0.00357
RMSE	0.00634	0.00571	0.00651	0.00683	0.00633	0.00587

Table A.77: Apple Non-Denoised: Feature Selection

Parameters	No		Correlation + MI		Correlation + MI + RFE (0.25)	
	Single Layer	Two Layers	Single Layer	Two Layers	Single Layer	Two Layers
Epochs (10,100)	82	25	52	25	25	42
Batch Size (16,128)	122	84	92	84	92	23
LSTM Units Layer 1 (1,200)	193	156	70	79	53	52
LSTM Units Layer 2 (1,200)	-	66	-	36	-	26
Learning Rate (0.0001,0.1)	0.00486	0.00442	0.00771	0.00442	0.00866	0.00368
MSE	0.000242	0.000108	0.000080	0.000103	0.000085	0.000185
MAE	0.01379	0.00770	0.00604	0.00718	0.00633	0.01122
RMSE	0.01556	0.01038	0.00893	0.01016	0.00925	0.01362

Table A.78: Microsoft: Feature Selection

Parameters	No		Correlation + MI		Correlation + MI + RFE (0.25)	
	Single Layer	Two Layers	Single Layer	Two Layers	Single Layer	Two Layers
Epochs (10,100)	63	14	52	72	25	64
Batch Size (16,128)	65	97	63	107	84	31
LSTM Units Layer 1 (1,200)	193	97	46	20	40	27
LSTM Units Layer 2 (1,200)	-	73	-	49	-	37
Learning Rate (0.0001,0.1)	0.00467	0.00640	0.00670	0.00570	0.00442	0.00571
MSE	0.000041	0.000124	0.000052	0.000058	0.000041	0.000035
MAE	0.00462	0.00883	0.00540	0.00557	0.00428	0.00369
RMSE	0.00639	0.01114	0.00724	0.00759	0.00642	0.00592

Table A.79: Alphabet: Feature Selection

Parameters	ALL INFO NO PRE-PROCESSING		Correlation + MI		Correlation + MI + RFE (0.75)	
	Single Layer	Two Layers	Single Layer	Two Layers	Single Layer	Two Layers
Epochs (10,100)	85	21	85	25	56	96
Batch Size (16,128)	88	116	104	84	57	67
LSTM Units Layer 1 (1,200)	193	177	49	40	95	91
LSTM Units Layer 2 (1,200)	-	147	-	21	-	54
Learning Rate (0.0001,0.1)	0.00392	0.00848	0.00817	0.00442	0.00690	0.00156
MSE	0.000073	0.000053	0.000037	0.000090	0.000036	0.000094
MAE	0.00721	0.00530	0.00440	0.00755	0.00412	0.00808
RMSE	0.00852	0.00726	0.00611	0.00950	0.00602	0.00971

Table A.80: Ambev: Feature Selection

Parameters	No		Correlation + MI		Correlation + MI + RFE (0.25)	
	Single Layer	Two Layers	Single Layer	Two Layers	Single Layer	Two Layers
Epochs (10,100)	23	65	82	97	19	46
Batch Size (16,128)	49	31	122	69	66	52
LSTM Units Layer 1 (1,200)	89	46	97	22	72	95
LSTM Units Layer 2 (1,200)	-	75	-	10	-	92
Learning Rate (0.0001,0.1)	0.00032	0.00490	0.00486	0.00423	0.00819	0.00326
MSE	0.000163	0.000114	0.000140	0.000132	0.000166	0.000133
MAE	0.00961	0.00680	0.00869	0.00800	0.00956	0.00770
RMSE	0.01278	0.01065	0.01182	0.01147	0.01289	0.01153

Table A.81: PB: Feature Selection

Parameters	No		Correlation + MI		Correlation + MI + RFE (0.5)	
	Single Layer	Two Layers	Single Layer	Two Layers	Single Layer	Two Layers
Epochs (10,100)	23	31	53	79	52	79
Batch Size (16,128)	49	126	84	59	92	68
LSTM Units Layer 1 (1,200)	89	137	75	21	70	49
LSTM Units Layer 2 (1,200)	-	128	-	43	-	63
Learning Rate (0.0001,0.1)	0.00032	0.00090	0.00411	0.00830	0.00771	0.00672
MSE	0.000226	0.000904	0.000251	0.000202	0.005345	0.000186
MAE	0.00993	0.00915	0.01148	0.00874	0.07146	0.00796
RMSE	0.01505	0.01402	0.01585	0.01421	0.07311	0.01364

