Vaasan yliopisto
UNIVERSITY OF VAASA

Hamid Hosseininesaz

Application of a Hybrid Model of Markov Switching and Machine Learning for Predicting Stock Prices

School of Technology and Innovations
Master's thesis in Industrial Systems Analytics

Vaasa 2024

UNIVERSITY OF VAASA
School of Technology and Innovations
Author: Hamid Hosseininesaz
Title of the Thesis: Application of a Hybrid Model of Markov Switching and Machine Learning for Predicting Stock Prices
Degree: Master of Science
Programme: Industrial Systems Analytics
Supervisor: Ahm Shamsuzzoha
Year: 2024 **Pages:** 67

ABSTRACT:
The stock market is considered one of the most attractive markets for investors, as there is no limit to the amount of capital that can be invested. The most important goal of investors in the stock market is to make a profit. For this reason, knowing the behavior and predicting the movement of the stock price plays an important role in making profits. Therefore, many researchers have tried to develop models to accurately predict stock prices. However, due to the complexity of the stock market and the influence of various factors on the stock price, there is still no model that can accurately predict the stock price. The stock price has two linear and non-linear components. The linear component shows the trend of the stock price, and the non-linear component shows the price fluctuations, which can be due to emotions, expectations, news, etc. For example, the linear component can indicate the overall trend of a stock's price increase or decrease, while the non-linear component indicates momentary and unexpected fluctuations and changes caused by external factors.
Due to the complexity and rapid changes in the stock market, using a single model to predict prices may be insufficient (Pan, 2010). Therefore, this study presents a hybrid model to predict the trend and price fluctuations. The proposed hybrid model is a combination of the Markov Switching (MS) model for predicting stock trends and the Long Short-Term Memory (LSTM) model for predicting price fluctuations. The MS model can recognize changes in different market regimes, while the LSTM model is able to learn complex and long-term patterns in data due to its special structure in neural networks.
In this study, the historical data of the stock prices of five companies (Digia, Kone B, Nokia, UPM Kymmene and Wartsila) from January 2019 to December 2023 were used. This data includes daily stock prices. The use of historical data allows us to evaluate the performance of the models under real market conditions. To verify the performance of the proposed model, we compared it with MS and LSTM models. For this purpose, the three criteria Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE) and R-Squared (R^2) were used. The results show that the MS model has the best performance in predicting stock prices compared to the other models and the proposed model has the worst performance.
To increase the robustness and validity of the results, the models were compared again in a weekly time frame. For this purpose, the historical data of the stock prices of the listed companies in the period from January 2000 to May 2024 were used. The results of comparing the performance of the models showed that the proposed model outperformed other models in predicting stock prices for all criteria for all stocks. This could be due to the greater volatility in the weekly time frame than in the daily time frame. Therefore, the proposed model is more efficient for markets with higher volatility, such as the cryptocurrency market.

KEYWORDS: Price Prediction, Markov Switching, LSTM, Hybrid Model

Contents

Figures

Tables

1 1 Introduction

Financial markets are one of the most important pillars of the economy due to their role in the creation and expansion of productive investment. The stock exchange is one of the components of financial markets, and its state usually indicates the state of the economy. The stock exchange helps with investment by mobilizing private savings, making them liquid, reducing information and search costs, and providing an appropriate market for investment (Sebastian 1989). On the other hand, due to the position and importance of the stock exchange in a country's economy, the forecasting of other important capital market variables, the examination of the effects of and on such important factors as stock prices and returns on economic variables appears to be inevitable and is useful in helping economic policymakers in their decision making (Kuwornu & Owusu-Nantwi 2011).

Capital markets are a large part of the economy and depend on and impact the development and well-being of countries and the state of the economy. Today, policymakers have realized the role of these markets, thus turning their attention to policymaking. Therefore, the analysis of the impact of economic factors on capital markets has gained more attention in the recent past.

1.1 Study background

In every society, many individuals seek to increase their well-being and gain benefits, and the return on investment reflects the profits derived from that investment (Wijesinghe & Rathnayaka 2020). Consequently, every investor considers the primary factor in their decision-making is the return, and investors seek the most profitable opportunities to invest their surplus resources in the capital markets. People mainly invest in the stock market to make a profit, which requires accurate information about the market, stock fluctuations, and forecasting future trends (Kabir et al. 2023). Therefore, investors need powerful and reliable tools to predict stock prices. Hence, the best method for investors to forecast the future of stocks in the capital market is paramount to maximizing their profits.

Stock price data are time series data, and one of the critical issues in the field of time series is the forecasting of financial time series, which has attracted significant attention. Financial time series forecasting addresses investment issues and economic decision-making, providing a profitable model for the future based on past economic results. The forecasting process typically involves historical information and its generalization to the future using mathematical models. Since forecasting future events plays a crucial role in the decision-making process, forecasting is essential for many organizations, institutions, investors, governments, and others. Forecasting can be considered a useful tool for short-term and long-term planning. Forecasting stock prices and recognizing their behavior is a complex endeavor. Various factors, such as political events, the general state of the economy, and investor expectations, influence the stock market. Many recent studies have shown that the stock market is a non-linear, chaotic, non-scaling, and non-parametric system (Gupta et al., 2021) that depends on numerous political, economic, and psychological factors. Therefore, studying this system appears to be very challenging. Since the importance of forecasting economic variables, different models like analyses, regressions, and time series have been created to model the variables and predict them. However, the current research findings indicate that artificial intelligence (AI) methods can capture non-linear relationships between variables with less error. Because capital market variables are nonstationary, these models have been applied in the recent past to forecast stock returns in stock markets (Hossain et al. 2018, Vijh et al. 2020).

1.2 Study aim

The objective of this study is to propose a novel hybrid model to enhance the accuracy of stock price predictions. Predicting stock prices is a key interest for investors in the stock market, as improved prediction accuracy can lead to increased investment profits. This topic has been a focal point in the financial sector. Time series models are highly effective at capturing the linear aspects of stock price movements, while AI models excel at identifying nonlinear features (Pai 2005). In this study, we leverage the strengths of both approaches. Specifically, we employ a combination of Markov Switching model and LSTM to improve the precision of stock price predictions. Consequently, this study aims to address the following question:

Question 1: How does the integration of Markov Switching and LSTM models improve the accuracy of stock price predictions compared to using single models?

1.3 Study method

The study employs quantitative approaches to analyze the results. By reviewing previous research, the appropriate models—Markov Switching (MS) and Long Short-Term Memory (LSTM)—were identified as suitable for this study. These models were then implemented and solved using Eviews software and MATLAB. To thoroughly test the proposed hybrid model, two distinct timeframes were utilized: a daily timeframe spanning from January 2019 to December 2023, and a longer timeframe extending from January 2000 to May 2024. This dual timeframe approach allows for a comprehensive evaluation of the model's performance across different market conditions and periods.

1.4 Study structure

This research work is made up of five chapters. In the second chapter, first the efficient market hypothesis is introduced, and then the research on stock price prediction is discussed. The research gap is established from previous studies, and a model to address the research question is developed. The methods used in the proposed model are discussed in the third chapter. In the fourth chapter, the data collected in relation to the Finnish stock market stock prices are analyzed using the methods described in chapter three, and the results of the model are presented. In the fifth chapter, the study's findings are summarized, conclusions and recommendations are made, research limitations are discussed, and suggestions for future research are given. Finally, the list of sources and references was provided.

2 Literature Review

2.1 Introduction

One of the essential and fundamental questions of capital market investors is predicting the future price of stocks. Stock price prediction is among the most challenging tasks for financial investors worldwide due to the uncertainty and volatility of stock prices in the market (Hossain et al. 2018). Forecasting assesses unknown situations, predicts future events, and turns experiences into predictions as a key element in management decisions. An analysis of the consequences of a decision and the possible events following the decision is essential, given the importance of decision-making. For this reason, management systems require forecasting to design and control their organizational functions (Armstrong, 2001). There are four methods for forecasting stock prices (Van Eyden, 1996).

- Technical approach
- Fundamental approach
- Time series forecasting approach
- Intelligent approach (machine learning)

2.2 Efficient Market Hypothesis

The Efficient Market Hypothesis is a theory that holds that the current stock market prices contain all the available information and that it is impossible to outperform the market (Fama, 1970).

If the capital market is efficient, the price of the stock will be equal to its fair value, and savings will be channeled to the most profitable investment opportunities (Naseer & Bin , 2015). This is so because it is helpful to the investors and the economy of the country of concern. The efficient market hypothesis has been a significant part of finance and economics and is based on the efficient processing of information by market participants. Efficiency is a term used to refer to a market that has information about the stocks reflected in the price (Fama, 1970). In the capital market context, market efficiency can be defined as the level of achievement of the right price for stocks. Efficiency of the market implies that the price always incorporates new information and information is a body of knowledge about companies and their stocks. Prices in an efficient market should reflect this information. Thus, it is possible to conclude that efficiency in the processing of information can be the basis for claiming the efficiency of a market. The prices at any one time are accurate given the information available; thus, the prices will contain all the information available. Therefore, the efficiency of the capital market has always been in question since the very start (Naseer & Bin, 2015).

The efficiency of the capital market pays special attention to two important aspects in determining prices: velocity and accuracy of price formation (Santoso & Ikhsan 2020). If a market is inefficient in terms of the speed and accuracy of price discovery, then informed traders can make large profits for themselves. If current prices contain such valuable information, it will be extremely challenging to identify stocks that are cheap and will produce high returns or identify expensive stocks that will produce low returns (Allen et al, 2011). Accurately predicting the future is the only way to have a good investment. The efficient market hypothesis assumes that stock prices quickly adjust to new information, and the current price fully reflects all information.

In 1970, Fama provided the first practical definition of an efficient market. He called a market efficient if the expected abnormal returns from various strategies based on the information

2.3 Threefold Classification of Informational Efficiency

As defined, an efficient market is one where all information is immediately and fully incorporated into the stock prices. Thus, information is the measure of the effectiveness of the capital market. In a perfectly efficient market, the price of stocks adjusts to all available information, and investors cannot make any extra profits from the available information since it is already reflected in the prices (Naseer & Bin, 2015). In an efficient market, the price of each security is equal to the true value of the security and contains all information about the future prospects of the security. If some types of information are not fully incorporated into prices or if the incorporation of information into prices takes time, then the efficiency of the market is less than perfect (Mishkin & Eakins 2006). However, it is impossible to state that a market is perfectly efficient or perfectly inefficient (Degutis & Novickytė 2014). Efficiency is not absolute, and every market is efficient to a certain level. Thus, the efficiency of the capital market can be categorized based on certain information and the issue of whether the average investor can earn abnormal returns by employing certain information to trade in stocks. Market efficiency can be examined with respect to three sets of information; according to Fama's theory (1970), these three categories of information are:

- Information related to past stock prices or historical information
- Published public information
- All available information, especially inside and confidential information

Based on this information, the market is examined for efficiency at three levels:

- Weak form
- Semi-strong form
- Strong form

The strong form of efficiency encompasses the lower levels. For example, if a market is efficient in the strong form, it means that it is also efficient in the weak and semi-strong forms. Similarly, if a market is not efficient in the weak form, it will not be efficient in the other levels.

2.3.1 Weak Form of Efficiency of what?

The weak form of capital market efficiency holds that the current stock price represents the fair value of the respective stock and is a past image of the stock price. As per weak efficiency, the information set contains only historical price information like prices, trading volume, and other related market information. This form of the hypothesis particularly asserts that one cannot get further information through technical analysis to determine the up or down movements of the prices of assets. In other words, in this form of efficiency, all information about the asset's past is incorporated in the current price, and therefore, technical analysis will not be of any more help in making profits. However, in this form, prices do not reflect the true value, and fundamental analysis may bring profit. The weak form of the efficient market hypothesis can be tested by gathering historical price information and analyzing the ability to forecast returns by searching for patterns in the data (Couillard & Davison, 2005).

2.3.2 Semi-Strong Form of Efficiency of what??

In the semi-strong form of capital market efficiency, it is assumed that the current stock price incorporates all the available and accessible information. This information is not limited to past information and includes a set of annual reports, income reports, announcements and advertisements, economic and political news, and other public information. In this form of efficiency, we say that a market is efficient if stock prices are fast enough to incorporate new information released to the public. In such a market, the consumers of information may have divergent perceptions of the value of data, but the price will always reflect the best possible analysis of the information (Malkiel, 2011). This form of efficiency postulates that earning above-average returns through public information is impossible. Therefore, at this level of efficiency, both technical and fundamental analysis is not helpful for making profits (Malkiel, 2011).

Analyzing the semi-strong form is associated with a certain vagueness owing to the joint hypothesis problem. This issue emerges because to test the semi-strong form of efficiency, one must identify the right effect of fundamentals. To determine whether prices are aligned with the economic fundamentals or not, one must have a model that explains the relationship between the economic fundamentals and the prices of the assets. While there are models suggested in this regard, nobody knows that these models define this relationship in a convincing way from the empirical perspective (Zhu & Zhu, 2013).

More specifically, the joint hypothesis means that one should consider two hypotheses to test for semi-strong efficiency. The first hypothesis is about the structural model that sets the equilibrium prices, and the second hypothesis is about the agents' efficiency in pricing (Shleifer 2000, P.2). Thus, if an empirical test is carried out based on an incorrect and incomplete equilibrium model or a model that is not available to the public, then it may lead to the rejection of the efficient market hypothesis.

In this regard, Fama (1976) notes that the rejection of the efficient market hypothesis should not be viewed as conclusive evidence of market inefficiency but rather as a signal that the rejection may be due to the incorrect specification of the market model. The application of the weak form does not generally have such a problem, as testing the weak form of efficiency does not involve the formulation of a specific equilibrium pricing structure.

2.3.3 Strong Form of Efficiency

The last level of market efficiency is where the current prices incorporate all information, past price information, public information and private information. The strong form of efficiency means that not even inside and confidential market information can provide an edge to one investor over the other (Fama, 1970). In this form of market efficiency, if an event happens in a company, one cannot argue that only the CEO is aware of it and others are not. In this regard, the stock price of the company in question immediately responds to this event because there should not be any insider information. If the capital market is not highly efficient, then an individual with capital and more information than others and with experts to analyze this specific information will get higher returns. However, the level of such abuse is not as high in large stock exchanges as in other markets (Malkiel, 2011). This is the highest level of capital market efficiency, and even the management or insiders of the company cannot earn higher returns because they possess inside information. In other words, all the public and hidden information is incorporated into the market prices, and no group of investors can earn more than the normal rate of return. This extreme form of efficiency is mainly seen as a theoretical concept, and its applicability in the real world appears improbable. According to Grossman & Stiglitz (1980), when information is costly, which means it is not freely available, the market cannot be perfectly efficient. Thus, any reasonable model of market equilibrium must allow for investor incentives to acquire information. Thus, it is necessary to admit the existence of some, albeit minor, market imperfections (Jones and Netter, 2008; Dimson and Mussavian, 1998).

Fama (1991) further builds on his previous statements by noting that the efficient market hypothesis assumption is that the cost of reflecting information and transaction costs is zero. A more general definition of efficiency could be that prices incorporate information in the sense that the marginal gain from information is equal to the marginal cost of acquiring it (Jones and Netter, 2008). Shaker (2013) examined market efficiency at a weak level in the Finnish and Swedish stock markets. To this end, the study analyzed the

OMX Stockholm 30 and OMX Helsinki 25 indices for the years 2003-2012. The findings of the study revealed that the weak form of market efficiency does not hold in the stock markets of these countries, which is in line with previous studies, including Jennergren and Korsvold (1974), Berglund et al. (1983), Frennberg, and Hansson (1993), and Metghalchi et al. (2008). Hence, due to inefficiency at the weak level in the Finnish stock market, it is possible to forecast the stock prices with the help of mathematical models.

2.4 Time series approach.

The data obtained from the observation of a phenomenon over time form a time series. In fact, a time series is a sequence of observed values of a financial or physical variable at equal time intervals, represented as discrete values (Box et al. 2015). Time series models for short-term forecasting attempt to explain the behavior of a variable based on its past values (and possibly past values of other variables, which are wished to predict). In addition, time series models can provide accurate predictions for the desired variable, even if the economic model of the infrastructure is unclear. Moreover, time series models do not require information related to economic theories and only use information related to statistical data. A univariate time-series model relates only the current value of a variable to its past value and its past and current error value. Univariate time-series models include autoregressive processes, moving average processes, autoregressive moving average processes, and autoregressive integrated moving average processes. Multivariate time series models, such as the vector auto-regressive model, attempt to explain a variable's behavior based on its past values and several variables simultaneously. In the following, several related studies are examined.

Mondal et al. (2014) focused on stock price prediction in the Indian market using the autoregressive integrated moving average (ARIMA) model. For this purpose, various listed companies, including 56 companies from different sectors, were examined. The results showed that the ARIMA model performs well in predicting stock prices, with over 85% prediction accuracy for all sectors. According to this study, prediction results are better in some sectors than others, such as the fast-moving. The consumer goods sector performs best, but the banking sector performs less than others. Ariyo et al. (2014) evaluated the performance of the ARIMA model on the New York Stock Exchange (NYSE) and

Nigeria Stock Exchange (NSE) markets. In this study, Nokia stocks were selected from the NYSE market, and their historical data from April 1995 to February 2011 were used. The Zenith Bank Index was selected from the NSE market, and historical data was collected from January 2006 to February 2011. The results showed that the model has significant potential for short-term prediction of stock prices and can help in decision-making in the financial markets. Afeef et al. (2018) used an ARIMA model to predict the stock price of the company Oil & Gas Development Company Limited in the Pakistan stock market. The historical data from 2004 to 2018 were examined, and their results showed that the model can predict short-term periods. Ganesan & Kannan (2021) specifically discussed the short-term forecasting ability of the ARIMA model, and their results from studying three sectors of the NSE from 2018 to 2021 showed that the ARIMA model efficiently forecasts short-term periods and keeps up well with rising prediction techniques in short-term periods, but their study does not mention any models or compare them to other methods. Khanderwal & Mohanty (2021) used the Auto Regressive Integrated Moving Average (ARIMA) model to predict stock prices. They developed predictive models using stock data from ICICI Bank, sourced from Yahoo Finance, covering the period from 30.03.2020 to 17.04.2020. Their experimental results concluded that the ARIMA model can reasonably predict stock prices over a short time frame.

Many studies have focused on comparing models' performance under different criteria. Mahadik et al. (2021) investigated the performance of long short-term memory (LSTM) and ARIMA models. Accordingly, the historical data from two NSE TATA and ADANI PORTS datasets were utilized. This study investigated the effect of features such as preprocessing steps and automatic correlation. The results revealed that both models have an accuracy of over 90%, but LSTM performs better on larger datasets, and the ARIMA model requires more processing time. However, ARIMA has higher accuracy when all dataset features contain legitimate values. Kobiela et al. (2022) compared the performance of ARIMA and LSTM models in predicting the average daily and monthly prices of companies listed on the NASDAQ stock exchange using the mean square error (MSE) and mean absolute percentage error (MAPE) criteria. Thus, a model was found to provide better results regarding input data, parameters, and number of features. The results indicated that the ARIMA model performs better than the LSTM model when historical price values are used as input to predict more than one period. In other words, ARIMA performs better for more extended data periods. Meanwhile, the performance of the LSTM model deteriorates as the data periods increase. The comparison of the errors showed that the ARIMA model performs 1.8 times better than the LSTM model for the 30-day forecast, 10.2 times better for the three-month average, and 3.4 times better for the 9-month average.

2.5 Machine learning approach

Using linear and nonlinear patterns in market data enables intelligent systems to discover the patterns in future time series. The prediction of future stock price movements with AI models is more accurate and faster than time series models since there are no limiting assumptions. An artificial neural network (ANN) is a nonlinear forecasting model that can estimate any function and model processes with unknown behavior. An ANN is an information-processing idea inspired by the biological nervous system, which processes information like the brain. The key element of this information-processing idea is the new structure of the information-processing system consisting of many interconnected processing elements that work together to solve a problem.

ANNs can learn, generalize, parallel information processing, and fault tolerance, leading ANN to play a significant role in solving complex problems (Wang et al. 2011). The ability of the ANN to learn and generalize data with nonlinear processes is very suitable. In addition, the ANN can establish the relationship between inputs and outputs with a higher prediction accuracy than statistical methods (Vui et al. 2013). In other words, the ANN does not need to know the nature of the relationship between dependent and independent variables and can be replaced in cases where appropriate mathematical relationships between dependent and independent variables are not formed.

Devadoss & Ligori (2013) used the ANN model on the Indian stock market to predict the price for the next five days. The developed model consisted of an input layer, a hidden layer, and an output layer, where the inputs were the opening, high, low, closing price, and volume. The results showed that the model's predictive power increases when the number of inputs is three compared to five. Therefore, properly selecting parameters is essential to obtain a model with low error. De Oliveira et al. (2013) focused on developing a neural model to predict short-term stock closing prices in the Brazilian stock exchange BM&FBOVESPA. This study aimed to incorporate historical series variables into mathematical models or computer algorithms to estimate expected price fluctuations. The applied method combines technical, fundamental, and time series analysis to predict the price behavior and the percentage of correct price series predictions (POCID) predictions. In this research, the Petrobras stock PETR4 was used as a case study, and it was found that a window size (the number of points in the time series used as input to the neural network) of three worked best, and the POCID index predicting the correct direction was 93.62% for the test set and 87.50% for the validation set. Moreover, the MPE error value was 5.42%, and the root mean square error (RMSE) error was negligible. Dutta et al. (2006) presented two types of ANN models named ANN1 and ANN2 and compared them for the Indian stock market. The model ANN1 used inputs such as weekly closing value, moving average, and oscillators, while ANN2 uses inputs such as weekly closing value, moving average, and volatility. The networks were trained with 250 weeks of data from 1997 to 2003 and then predicted the weekly closing prices for two years. The results of this study showed that the ANN1 model achieved an RMSE of 4.82% and an Mean Absolute Error (MAE) of 3.93%, while ANN2 achieved an RMSE of 6.87% and an MAE of 5.52%, so the ANN1 model performed better.

Devadoss and Ligori (2013) and Dutta et al. (2006) have shown that determining the optimal parameters significantly affected the performance of the neural network model. Khan et al. (2011) utilized the backpropagation algorithm for training and a multilayer feedforward network as a price prediction network to increase the predictive power of the ANN model. Applying the proposed approach improved the model's predictive power in this study. Göçken et al. (2016) investigated the market's technical indicators to create a model for predicting stock prices in the Turkish market. The harmony search (HS) and genetic (GA) algorithms were also applied to select the most essential technical indicators within the framework of an ANN model. The researchers avoided overfitting and underfitting ANN models by correctly selecting neurons for the hidden layers. There were four criteria to evaluate the proposed model: loss functions, return from investment analysis, buy and hold analysis, and graphical analysis. The results showed that the HS-based ANN model is highly efficient for forecasting markets based on the proposed performance criteria. Senapati et al. (2018) applied the particle swarm optimization (PSO) algorithm to adjust the model parameters to increase the predictive power of the ANN model. Comparing the proposed model with the CMS-PSO and Bayes ANN models on the Bombay Stock Market showed that the proposed model performs better regarding the MAPE criterion.