Table A.82: Vale: Feature Selection

Parameters	No		Correlation + MI		Correlation + MI + RFE (0.25)	
	Single Layer	Two Layers	Single Layer	Two Layers	Single Layer	Two Layers
Epochs (10,100)	63	25	83	25	53	71
Batch Size (16,128)	65	84	80	84	84	82
LSTM Units Layer 1 (1,200)	193	156	12	40	75	87
LSTM Units Layer 2 (1,200)	-	66	-	21	-	53
Learning Rate (0.0001,0.1)	0.00467	0.00442	0.00427	0.00442	0.00411	0.00186
MSE	0.000062	0.000085	0.000070	0.000091	0.000060	0.000064
MAE	0.00502	0.00633	0.00560	0.00658	0.00477	0.00510
RMSE	0.00790	0.00922	0.00835	0.00953	0.00775	0.00802

A.5.4.1 RFE Indicator Output, post C+MI

Table A.83: SAP: RFE Indicators

Close_lag	EMA	TP	Coconut oil
Sawnwood, Cameroon	Crude oil, average	Sawnwood, Malaysian	Lead
Logs, Cameroon	Gold	Fish meal	Palm oil
Tobacco, US import u.v.	Tin	Soybeans	Silver
Logs, Malaysian	High	Liquefied natural gas, Japan	Natural gas index
Rapeseed oil	Maize	Low	Rice, Thai 25%
Open	Shrimps, Mexican	Adj Close	Urea
Economic Activity, Industrial Production, Index	Groundnut oil **	Beef	SMA_5
Tea, Kolkata	Tea, Colombo	Rubber, TSR20 **	Cocoa
Cotton, A Index	Gross Profit	UpperBand	Net Change In Investments - Total

Table A.84: Siemens: RFE Indicators

'Close_lag'	'EMA'	'TP'	'Lead'
'Platinum'	'Gold'	'Crude oil, average'	'Copper'
'Rice, Thai A.1'	'Fish meal'	'Tobacco, US import u.v.'	'Rapeseed oil'
'Maize'	'Coal, South African **'	'Liquefied natural gas, Japan'	'High'
'Urea'	'Groundnut oil **'	'Low'	'Rice, Thai 25%'
'Shrimps, Mexican'	'Economic Activity, Industrial Production, Index'	'Beef'	'Lamb **'
'Rubber, RSS3'	'Open'	'Rubber, TSR20 **'	'Tea, Kolkata'
'Tea, Colombo'	'Tea, Mombasa'	'Cocoa'	'Phosphate rock'
'SMA_5'	'Operating Cash Flow Per Share'	'Total Long-Term Assets'	'Operating Margin'
'Chicken'	'Tea, avg 3 auctions'	'Orange'	'Adj Close'
'Banana, US'	'UpperBand'	'RollingMean'	'sma'
'LowerBand'	'Sugar, US'	'Banana, Europe'	'USD/RUPEE'
'USD/REAL'	'USD/CNY'	'USD/JPY'	'RUPEE/REAL'
'RUPEE/EURO'	'USD/EURO'	'USD/GBP'	'GBP/USD'

Table A.85: DT: RFE Indicators

'Close_lag'	'EMA'	'Gold'	'TP'
'Shrimps, Mexican'	'High'	'Low'	'Tea, Colombo'
'Tea, Kolkata'	'Cocoa'	'Open'	'SMA_5'
'Tea, avg 3 auctions'	'Adj Close'	'UpperBand'	'sma'
'RollingMean'	'LowerBand'	'USD/RUPEE'	'USD/REAL'
'USD/JPY'	'USD/CNY'	'USD/EURO'	'RUPEE/REAL'
'RUPEE/EURO'	'USD/GBP'	'GBP/USD'	'RSI'

Table A.86: Apple: RFE Indicators

Close_lag	EMA	TP	Low
Adj Close	High	Open	Prices, Consumer Price Index, All items, Index
SMA_5	Palm kernel oil	RollingMean	UpperBand
sma	LowerBand	Beef	Rubber, RSS3
Tea, avg 3 auctions	USD/RUPEE	USD/REAL	USD/CNY
USD/JPY	RUPEE/REAL	USD/EURO	USD/GBP
GBP/USD	RSI	Volume	MACDs_12_26_9