Many studies have compared neural network models with different models in terms of their performance. Wijaya et al. (2010) compared an ARIMA model with an ANN one for Indonesian markets and showed that the ANN model leads to smaller errors than the ARIMA model. Ma (2020) compared three ANN, LSTM, and ARIMA models to predict stock prices and showed that the LSTM model offers interesting possibilities for stock price prediction while ARIMA and ANN have their strengths and weaknesses. Even though the LSTM model provides the best predictive ability, data processing significantly affects its performance. In addition, the ANN model performs better than the ARIMA model regarding prediction accuracy. Meanwhile, Islam & Nguyen (2020) compared the performance of three ARIMA and ANN models and the stochastics process-geometric Brownian motion in predicting next-day prices. For this purpose, the historical data of the S&P500 index was used. The results showed that the ARIMA geometric Brownian motion performed better than the ANN model, and the ARIMA and Brownian motion stochastic models performed almost the same. Esfahanipour & Mardani (2011) presented an ANFIS model for predicting stock prices. The applied model was developed using data from the Tehran Stock Exchange from 2001 to 2010 with a multilayer perceptron (MLP) ANN model and adaptive network-based fuzzy inference system (ANFIS) via network partitioning and C-Mean fuzzy clustering and their comparison. The results showed that the proposed model performed better compared to other models. Billah et al. (2015) compared the ANN model with the ANFIS model for predicting Bangladesh's stock market data and found that the ANFIS model was a more efficient technique. Khare et al. (2017) examined the feedforward neural network (FFNN) and recurrent neural network (RNN) in the New York market. Examining ten stocks showed that the feedforward network performs better in short-term price prediction. Ho et al. (2021) compared three models, ARIMA, ANN, and LSTM, to predict the closing price of stocks. Using data from the Malaysian Stock Exchange from 2019 to 2020 and applying the RMSE and Mean Absolute Percentage Error (MAPE) criteria showed that the LSTM model achieves more than 90% accuracy in stock prediction. Thus, machine learning techniques have potential in stock price prediction, especially in times of high volatility, such as during the coronavirus epidemic.

2.6 Markov switching

Many parameters undergo dramatic changes in behavior due to events, which can be observed in macroeconomics and financial time series over sufficiently long periods. Such changes in the time series process can occur due to events such as war, financial crises, and major changes in government policy. Model instability is often defined as a change in an equation from one regime to another. In most studies, there is little information about the times of changing parameters, so it is necessary to produce results about the turning point after which the change in parameters is significant. When a process changes in the past, these changes may also occur in the future, and this aspect should be considered in the predictions. Furthermore, regime change is a random and exogenous variable and should not be regarded as a predictable and final matter.

Sattayatham et al. (2012) used the Markov switching generalized autoregressive conditional heteroskedasticity (MS-GARCH) model to predict the volatility and price of the SET50 index, which allows for different dynamics of volatility based on unobserved regimes. This study aimed to compare the performance of MS-GARCH and GARCH models in predicting stock prices. The results showed that MS-GARCH models generally have a higher potential in predicting the index price and volatility under different distributions than GARCH models. Nguyen (2017) utilized the hidden Markov model (HMM) to predict the daily stock prices of Apple, Google, and Facebook. Researchers used the Akaike information criterion (AIC) and Bayesian information criterion (BIC) criteria to determine the number of optimal modes in the HMM and proposed three optimal categories: two modes, three modes, and four modes, respectively. The results showed that the model with two modes performs better than those with three and four modes. In addition, the active traders who used HMM achieved higher returns than naive forecasts. Wang et al. (2022) predicted the volatility of renewable energy stocks using GARCH-MIDAS-MS models for short and long periods. GARCH-MIDAS-MS models performed better than others by combining regime shifts in short- and long-term volatility components. This study indicated that appropriate predictive models that include short-term, long-term, or both terms in switching regimes are needed to understand the dynamic behavior of the renewable energy stock market over time. Furthermore, this study suggested that governments should adopt a combination of short-term and long-term measures that consider the different roles of regime change to varying horizons in predicting fluctuations in renewable energy reserves. Gopinathan et al. (2024) introduced an HMM-Gaussian mixture (GMM) model to predict the next day's stock price. The MAPE criterion was used to compare the introduced model with HMM and ANN models following the training. This study revealed that the proposed model performs better than the HMM model in 66 out of 72 cases and better than the ANN model in 24 out of 36 cases in predicting the next day's stock price. Banerjee & Mukherjee (2022) predicted the short-term stock prices of five stocks in the NSE index using five neural network models: MLP, LSTM, gated recurrent unit, BLSTM, and gated bidirectional recurrent unit. The results showed that the MLP network performed much better than the others.

2.7 Hybrid Model

It has been confirmed that the models do not work well under all conditions. In other words, each model has its strengths and weaknesses, and the use of combined models can compensate for these weaknesses and lead to predictions with fewer errors. For this reason, the development of hybrid models has been very rapid in recent years. The use of hybrid models based on neural networks and time series models has been extensively studied regarding their strengths and weaknesses.

Khashei & Hajirahimi (2020) applied ARIMA and multilayer perceptron (MLPs) models for stock price's linear and nonlinear properties. As noted, the hybrid model was not compared with each model individually; rather, the combination of the models was favored. Thus, the ARIMA-LPMS model was compared with the LPMS-ARIMA model. The first model predicted the linear pattern of the price time series and then the nonlinear part, while the second model predicted the reverse. The comparison of the results based on different criteria indicated that the LPMS-ARIMA model performs 16.5%, 2.57%, 2.57%, and 3.71% better than the ARIMA-LPMS model for the criteria RMSE, MAPE, MSE, and MAE, respectively. Merh et al. (2010) presented a combined model of ANN and ARIMA to predict the value and trend of SENSEX, BSE IT, BSE Oil & Gas, and BSE 100 indicators. The model inputs included the daily high, low, open, and close prices, and the model's output was the next day's close price. The ARIMA-ANN and ANN-ARIMA models were compared, showing that the ARIMA-ANN model performed better than the ANN-ARIMA model. Khuat & Le (2017) combined the fuzzy logic model with the ANN model to obtain a model with high predictability. The fuzzy logic as a method to conclude from vague, ambiguous, or imprecise information and the ANN model with a high ability to learn and recognize nonlinear variables can be combined to create a model that achieves a better hybrid model by using the capabilities of both models. The historical data of three companies, Google, Apple, and Yahoo were analyzed from 2011 to 2015 to verify the performance of the proposed model. Using the MAPE, MAE, and RMSE criteria showed that the model performed well in predicting the companies' stock prices. Hossain et al. (2018) proposed a hybrid model using LSTM and gated recurrent units (GRU) to solve problems based on random stock price movements. The researchers used S&P 500 time series data. According to this study, the proposed model, which considers the random nature of stock price changes, had fewer errors based on the RMSE criterion, which is particularly suitable for large data sets. Yu & Yan (2020) suggested using a deep neural networks (DNN) model as a possible solution to effectively deal with the nonlinear nature of financial data. A phase-space reconstruction time series phase reconstruction method was used in the LSTM model along with the DNN model to reconstruct the time series. The proposed model was compared with ARIMA, SVR, MLE, and LSTM models using various criteria such as Pearson correlation coefficient, MAPE, RMSE, and directional accuracy. In all cases, the proposed model performed better. Vijh et al. (2020) used a hybrid ANN and forest model to investigate the problem of predicting the stock market arising from the volatile and nonlinear nature. The financial data of five companies, including the opening, high, low, and closing prices, were evaluated to create a new variable that served as input to the model. The performance of the proposed model was examined using the RMSE and MAPE criteria. According to the results, the proposed model can predict the closing price the next day due to the low values of these criteria. Zhu & Zhu (2013) presented a Bayesian regime-switching combination model to show regime-switching and parameter uncertainty. The results of this study indicated that the combined predictions of regime switching were consistently better than the historical average and the model predictions proposed by Rapach et al. (2010) This study also provided insights into the economic drivers of return predictability, which are valuable for investors trying to understand and predict stock market movements in the context of economic conditions. Bildirici & Ersin (2014) integrated a series of MS-GARCH models with ANNs to improve the predictive power using two objectives. The first objective was to present a class of GARCH models that combines the characteristics of asymmetric performance in MS-GARCH processes, and the second objective was to benefit from the global approximation properties by combining them with artificial networks. Therefore, MS-ARMA-FIGARCH, APGARCH, and FIAPGARCH models were combined with neural network models, RNN, and Hybrid NN. The results were evaluated based on three criteria: MAE, MSE, and RMSE, and it was revealed that the ANN-MS-ARMA-FIGARCH model performed the best. Chung & Shin (2018) combined the LSTM with a GA to predict stock prices using price data such as highest and lowest price, opening price, and closing price, as well as indicators such as moving average and stochastic direction. The MAE, MSE, and MAPE showed that the GA-LSTM algorithm has an acceptable prediction accuracy. Machová & Vochozka (2019) compared the MLP network with the primary radial function network using data from 2006 to 2018 of Unipetrol in the Paraguayan stock market. In this study, the MLP achieved better results than the RBF. Chen & Zhou (2020) utilized LSTM and GA to predict stock prices, including market factors, stock fundamentals, and CSI 300 datasets. The Construction Bank of China data were examined, and the results indicated that the combined GA-LSTM model performs better than the individual models. Kumar Chandar (2021) presented a hybrid model to predict stock prices using the parameter optimization of the Elman neural network model. The daily data of eight companies, IOS, MSFT, HAL, ORCL, GS, CTSH, BAC, and AAPLE, were used from 2009 to 2018.

The results exhibited that the proposed model has better prediction accuracy for the next day than the CSO-ARMA, PSO-Elman, PSO-MLP, FPA-ELM, and GA models. Ji et al. (2021) predicted the stock price of ASM on the Australian stock market using a hybrid model. Combining the LSTM model with the PSO algorithm showed that the proposed model performs better in predicting stock prices than the SVM-PSO and LSTM models.

2.8 Conclusion

In Table 1 below, the four methods discussed in the literature review are compared, highlighting the advantages and disadvantages of each.

Table 1: summary of methods

Method	Advantages	Disadvantages
ARIMA	Good at capturing linear patterns	Limited in capturing non-linear relationships
	Well-established and widely used in time series forecasting	Not suitable for long-term forecasting
ANN	Excellent at capturing non-linear relationships	Requires a large amount of data for training
	Can model complex patterns and interactions	Can overfit the training data if not properly regulated
Markov Switching	Effective at modeling regimes and structural changes in time series	Requires assumptions about the number of states and transitions
	Useful for identifying different market conditions	
Hybrid Markov Switching-LSTM	Combines the strengths of Markov Switching and LSTM for better accuracy	High computational time due to the complexity of the model
	Better performance in terms of accuracy compared to single models	More complex to implement and tune compared to individual models
	Effective at capturing both linear and non-linear relationships	

According to the literature review, the LSTM neural network model is effective in predicting non-linear variables. Additionally, the Markov switching model, utilizing the Markov chain characteristic, can better capture regimes in time series data, thereby enhancing predictive power. Therefore, it is recommended to use both LSTM and Markov switching models for predicting stock prices.

Additionally, in this study, we propose a novel approach to combining these models. Previous studies typically used machine learning methods to capture the nonlinearity of stock price movements and then combined these with the linear components. Our approach, however, differs. Instead of predicting the nonlinearity of stock price movements directly, we divide the movement into trend and residual error components. This innovative method treats the residual error as the nonlinear part of stock price movement, offering a new perspective in predicting stock price trends.

3 Methodology

In this chapter, we provide a detailed explanation of the structures of the Markov Switching (MS) model and the Long Short-Term Memory (LSTM) neural network, which are integral components of our proposed hybrid model. We then outline the steps involved in predicting stock prices using this hybrid model. These steps include data preprocessing, model training for both the MS and LSTM components, integrating the outputs of these models, and evaluating the hybrid model's performance.

3.1 Markov Regime-Switching Model

Most variables comprise sections where the series behavior changes remarkably. In other words, each macroeconomic parameter or financial data encounters numerous failures over a long period. Wars, a general panic in the financial market, or significant changes in government policies may result in such evident changes in time series. (Hamilton; 1989)
Notably, a process that has changed in the past may experience changes in the future. This point needs to be considered in predictions. Moreover, regime changes are not predictable and definite; rather, it is a random and exogenous parameter.
In Markov-switching models, the time series process under consideration is a function of an unobserved random variable (S_t), with S_t being the regime or state where the time series process is situated at time t. S_t also takes integer values, and the Markov chain is the simplest time series model for a discrete random value (Hamilton, 1989).

3.1.1 Markov chain

S_t is a random variable that takes only integer values. Imagine that S_t equals a specific value j depending on the previous period's value (Hamilton, 1989):

$$P\{S_t = j \mid S_{t-1} = i,\ S_{t-2} = k,\ \ldots,\ S_{t-n} = n\} = P\{S_t = j \mid S_{t-1} = i\} = P_{ij} \quad \text{Eq (1)}$$

In that case, such a process is a Markov chain with n regimes with transition probabilities P_{ij}, where P_{ij} represents the probability of transitioning from regime i to regime j (Hamilton, 1989).

$$P = \begin{bmatrix} P_{11} & P_{12} & \dots & P_{1n} \\ P_{21} & P_{22} & \dots & P_{2n} \\ . & . & \dots & . \\ . & . & \dots & . \\ P_{n1} & P_{n2} & \dots & P_{nn} \end{bmatrix} \quad \text{Eq (2)}$$

The element in row j and column i of the matrix n with the probability P_{ij} presents the probability that after regime i, regime j will occur. For example, P_{21} is in the second row and first column and presents the probability of transitioning from regime 2 to regime 1 (Hamilton, 1989).

3.1.2 Representing a Markov Chain with a Vector Autoregressive Process

A representation of the Markov chain is such that a N*1 vector called E_t is considered if $S_t=j$, with its element j being one, and otherwise, zero. If $S_t=1$, the first element E_t is one, others are zero, and E_t is the first column of an identity matrix (I_n).

$$E_t = \begin{cases} (1,0,0,\dots,0)' \rightarrow S_t = 1 \\ (0,1,0,\dots,0)' \rightarrow S_t = 2 \\ \\ (0,0,0,\dots,1)' \rightarrow S_t = n \end{cases} \quad \text{Eq (3)}$$

If $S_t=i$, the j^{th} element of E_{t+1} is a random variable with the value 1 and probability P_{ij}. This random variable has an expectation P_{ij}, and if $S_t=1$, the conditional expectation of E_{t+1} is:

$$E\left(E_{t+1} \mid s_t = i\right) = \begin{bmatrix} p_{i1} \\ p_{i2} \\ \\ p_{in} \end{bmatrix} \quad \text{Eq (4)}$$

Which is the i^{th} column of matrix P. Moreover, when $S_t= i$, E_t equals the i^{th} column of the identity matrix, leading to:

$$E\left(E_{t+1} \mid E_t\right) = PE_t \quad \text{Eq (5)}$$

The primary Markov theorem states that the situation in period $t+1$ is dependent on the situation in period t. So, we have:

$$E\left(E_{t+1} \mid E_t, ...\right) = PE_t \qquad \text{Eq (6)}$$

Thus, the Markov chain can be written as (Hamilton, 1989):

$$E_{t+1} = PE_t + V_{t+1} \Rightarrow V_{t+1} = E_{t+1} - E\left(E_{t+1} \mid E_t, ...\right) \qquad \text{Eq (7)}$$

This is an autoregression vector for E_t so that V_t takes limited values and is zero on average. We can also have (Hamilton: 1989):

$$E_{t+m} = V_{t+m} + PV_{t+m}P^2V_{t+m-2} + ... + P^{m-1}V_{t+1} + P^mE_t \Rightarrow E\left(E_{t+m} \mid E_t, E_{t=1}...\right) = P^mE_t \qquad \text{Eq (8)}$$

If $S_{t+m}=j$, the j^{th} element of E_{t+m} *matrix* is 1, and otherwise, it is zero. That is, the j^{th} element of $E\left(E_{t+m} \mid E_t, E_{t=1}...\right)$ shows the probability that $S_{t+m}=j$, considering the regime in period t. For example, if the process is in regime i at period t:

$$\begin{bmatrix} P\left\{S_{t+m} = 1 \mid S_t = i\right\} = P_{i1} \\ P\left\{S_{t+m} = 1 \mid S_t = i\right\} = P_{i2} \\ P\left\{S_{t+m} = 1 \mid S_t = i\right\} = P_{in} \end{bmatrix} = P^m e_i \qquad \text{Eq (9)}$$

where, e_i presents the i^{th} column of the identity matrix I_n, suggesting that the m-step-ahead transition probabilities for Markov chain can be calculated by multiplying P by itself *m* times. $P\left\{S_{t+m} = 1 \mid S_t = i\right\}$ is the probability that the i^{th} regime in m period is followed by the regime j, found in the j^{th} element of matrix P_m (Hamilton: 1989).

3.1.3 Reducible Markov chain

For a Markov chain with two regimes, the transition matrix is:

$$\begin{bmatrix} P_{11} & 1 - P_{22} \\ 1 - P_{11} & P_{22} \end{bmatrix}$$

If $P_{11}=1$, the above Markov chain is an upper triangular. If the process enters regime 1, the process is not likely to be in regime 2. Regime 1 is an absorbing Markov chain, and Markov chain is reducible.

A Markov chain with *N* regimes is reducible if the transition matrix is $P = \begin{bmatrix} B & C \\ O & D \end{bmatrix}$, where B is a $K*K$ matrix where $N>k>1$. If P is an upper triangular, P_m for any V_m will also be upper triangular. Accordingly, if the process enters regime j, $k \geq j$ will not return to regimes $K+1$

and $K+2$ and …. If we have $P_{22}<1$ and $P_{11}<1$ for a chain with two regimes, a Markov chain is not reducible (Gebali & Gebali 2015, P.188).

3.1.4 Statistical Investigation of Homogeneous and Independently Distributed Compositions:

Imagine that random variable S_t presents a compound regime at time t, there are N regimes $(St=1,2,\dots,N)$. When in regime 1, the observed variable y_t follows distribution $N(\mu_1,\sigma_1)$. In regime 2, y_t is in distribution $N(\mu_2,\sigma_2)$, and so on. Conditional density of y_t if $S_t= i$ is (Hamilton, 1989):

$$\left\{ f\left(y_t \mid S_t = j,\theta\right) = \frac{1}{\sqrt{2\pi\sigma_j}} \exp\left(\frac{-\left(y_t - M_j\right)}{2\sigma_j}\right)^2 \right\} \qquad \text{Eq (10)}$$

where, θ is a vector of the following parameters:

$$\theta \to \left(\sigma_1,\sigma_2,\sigma_3,\dots,\sigma_N \quad \mu_1,\mu_2,\mu_3,\dots,\mu_N\right)$$

Unobserved regime S_t is generated from a probability distribution, where unconditional probability S_t takes value j from probability π_j:

$$P\{S_t = j,\theta\} = \pi_j \qquad j=1,2,\dots,N \qquad \text{Eq (11)}$$

The probabilities π_1 to π_N are included in θ .

the conditional probability of A in condition B for each occurrence A and B is:

$$P\left(A \mid B\right) = \frac{(A,B)}{P\left(B\right)} \qquad \text{Eq (12)}$$

In other words, the probability that A and B occur together is:

$$P\left(A,B\right) = P\left(B\right)P\left(A \mid B\right) \qquad \text{Eq (13)}$$

For example, if we want to calculate the joint probability $S_t=j$ where y_t is in the range $[c,d]$, we have:

$$P\left(y_t,S_t = j,\theta\right) = f\left(y_t \mid S_t = j,\theta\right).P\left(S_t = j,\theta\right) \qquad \text{Eq (14)}$$

This is joint density distribution function S_t and y_t. Accordingly (Hamilton, 1989):

$$P\left(y_t \mid S_t = j,\theta\right) = \frac{\pi_j}{\sigma\sqrt{2\pi}} \exp\left(\frac{-\left(y_t - \mu_j\right)^2}{2\sigma_j^2}\right) \qquad \text{Eq (15)}$$

Summing up all values of j in above statement presents unconditional density of y_t (Hamilton, 1989):

$$f\left(y_t,\theta\right)=\sum_{j=1}^{N}P\left(y_t,S_t=j,\theta\right) \qquad \text{Eq (16)}$$

As S_t is not observed, Expression 16 is the unconditional density of y_t explaining the actual observed y_t. If regime S_t is distributed homogeneously and independently over different times, maximum likelihood function for the observed data is computed as in expression Eq.17(Hamilton, 1989):

$$L\left(\theta\right)=\sum_{t=1}^{T}Logf\left(y_t,\theta\right) \qquad \text{Eq (17)}$$

Markov chain θ is estimated according to constraint $\pi\geq 0,\ \ \pi_1+\pi_2+,....,\pi_j=1.$

Functions $f\left(y_t,\theta\right)$ are used to show a wide range of densities. Figure 1 is an example of $N=2$. The joint distribution function $P\left(y_1,S_t=1,\theta\right)$ is π_1 multiplied by distribution $N\left(\mu=1,\sigma_1\right)$ and $P\left(y_t,S_t=2,\theta\right)$ is π_2 multiplied by distribution $N\left(\mu=3,\sigma_t\right)$. The unconditional density for observed variable $f\left(y_t,\theta\right)$ is the sum of these two.

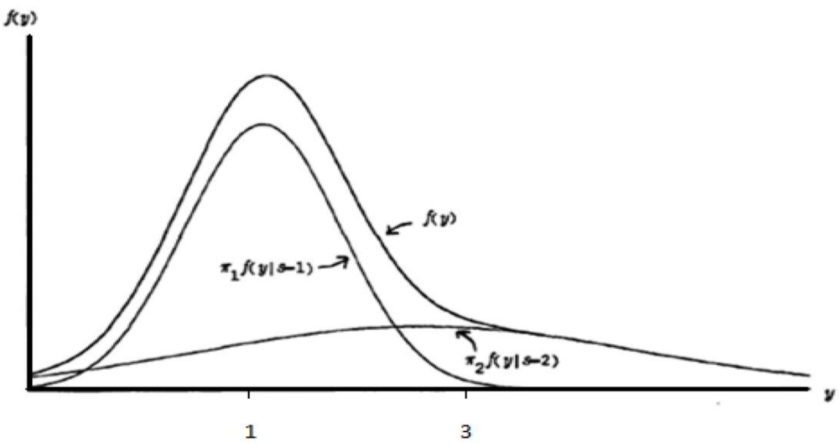

Figure 1:Joint distribution diagram of $P(y_1,S_t=1,\theta)$ **for two-regime case (Hamilton 1994, P.687)**

3.1.5 Inferences on unobserved regime

When evaluation θ is obtained, a conclusion can be obtained regarding which regime is more responsible to generate y_t in time t. According to the definition of conditional probability, we can have (Hamilton, 1989):

$$P\left(S_t = j \mid y_t,\theta\right) = \frac{P(y_t, S_t = j,\theta}{f\left(y_t,\theta\right)} = \frac{\pi_j . f\left(y_t \mid S_t = j,\theta\right)}{f\left(y_t,\theta\right)} \qquad \text{Eq (18)}$$

When parameters θ are available, expression $f\left(y_t \mid S_t = j,\theta\right)$ and $f\left(y_t,\theta\right)$ can be calculated for each y_t by using equation (18). The obtained value indicates the probability that the observed y_t at time *t* comes from regime *j*. If the observed y_t is one for Figure 1, it can be concluded that y_t is produced from $N\left(\mu = 1,\sigma_1\right)$, and $P\left(S_t = 1 \mid y_t,\theta\right)$ must be close to one. If the observed y_t is two, it is almost equally inclined towards both regimes. $P\left(S_t = 1 \mid y_t,\theta\right)$ is close to 0.5 for such data.