Table A.87: MSFT: RFE Indicators

'Close_lag'	'EMA'	'TP'	'High'
'Low'	'Adj Close'	'Tin'	'Sawnwood, Cameroon'
'Natural gas index'	'Logs, Cameroon'	'Copper'	'Open'
'Logs, Malaysian'	'SMA_5'	'RollingMean'	'UpperBand'
'sma'	'LowerBand'	'Tea, Kolkata'	'USD/RUPEE'
'USD/REAL'	'USD/CNY'	'USD/JPY'	'RUPEE/REAL'
'USD/GBP'	'USD/EURO'	'GBP/USD'	

Table A.88: Alphabet: RFE Indicators

'Close_lag'	'EMA'	'TP'	'Adj Close'
'Low'	'US CPI'	'High'	'Prices, Consumer Price Index, All items, Index'
'Zinc'	'Lead'	'Coconut oil'	'Crude oil, average'
'Palm oil'	'Crude oil, WTI'	'Soybean oil'	'Soybeans'
'Tin'	'Tobacco, US import u.v.'	'Logs, Cameroon'	'Wheat, US HRW'
'Crude oil, Dubai'	'Fish meal'	'Copper'	'Rapeseed oil'
'Natural gas index'	'Coal, South African **'	'Rice, Thai A.1'	'Logs, Malaysian'
'Silver'	'DAP'	'Maize'	'Coal, Australian'
'Liquefied natural gas, Japan'	'Platinum'	'Open'	'Economic Activity, Industrial Production, Index'
'SMA_5'	'Natural gas, Europe'	'Urea'	'Rice, Thai 25%'
'Shrimps, Mexican'	'Groundnut oil **'	'TSP'	'RollingMean'
'Lamb **'	'sma'	'UpperBand'	'Groundnuts'
'Beef'	'LowerBand'	'Rubber, RSS3'	'Rubber, TSR20 **'
'Tea, Kolkata'	'Tea, Colombo'	'Tea, Mombasa'	'Cocoa'
'Cotton, A Index'	'Coffee, Robusta'	'SG&A Expenses'	'Phosphate rock'
'Basic Shares Outstanding'	'Total Non-Cash Items'	'Net Cash Flow'	'ROI - Return On Investment'
'Cash Flow From Investing Activities'	'ROE - Return On Equity'	'Income Taxes'	'Chicken'
'Tea, avg 3 auctions'	'Orange'	'Banana, US'	'Sugar, US'
'Sugar, world'	'USD/RUPEE'	'Banana, Europe'	'USD/REAL'
'USD/CNY'	'USD/JPY'	'RUPEE/REAL'	'USD/GBP'
'USD/EURO'	'GBP/USD'	'RSI'	

Table A.89: Ambev: RFE Indicator

Close_lag	EMA	Prices, Consumer Price Index, All items, Index	Gold
Tobacco, US import u.v.	Fish meal	Sawnwood, Cameroon	Aluminum
Zinc	TP	Coconut oil	Rice, Thai A.1
Soybean meal	Soybeans	Logs, Malaysian	Wheat, US HRW
Sunflower oil	Rapeseed oil	Natural gas index	Maize
Coal, South African **	Urea	Low	High
Beef	Lamb **	Groundnuts	Economic Activity, Industrial Production, Index
Tea, Kolkata	Tea, Colombo	Tea, Mombasa	Cocoa
Cotton, A Index	Open	Operating Cash Flow Per Share	SMA_5
Chicken	Tea, avg 3 auctions	RollingMean	UpperBand
Orange	sma	LowerBand	Adj Close
Sugar, US	USD/RUPEE	USD/REAL	USD/CNY
USD/JPY	RUPEE/EURO	USD/EURO	USD/GBP
GBP/USD			