3.1.6 Maximum Likelihood Estimation

The maximum likelihood Estimate of $P\left(S_t = 1 \mid y_t,\theta\right)$ is as follows (Hamilton 1994, P 688):

$$\hat{\mu}_j = \frac{\sum_{t=1}^{T} y_t . P\left(S_t = j \mid y_t,\theta\right)}{\sum_{t=1}^{T} y_t . P\left(S_t = j \mid y_t,\theta\right)} \qquad \text{For j=1,2,...,N} \qquad \text{Eq (19)}$$

$$\hat{\sigma}_j = \frac{\sum_{t=1}^{T} \left(y_t - \hat{\mu}_j\right)^2 . P\left(S_t = j \mid y_t,\theta\right)}{\sum_{t=1}^{T} P\left(S_t = j \mid y_t,\theta\right)} \qquad \text{For j=1,2,...,N} \qquad \text{Eq (20)}$$

$$\hat{\pi}_j = 1/T \sum_{t=1}^{T} P\left(S_t = j \mid y_t,\theta\right) \qquad \text{For j=1,2,...,N} \qquad \text{Eq (21)}$$

If coming from the regime *j* was definite, $P\left(S_t = j \mid y_t,\theta\right)$ is one for the likelihood of coming from regime *j* and zero for others. The estimate of the mean (Eq. 19) for regime *j* is calculated in regime *j* observance average. In the general case where $P\left(S_t = j \mid y_t,\theta\right)$ is between zero and one, the estimate $\hat{\mu}_j$ is the weighted average of all observations in sample, where the weight of observation y_t is the probability that y_t was generated by regime *j*. The higher the probability for regime *j*, the greater the weight given to that observation in calculating $\hat{\mu}_j$. Then the estimate of the standard deviation for regime j in (Eq. 20) would simply be the average value of y, for those observations known to have come from regime j. Eq (21) represents the estimated probability of being in state j over the entire time period T.

3.1.7 Time Series Models with Regime-Switching

A model in which one variable can follow different time series in different samples is presented in this section (Hamilton 1994, P.690).

$$y_t = C(s_t) + \varphi(S_t) y_{t-1} + e_t \quad \text{Eq (22)}$$

where, $e_t \approx N(0, \sigma^2)$ and S_t result from a Markov chain with N regimes. S_t is independent of e_t for all t. Why the Markov chain is used in generating these models is a question. A regime change in Figure 2 is modeled with a chain with two regimes, where regime two is an absorbing regime indicating a permanent change. Using a Markov chain advantage for such a definitive specification is that it allows for meaningful predictions regarding the transition from regime 1 to regime 2 in the future.

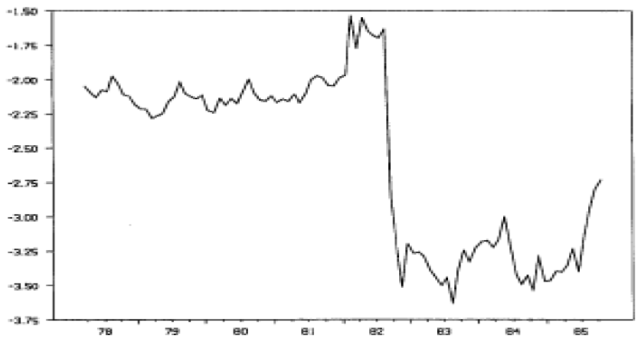

Figure 2: Regime switching model with an absorbing regime (Hamilton 1994, P.678)

If World War II is in a time series with regime change in calculation, regime 2 spans 5 periods with100 period of data in short section. Markov chain review suggests the reoccurrence of this event if we have data for the next 100 periods.

The general model in this section mentions that y_t is an $N*1$ vector of endogenous variables, and Y_t is a vector including all observations obtained in period t.

$$Yt = (y'_t, y'_{t-1}, y'_{t-2}, \ldots, y'_{t-m}) \quad \text{Eq (23)}$$

If the process in period t follows regime j, conditional density y_t is (Hamilton 1994, P.690):

$$f(y_t \mid S_t = j, Y_{t-1}; \alpha) \quad \text{Eq (24)}$$

where, a is parameter vector describing conditional density. If N different regimes have N different densities, these are presented $j=1:N$ in Equation (24). These densities are collected in vector $N*1$, called n_t. In the example Eq (22), y_t is an endogenous variable and exogenous variables include constant component, and parameter a equal $\left(C_1, C_2, ..., C_N, \varphi_1, \varphi_2, ..., \varphi_N, \sigma^2\right)$ (Hamilton 1994, P.691).

$$n_t = \begin{bmatrix} f\left(y_t \mid S_t = 1, Y_{t-1}; \alpha\right) \\ f\left(y_t \mid S_t = 2, Y_{t-1}; \alpha\right) \end{bmatrix} = \begin{bmatrix} \dfrac{1}{\sigma\sqrt{2\pi}} \exp\left(\dfrac{-\left(y_t - C_1 - \varphi_1 y_{t-1}\right)^2}{2\sigma^2} \right) \\ \dfrac{1}{\sigma\sqrt{2\pi}} \exp\left(\dfrac{-\left(y_t - C_1 - \varphi_1 y_{t-1}\right)^2}{2\sigma^2} \right) \end{bmatrix} \qquad \text{Eq (25)}$$

3.1.8 Optimal Inference and Likelihood Function Measurement

Parameters describing a time series generated by $f\left(y_t \mid S_t = j, Y_{t-1}; \alpha\right)$ and $P\left(S_t = j \mid S_{t-1} = i\right) = P_{ij}$ include *a* and transition probabilities P_{ij}, collected in a vector θ. The goal is to evaluate parameters inside θ based on the observations y_t. Even if the values θ are known for definite, it is unknown which regime governs the process. In contrast, a probabilistic inference can be proposed, which is an extension of $P\left(S_t = j \mid y_t; \theta\right)$. The obtained procedure is described in Equation (19). When distribution is homogeneous and independent, the researcher's inference regarding value depends on the value of y_t. Time series is dependent on all present observations in more general case.

Imagine that $P\left(S_t = j \mid y_t; \theta\right)$ is inferred to be present the value of S_t by the researcher. This inference is a conditional probability with period data t and θ parameters assumption known, which is the probability that the observation t is generated in the regime j. These conditional probabilities $P\left(S_t = j \mid y_t; \theta\right)$ for $j=1:N$ are collected in a vector $N*1$, called $\hat{E}_{t|t}$. The period data t can predict the process incline in period $t+1$. Vector $N*1$, called $\hat{E}_{t+1|t}$, collects prediction. The j^{th} element in this vector is $P\left(S_{t+1} = j \mid y_t; \theta\right) = P_{ij}$.

Optimal inference and prediction for period t in sample can be obtained by dual equations (Hamilton 1994, P.692):

$$\hat{E}_{t|t} = \frac{\left(\hat{E}_{t|t-1} \otimes \eta_t\right)}{1'\left(\hat{E}_{t|t-1} \otimes \eta_t\right)}$$ Eq (26)

$$\hat{E}_{t+1|t} = P.\hat{E}_{t|t}$$ Eq (27)

where, η_t presents vector $N*1$. The j^{th} element is represents the conditional density $f\left(y_t \mid S_t = j, Y_{t-1}; \alpha\right)$. *P* shows transition matrix $N*N$. $1'$ is the vector $N*1$ with elements equal to one, $\otimes$ presents elemental multiplication, $t=1:T$ in two above statements can repeat with initial value $\hat{E}_{1|0}$ and assumed vector parameters θ, for $t=1:T$ two above statements can be repeated then, $\hat{E}_{t|t}$ and $\hat{E}_{t+1|t}$ values are obtained in each period in the sample. A side conclusion calculates maximum likelihood $L(\theta)$ for observations y_t in value θ, used for iteration in this algorithm (Hamilton 1994, P.692).

$$L(\theta) = \sum_{t=1}^{T} \log f\left(y_t \mid Y_{t-1}; \theta\right)$$ Eq (28)

$$f\left(y_t \mid Y_{t-1}; \theta\right) = 1'\left(\hat{E}_{t|t-1} \otimes \eta_t\right)$$ Eq (29)

3.2 Recurrent neural network

What can be said about how people think is that it is not reset every second and the thinking process starts from the beginning. The moment you read this article; you will understand the meaning of each word according to the knowledge you have gained from reading the words in the past. In other words: When you read a text, you do not throw away the understanding you have gained from reading the previous words, but with each new word you gain a new understanding of the text you are reading and understand the meaning of the text.

Before RNNs existed, the usual neural networks used by machine learning experts could not act like humans, which was seen as a significant disadvantage for these networks. Therefore, RNNs were developed to solve this problem. RNN contains a recurrent loop, which means that the information from previous moments is not lost and is retained in the neural network. Figure 3 shows an RNN.

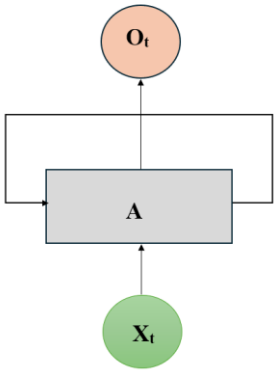

Figure 3: Recurrent Neural Networks (author's own observation)

In the figure above, section A receives the value of X_t as input and outputs the value of O_t. The loop causes information to be sent from one stage to the next. (Graves, et al., 2009)

3.2.1 Structure of a recurrent neural network

RNN is not different from ANN. RNN can be viewed as multiple identical copies of an ANN, each passing its information on to the next ANN. Figure 4 shows the state of the recurrent neural network in the form of an open scheme. In the following figure, the recursive cycle of section A, which was described in the previous section, is divided into several sections, and each section receives an input in the form of $\{X_1, X_2, X_3, \ldots, X_t\}$ and sends the value $\{O_1, O_2, O_3, \ldots, O_t\}$ at the output. The loop causes information to be sent from one stage to the next. Each part is considered a neuron. Thus, a RNN is a number of neurons whose output returns to itself t times.

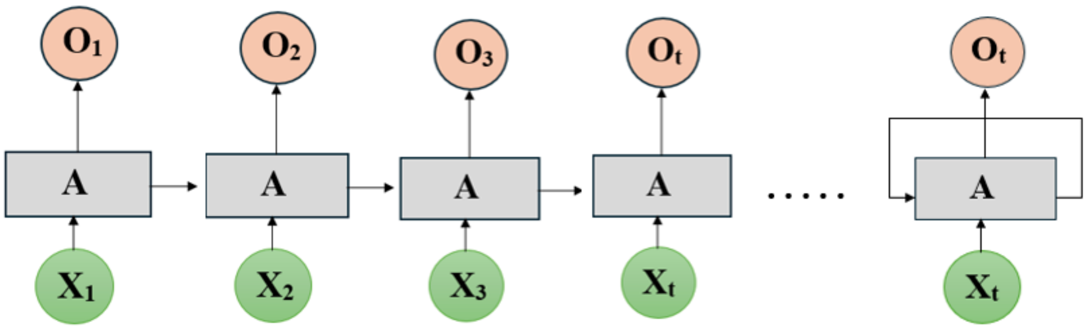

Figure 4: Recurrent Neural Network structure (author's own observation)

Due to the chain nature of RNN, these networks are strongly related to sequences and series. RNN is the first choice for working with such data. In recent years, these networks have been used repeatedly, leading to significant successes in various fields, such as speech recognition, language modeling, and time series prediction.

In theory, RNNs should be able to manage long-term dependencies, but in practice, unfortunately, RNNs cannot learn long-term dependencies.

3.2.2 LSTM neural network

The LSTM was first introduced in 1997 by Hochreiter and Schmidhuber. They were developing these networks aimed to solve the problem of long-term dependency. All RNNs have the form of recurrent chains of neural network units. In standard RNN, these recurrent modules have a simple structure, i.e. they contain only one hyperbolic tangent layer (Tanh). As can be seen in Figure 5, the structure of a recurrent network is described in terms of 3 blocks. The first block receives the input X_{t-1}, then the loop operates and sends not only the value .O_{t-1} to the output but also the same value O_{t-1} transmitted to block 2 and block 2 receives the input X_t in addition to the output of block 1 and performs tangential operations and generates the output O_t, which is transmitted as the output of block 2, and the input of block 3 and block 3 receives the value of X_t in addition to the input O_t also receives O_{t+1} as input and generates the output O_{t+1}.

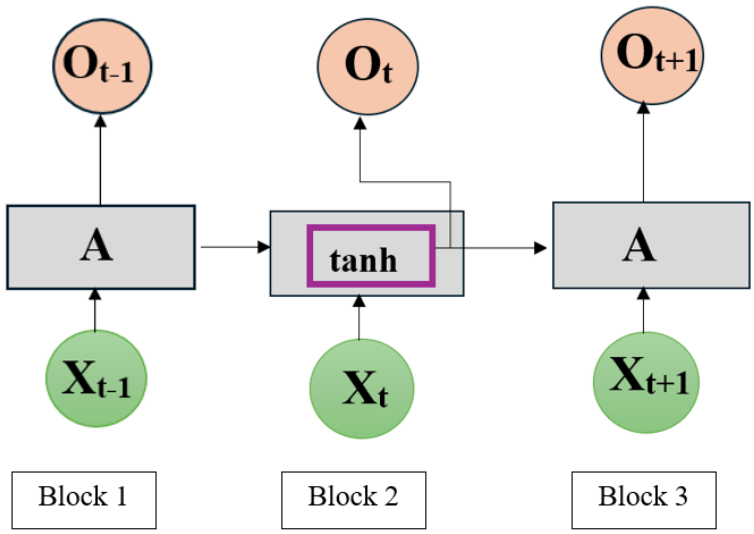

Figure 5: Recurrent modules in standard recurrent neural networks (author's own observation)

LSTM networks also have a chain-like structure, but their recurrent module has a different structure. Instead of just one neural network layer, they have 4 layers that interact and communicate with each other according to a particular structure.

3.2.3 LSTM - the structure of a neural network

The main element of LSTMs is the state cell, which is shown as a horizontal line in Figure 6. The state cell can be considered a conveyor belt that moves from the beginning to the end of the chain with partially linear interactions (its structure is straightforward, and few changes occur). Figure 6 shows the structure of a state cell that receives C_{t-1} as input and produces the output C_t after binary addition and multiplication.

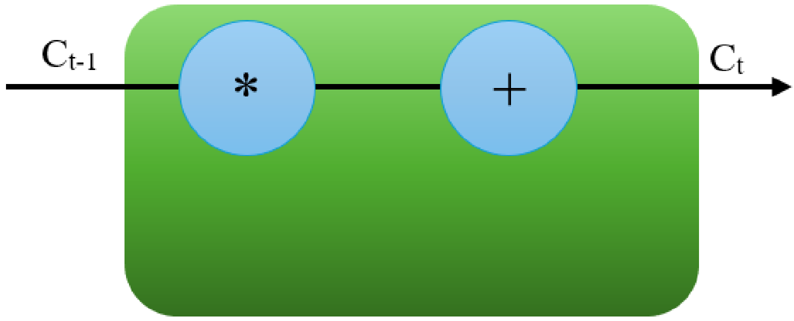

Figure 6: State cell (author's own observation)

The LSTM can add new information to the state cell or remove information from it. This is done through precise structures called gates. Gates are a way to input information optionally. They consist of a sigmoidal σ-layer of the neural network and a point-to-point multiplication operator (*), as shown in Figure 7.

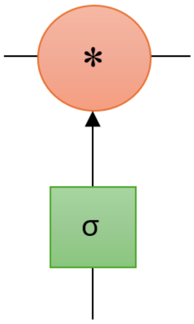

Figure 7: Gate (author's own observation)

The output of the sigmoid layer is a number between zero and one that indicates how much of the input should be sent to the output. A zero value means no information should be sent to the output, while a value of one means that all input should be sent to the output. The LSTM has 3 similar gates to control the value of the state cell.

3.2.4 Data input and output in the LSTM

The first step in LSTM is to decide what information we want to remove from the state cell. This decision is made by a sigmoid layer called the "forgetting gate". Depending on the values of O_{t-1} and X_t, this gate outputs the value zero or one for each number in the state cell C_{t-1}. A value of one means that the current value of the state cell (C_{t-1}) is completely transferred to C_t, and a value of zero means that the information of the current state cell (C_{t-1}) is completely erased, and none of it is transferred to C_t. As shown in Figure 8, the required values for deletion from the state cell result according to equation (30).

$$f_t = \sigma\left(w_f \cdot [O_{t-1}, x_t] + b_f\right) \qquad \text{Eq (30)}$$

x_t=Input vector

f_t= Output vector

σ= The original function of a sigmoid function

w_f and b_f = Matrixes and parameter vectors

O_{t-1}= Output vector of step $t-1$

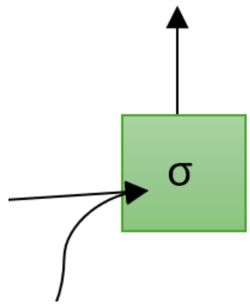

Figure 8: Forgetting gate (author's own observation)

The next step is to decide what new information we want to store in the state cell. This is a two-pronged decision. First, we have a sigmoid layer called input gate 1, which decides which values to update. The next step is a hyperbolic tangent layer that creates a vector of values labelled $\tilde{c}_t$ that can be added to the state cell. In the next step, these two steps are combined to update the value of the state cell. As can be seen in Figure 9, Equation (31) gives the updated values of the state cell and Equation (32) gives the values added to the state cell.

$$i_t = \sigma\left(w_i \cdot [O_{t-1}, x_t] + b_i\right) \qquad \text{Eq (31)}$$

i_t= Gate vector

x_t=Input vector

σ= The original function of a sigmoid function

w_f and b_f= Matrixes and parameter vectors

O_{t-1}= Output vector of step $t-1$

$$\tilde{C}_t = \tanh\left(w_c.[O_{t-1}, x_t] + b_c\right) \quad \text{Eq (32)}$$

$\tilde{c}_t$= State cell vector

tanh= The original function of a tangent function

w_f and b_f= Matrixes and parameter vectors

O_{t-1}= Output vector of step $t-1$

x_t=Input vector

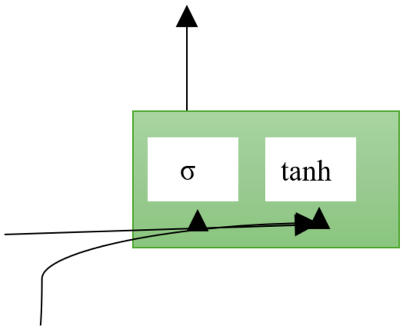

Figure 9: Input gate (author's own observation)

Now it is time for the old state cell, i.e. C_{t-1}, to be updated by the new state cell, i.e. C_t. According to equation (33), the previous value of the state cell is multiplied by f_t, which results in forgetting information that was previously decided to be overlooked. Then, the result of $i_t * \tilde{c}_t$ is added to it. Now, the new values of the state cell are determined according to the previously made decisions, as shown in Figure 10.

$$C_t = f_t * c_{t-1} + i_t * \tilde{C}_t \quad \text{Eq (33)}$$

$\tilde{c}_t$= State cell vector

i_t= Gate vector

C_t= The new value of the state cell

C_{t-1}= The old value of the state cell

f_t= Output vector

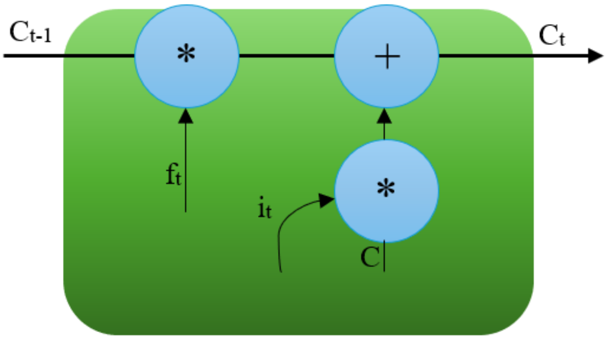

Figure 10: State cell new values block (author's own observation)

Finally, it must be decided which information is to be output. This output is the value of the state cell, which, however, passes through a particular filter. First, according to equation (34), a sigmoid layer decides which part of the state cell is to be output. Then, according to Equation (35), the value of the state cell (after being updated in the previous steps) is passed to a hyperbolic tangent layer (the values are between -1 and +1), and its value is multiplied by the output of the previous sigmoid layer until only superscripts (parts that are to go to the output, as shown in Figure 11 (Hochreiter and Schmidhuber 1997).

$$i_t = \sigma\left(w_0 . [O_{t-1}, x_t] + b_0\right) \quad \text{Eq (34)}$$

i_t= Gate vector

σ= The original function of a sigmoid function

w_f and b_f = Matrixes and parameter vectors

O_{t-1}= Output vector of step $t-1$

x_t=Input vector

$$O_t = i_t * \tanh(C_t) \quad \text{Eq (35)}$$

i_t= Gate vector

tanh= The original function of a tangent function

C_t= The new value of the state cell

O_t= Output vector of step t

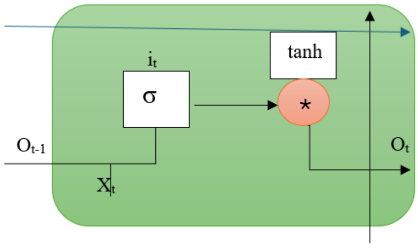

Figure 11: State cell output block (author's own observation)

3.3 Parameter setting

The parameter allocation of AI algorithms significantly affects the quality of the solution to the problem. Many researchers set the parameters based on the values determined in previous research. In this way, the quality of the solution to the problem cannot be guaranteed. Thus, it seems logical to set the parameters by adopting conventional mathematical and statistical approaches to the design of experiments (DOE). Experimenting all possible states is inefficient if there are many parameters.

3.3.1 Taguchi method

There are various statistical approaches to DOE to set algorithm parameters. Taguchi (1995) improved a family of fractional factorial experimental matrices. After many experiments, he managed to implement a DOE where the number of experiments for a given problem was reduced. The Taguchi method adopts orthogonal arrays to study many decision variables with a small number of experiments. Taguchi (1995) divides factors into two main classes: controllable factors and noise factors (directly uncontrollable). When it is impossible and infeasible to remove the noise factors, the Taguchi method minimizes the effect of noises and determines the optimal levels of controllable factors. Taguchi transforms repetition data into values that are a measure of variation in the results (S/N, where S indicates desirable values and N indicates undesirable values). Here, the aim is to maximize the S/N. In other words, Taguchi (1995) recommends analyzing changes by a properly selected S/N. His three standard SRs are:

Nominal-the-best: It is used to reduce the variation around a target value.

$$S/N_T = 10\log\left(\frac{\bar{y}^2}{S^2}\right) \quad \text{Eq (36)}$$

Larger-the-better: It is used to achieve the optimal system where the response is as large as possible.

$$S/N_L = -10\ \log\left(\frac{1}{n}\sum_{i=1}^{n}\frac{1}{y_i^2}\right) \quad \text{Eq (37)}$$

Smaller-the-better: It is used to obtain the optimal system where the response is as small as possible.

$$S/N_s = -10\ \log\left(\frac{1}{n}\sum_{i=1}^{n}y_i^2\right) \quad \text{Eq (38)}$$

where *n* is the number of executions of each experiment and process response.
Hence, according to objective functions, its corresponding rate is computed as:

$$S/N = 10\log\left(\frac{\overline{y}^2}{S^2}\right) \quad \text{Eq (39)}$$

3.4 Proposed model

This study proposes a new stock price prediction model that utilizes linear and non-linear time-series data components. Time-series data can be taken as components comprising linear and non-linear features (Eq 40).

$$\sum_{i=1}^{N}x_i(t) = \sum_{i=1}^{N}L_i(t) + \sum_{i=1}^{N}NL_i(t) \quad \text{Eq (40)}$$

Linear components indicate the general trend of data, while non-linear components indicate deviations or errors between observed values and predicted trends. This distinction is crucial as it allows deeper stock price perception and prediction, affected by both fixed market trends and sudden and unexpected changes.
The first and second components in equation (40) indicate linear and non-linear behaviors, respectively.
To this aim, the proposed model integrates machine learning techniques into time-series analysis by taking several steps designed to enhance prediction accuracy:

1) Data Division: Historical stock price data are divided into two subsets: training and testing. This division is essential for model training and evaluation.