Table A.90: PB: RFE Indicators

'Close_lag'	'EMA'	'Fish meal'	'Nickel'
'Aluminum'	'Plywood'	'Crude oil, WTI'	'Zinc'
'Logs, Cameroon'	'Gold'	'Soybean oil'	'Palm oil'
'Tobacco, US import u.v.'	'TP'	'Wheat, US HRW'	'Coal, South African **'
'Rapeseed oil'	'Sunflower oil'	'Rice, Thai 25%'	'Shrimps, Mexican'
'High'	'Natural gas, US'	'Low'	'Urea'
'Lamb **'	'Beef'	'Economic Activity, Industrial Production, Index'	'Open'
'Tea, Kolkata'	'Tea, Colombo'	'Cocoa'	'Operating Income'
'Net Cash Flow'	'Other Long-Term Assets'	'SMA_5'	'Total Non-Cash Items'
'Net Change In Investments - Total'	'Tea, avg 3 auctions'	'Orange'	'Banana, US'
'Adj Close'	'RollingMean'	'LowerBand'	'sma'
'Sugar, US'	'Days Sales In Receivables'	'USD/RUPEE'	'USD/REAL'
'USD/CNY'	'USD/JPY'	'RUPEE/REAL'	'RUPEE/EURO'
'USD/GBP'	'GBP/USD'		

Table A.91: Vale: RFE Indicators

'Close_lag'	'EMA'	'TP'	'Iron ore, cfr spot'
'Copper'	'Silver'	'Crude oil, Dubai'	'Low'
'High'	'Open'	'Beef'	'Economic Activity, Industrial Production, Index'
'Lamb **'	'SMA_5'	'sma'	'RollingMean'
'UpperBand'	'LowerBand'	'Adj Close'	'USD/RUPEE'
'USD/REAL'	'USD/CNY'	'USD/JPY'	'RUPEE/REAL'
'USD/EURO'	'USD/GBP'	'RSI'	

A.5.5 Overall

Table A.92: Overall Breakdown of Feature Selection and Best Indicator Combination

Stock	**Feature Selection**			**Indicator**		**BIC**	**O**
SAP	NO	-	2-L	T+M	2-L	T+M	LSTM (D Close)
SIEMENS	C+MI+RFE	50%	2-L	T+M	2-L	T+M	LSTM (D Close)
DT	C+MI+RFE	25%	1-L	T+F	2-L	T+F	LSTM (D Close)
APPLE	NO	-	2-L	T+F	2-L	T+F	ARIMA (ND Close)
MSFT	C+MI+RFE	25%	2-L	T+F	2-L	T+F	LSTM (D Close)
ALPHABET	C+MI+RFE	75%	1-L	T+F	2-L	T+F	LSTM (D Close)
AMBEV	NO	-	2-L	T+M	1-L	T+M+F	LSTM (D Close)
PB	C+MI+RFE	50%	2-L	BASIC	2-L	BASIC	LSTM (D Close)
VALE	C+MI+RFE	25%	1-L	T+F	2-L	T+F	LSTM (D Close)

Table A.93: Error Metrics for Optimal and Best Indicator Combination Models (Part 1)

	SAP		SIEMENS		DT		APPLE		MSFT	
	O	BIC	O	BIC	O	BIC	O	BIC	O	BIC
MSE	0.0000749	0.0077940	0.0004365	0.1279885	0.0001381	0.0001590	0.0003262	0.4716335	0.0000408	0.0066210
MAE	0.0086175	0.0831536	0.0166319	0.3326662	0.0076264	0.0093436	0.0173134	0.6787502	0.0055382	0.0765252
RMSE	0.0086549	0.0882837	0.5362446	0.3577548	0.4214191	0.0126081	0.0180617	0.6867558	0.0063901	0.0813695

Table A.94: Error Metrics for Optimal and Best Indicator Combination Models (Part 2)

	ALPHABET		AMBEV		PB		VALE		AVERAGE		
	O	BIC	O	BIC	O	BIC	O	BIC	O	BIC	Total
MSE	0.0021559	0.0070647	0.0000008	0.0116178	0.0000222	0.0029495	0.0004043	0.0026637	0.0004000	0.0709435	0.0356717
MAE	0.0410120	0.0782546	0.0004723	0.1043358	0.0044374	0.0397485	0.0142660	0.0424493	0.0128794	0.1605808	0.0867301
RMSE	0.0464315	0.0840515	0.0009134	0.1077858	0.0047148	0.0543095	0.0201065	0.0516114	0.1181041	0.1693922	0.1437481