2) MS Model Estimation: Training data are imported into the MS model as inputs to estimate stock prices. The MS model estimates the required parameters by adopting the maximum likelihood estimation method, thereby determining the trendline. The residual values are then computed as the difference between the actual stock prices and the values estimated by the MS model.

3) Long short-term memory (LSTM) training with error values: Residual values derived from MS model estimation are utilized as inputs to train the LSTM algorithm. At this stage, the LSTM algorithm is trained by its neural network layers to predict future error values based on historical error patterns.

4) MS Model Prediction: The stock price is predicted for the next day by the MS model based on the parameters derived from the second stage training data, reflecting the stock price trendline component.

5) LSTM Model Prediction: As in stage 4, the LSTM model predicts the residual values for the next day simultaneously through the knowledge acquired from the training data.

6) Combination of Predictions: The final stock price is predicted by combining the linear predictions of the MS model and the nonlinear predictions of the LSTM model. Stock prices predicted by the MS model and residual values predicted by the LSTM model are linearly added together according to Eq (40).

The following figure. 12 illustrates the general process of the proposed method herein.

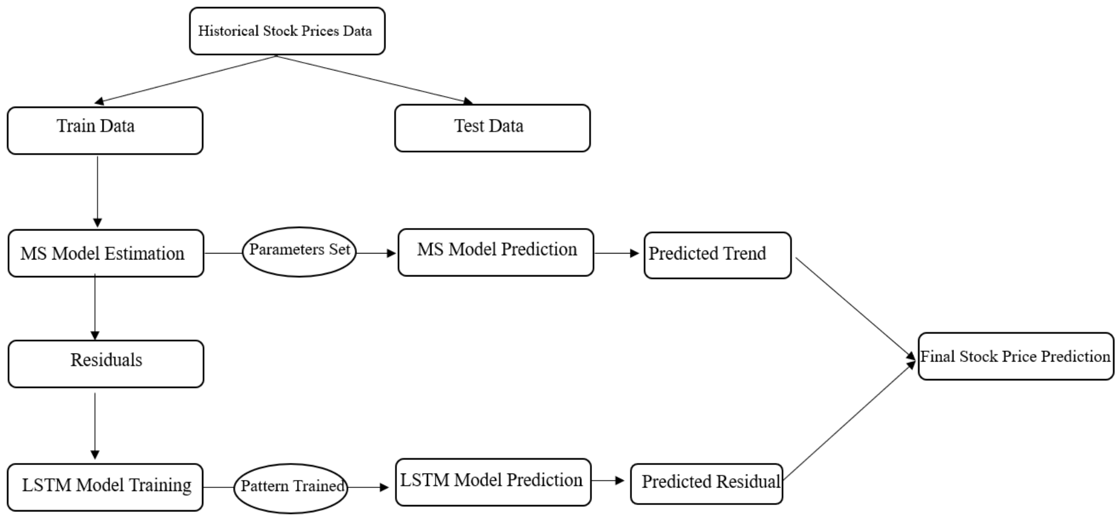

Figure 12: Steps of the proposed hybrid method (author's own observation)

4 Empirical results

4.1 Data Collection

According to the experts, the dataset used herein (Jan2019-Dec2023 period, including 5 years of stock price data) is employed to evaluate the performance of the proposed model, including historical stock prices obtained from 5 companies listed on the Helsinki Stock Exchange (Digia, Kone B, Nokia, UPM Kymmene and Wartsila). The companies were selected to demonstrate the Finnish market diversity and to ensure the evaluation of the model performance in various sectors and market conditions.

4.1.1 Descriptive Statistics

This research uses descriptive statistics to sum up the collected data and gain deeper insights into stock price behavior in the studied companies. The features and trends within the dataset can be better perceived by offering a comprehensive statistical overview. EViews 13 software is used to generate the following outputs.

Table 2: Descriptive statistics

	DIGIA	KONE_B	NOKIA	UPM_KYMMENE	WARTSILA
Mean	5.737477	54.39468	4.198585	29.73686	9.987796
Median	5.960000	51.77000	4.367000	30.51500	9.875000
Maximum	9.160000	75.86000	5.742000	36.95000	15.51500
Minimum	2.540000	38.17000	2.202500	20.98000	5.240000
Std. Dev.	1.546623	10.08558	0.740825	3.661382	2.262994
Skewness	-0.476402	0.353776	-0.112084	-0.394530	0.323779
Kurtosis	2.184784	1.904004	1.929437	2.077779	2.158229
Jarque-Bera	85.43433	92.46646	65.00209	80.03881	61.28307
Probability	0.000000	0.000000	0.000000	0.000000	0.000000
Sum	7481.670	70930.66	5474.955	38776.87	13024.09
Sum Sq. Dev.	3116.830	132539.7	715.1144	17467.65	6672.845
Observations	1304	1304	1304	1304	1304

4.2 Stock price prediction

This section implements the prediction process by going through the steps listed in chapter 3. Thus, the prediction steps by the proposed model are applied to the collected data.

Step 1: Data Division

The utilized dataset consists of daily stock prices for 1303 days from January 2019 to December 2023. The model is trained using the first 1279 data points, whereas the remaining 25 data points are reserved for testing the model performance.

Step 2: Markov model estimation

The following table 2 lists the outputs of the Markov-switching model estimation for Digia Co. The estimation has considered two different regimes for daily price time series. 13 software is used to generate the following outputs.

Table 3: Digia MS estimation output

Dependent Variable: DIGIA				
Method: Markov Switching Regression (BFGS / Marquardt steps)				
Date: 06/09/24 Time: 17:14				
Sample (adjusted): 2 1279				
Included observations: 1278 after adjustments				
Number of states: 2				
Initial probabilities obtained from ergodic solution				
Standard errors & covariance computed using observed Hessian				
Random search: 25 starting values with 10 iterations using 1 standard				
deviation (rng=kn, seed=1213001159)				
Failure to improve objective (non-zero gradients) after 11 iterations				
Variable	Coefficient	Std. Error	z-Statistic	Prob.
Regime 1				
C	5.558287	1.003928	5.536538	0.0000
Regime 2				
C	5.919431	1.004134	5.895061	0.0000
Common				
AR(1)	0.996622	0.002213	450.2828	0.0000
LOG(SIGMA)	-2.301345	0.023106	-99.60101	0.0000
Transition Matrix Parameters				
P11-C	4.004262	0.286051	13.99842	0.0000
P21-C	-2.426428	0.399678	-6.070965	0.0000
Mean dependent var	5.743318	S.D. dependent var		1.559931
S.E. of regression	0.117958	Sum squared resid		17.72667
Durbin-Watson stat	2.031674	Log likelihood		991.7020
Akaike info criterion	-1.542570	Schwarz criterion		-1.518377
Hannan-Quinn criter.	-1.533485			
Inverted AR Roots	1.00			

The following table 3 presents transition probabilities between different regimes. As can be seen, the regime transition probability is low, and a time series is most likely to stay in the same regime once it enters it. 13 software is used to generate the following outputs.

Table 4: Regimes transition probabilities

Equation: UNTITLED

Date: 06/09/24 Time: 17:15			
Transition summary: Constant Markov transition			
probabilities and expected durations			
Sample (adjusted): 2 1279			
Included observations: 1278 after adjustments			
Constant transition probabilities:			
P(i, k) = P(s(t) = k | s(t-1) = i)			
(row = i / column = k)			
		1	2
1		0.982089	0.017911
2		0.081179	0.918821
Constant expected durations:			
1		2	
55.83133		12.31838	

The following figure 13 depicts the filtered probabilities from the Markov model and the probability of certain time series in a specific regime at any time.

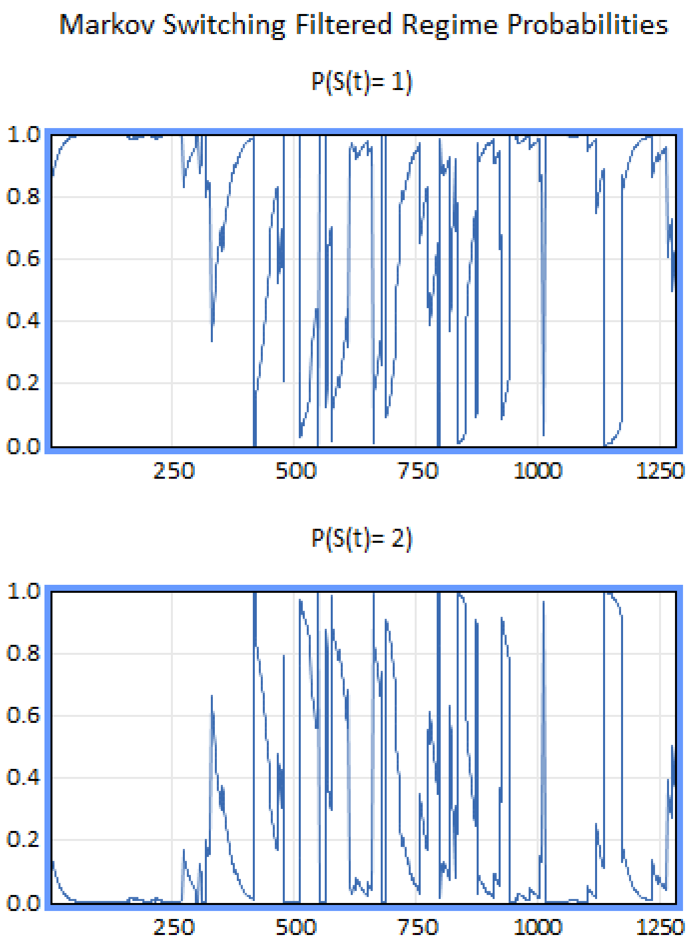

Figure 13:MS filtered regime probabilities

In regression analysis, an important assumption is that errors (i.e., residuals) follow the standard normal distribution (normal distribution with mean 0 and variance 1), which is essential to ensure the validity of regression-based prediction methods. Obviously, regression cannot be used if the above assumption does not hold true.

The following histogram (figure 14) illustrates the distribution of residuals (i.e., error values) from Markov model estimation. Since the errors almost follow the standard normal distribution according to the shape of the histogram, regression can be employed for prediction.

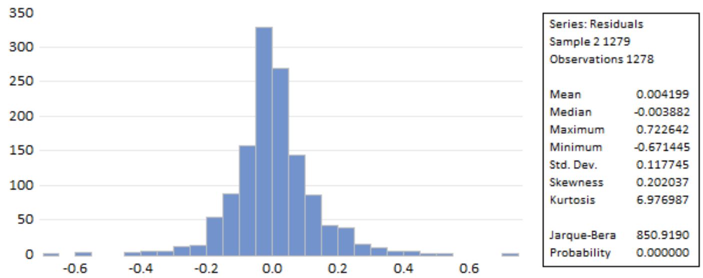

Figure 14: Residual distribution

The neural network parameters must be optimized before the second step on the prediction of the residual terms by the LSTM model. As some studies demonstrated earlier in Chapter 2, parameter setting helps improve the neural network performance.

For this purpose, we used the Taguchi (1995) method which is explained in section 3.3. Herein, three parameters were selected to be set, for each of which three levels were assumed. Therefore, 27 experiments must be conducted according to the orthogonal matrix.

Based on the computational results, the optimal parameters are those indicated with green highlight:

Table 5:LSTM optimized parameters

Initial Learn Rate	Max Epochs	Num Hidden Units
0.02	500	50
0.05	1000	100
0.08	1500	150

Step 3: LSTM training

In this step, Markov model estimation errors are taken as inputs for LSTM training. To ensure a robust training process, MS model training data are divided into two parts: training (n=1023; 80%) and testing (n=256; 20%). The LSTM algorithm exhibited a strong ability to track actual (real-world) data during the testing period and capture long-term dependencies in sequential data (figure 15). In the testing stage, the model performance demonstrates a close alignment with the real-world data, indicating that the LSTM algorithm has successfully learned the basic patterns in the error values.

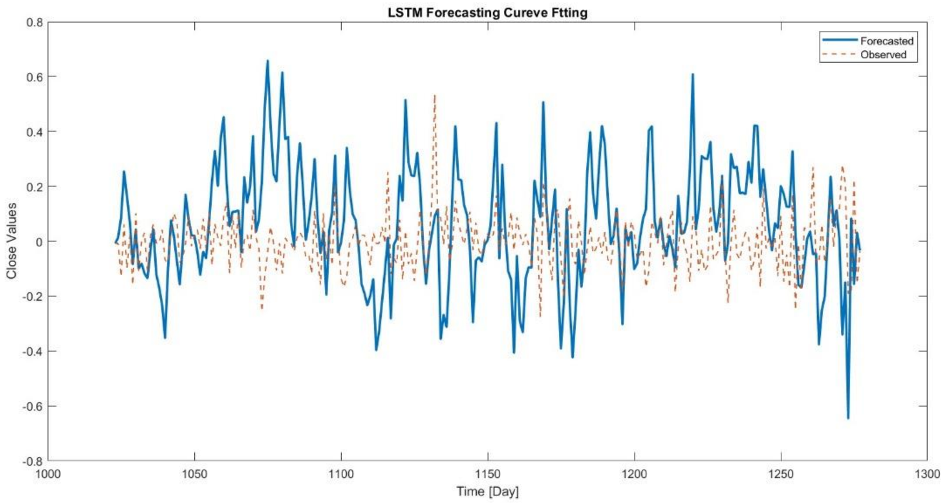

Figure 15:LSTM forecasting curve fitting

Step 4: Next-day price prediction by MS model

By taking advantage of Eviews prediction capabilities, the MS model is applied for next-day stock price prediction. Note that the model is constantly updated with new data to increase its accuracy. Particularly, following *t*-period price prediction, the actual *t*-day price value is included in the model for *t+1*-day price prediction. The iterative updating process ensures that the model is consistent with the latest market conditions.

The following figure 16 presents the predicted stock prices generated by the MS model, indicating its ability to predict future price volatility according to historical data and real-time updates.

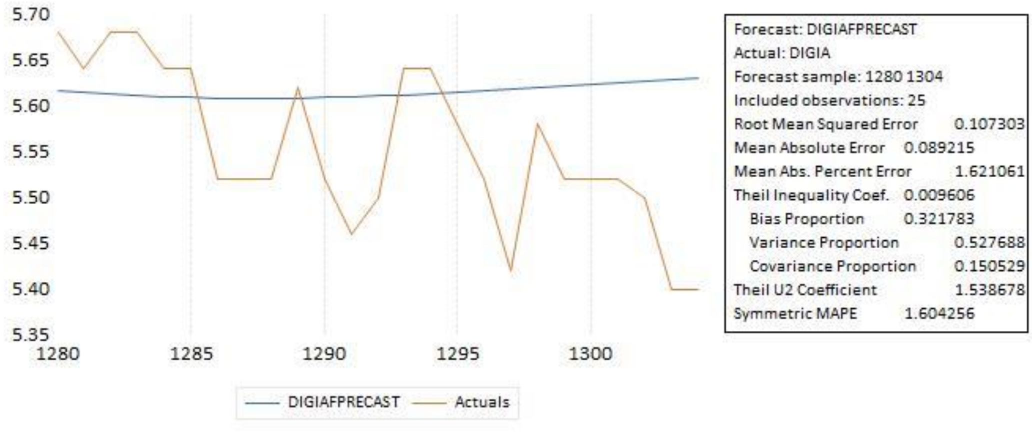

Figure 16:Digia trend prediction with MS

Step 5: Residual value prediction by the LSTM algorithm

Given Step-2 learning, residual values of future stock prices are predicted by the LSTM algorithm. Like Step 3, the model is constantly updated. Following *t*-period price prediction, the actual *t*-day price value is imported into the model to predict the *t+1*-day price. The following figure 17 illustrates the diagram of predicted stock prices generated by the LSTM algorithm, implying its effectiveness in future residual value prediction according to the iterative updating process.

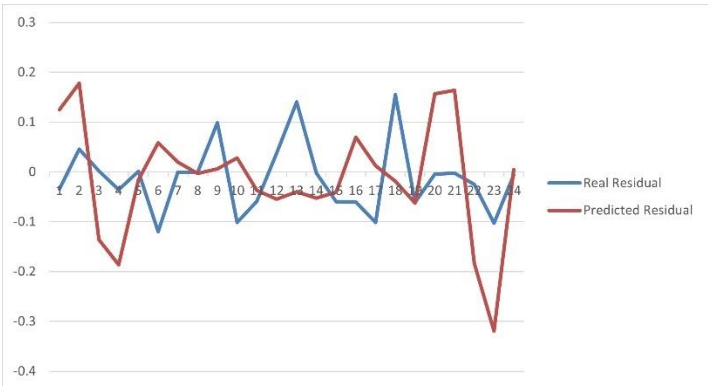

Figure 17:Digia predicted residual

Step 6: Calculating the final predicted price

As previously mentioned, the price time series can be expressed as Eq. (36). Thus, the values obtained in Steps 3 and 4 are added together to obtain the predicted price.

The following figure 18 displays the predicted price *vs* the actual value.

Every novel method must be tested, and its results need to be compared with existing methods to verify its superiority compared to previous research. To this end, herein, the proposed hybrid method is compared with MS and LSTM using three performance measures. To obtain the LSTM model, we input the historical stock prices instead of residual values. By following steps 3 and 5, we achieve the result shown as the blue line in Figure 18. For the MS model, we follow steps 2 and 4, resulting in the output depicted as the green line in Figure 18.

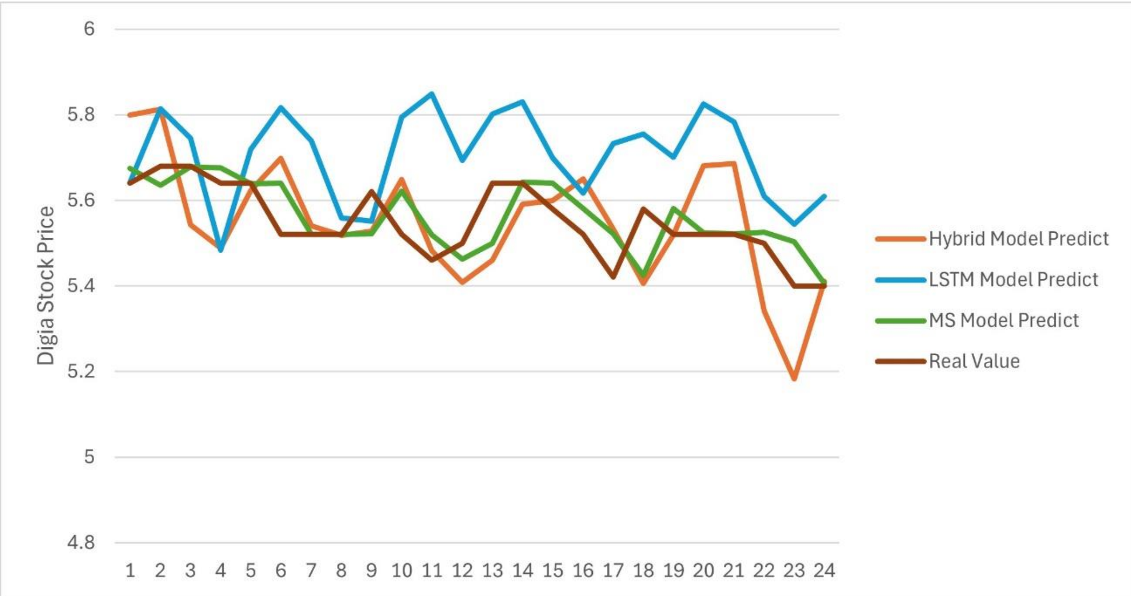

Figure 18:Models daily predict

4.4. Performance Measures

Prediction models are compared using the following measures, each of which will be briefly described later.

4.2.1 Root Mean Squared Error (RMSE)

If y_t is the value of a parameter, and $\hat{y}_t$ is the estimate of the said parameter, we will have:

$$\varepsilon = y_t - \hat{y}_t \quad \text{Eq (41)}$$

where ε is the *y* estimation error. The value of a proper estimate of y_t should be close to y_t, which can be evaluated by certain measures, including RMSE. RMSE measures the average magnitude of the error and calculates the difference between similar squared predicted and observed values . The number of samples is then averaged, and finally, the root mean is calculated. Since the errors are squared before they are averaged, the

RMSE gives a relatively high weight to large errors. This means the RMSE is most useful when large errors are particularly undesirable. RMSE is calculated as:

$$\mathrm{RMSE} = \sqrt{1/\mathrm{T}\sum_{\mathrm{t}=1}^{\mathrm{T}}(\mathrm{y_t} - \hat{y}_{\mathrm{t}})^2} \qquad \text{Eq (42)}$$

4.2.2 Mean Absolute Percentage Error (MAPE)

Another performance measure is MAPE, which is often utilized to measure the precision of a prediction model. It measures the average absolute percentage difference between actual and predicted values. MAPE is calculated as:

$$\mathrm{MAPE} = 1/\mathrm{T}\sum_{\mathrm{t}=1}^{\mathrm{T}} \frac{|y_{\mathrm{t}} - \hat{y}_{\mathrm{t}}|}{y'_{\mathrm{t}}} \qquad \text{Eq (43)}$$

MAPE output is a non-negative floating point. The best value is 0.0, indicating the best model prediction without any error.

4.2.3 R-Squared (R2)

The third performance measure is R^2, which determines the degree to which a regression model fits the data. It is also known as the coefficient of determination, indicating the variability in the target variable explained by the predictor. R^2 is calculated as:

$$R^2 = 1 - \frac{\sum_{t=1}^{T}(y_t - \hat{y}_t)}{\sum_{t=1}^{T}(y_t - \bar{y})} \qquad \text{Eq (44)}$$

where $\bar{y}$ is the sample mean value. This measure falls in the 0-1 range, where 0 means that the model does not explain any of the variability in the target (dependent) variable, and 1 means that the model explains all the variability. Hence, it can be concluded that the higher the R^2, the better the model fits the examined data.

The following tables 5-7 compare the proposed hybrid model with the MS & LSTM models.

Table 6:RMSE comparation of models (daily timeframe)

	Hybrid	LSTM	MS
digia	0.136894	0.098276	0.071239

kone_b	0.509117	0.716184	0.509714
nokia	0.128977	0.080398	0.075984
upm_kymmene	0.505974	0.378154	0.367094
wartsila	0.277577	0.245292	0.152377

Table 7: MAPE comparation of models (daily timeframe)

	Hybrid	LSTM	MS
digia	0.020243	0.014587	0.009473
kone_b	0.009114	0.01344	0.008912
nokia	0.032382	0.018752	0.018025
upm_kymmene	0.011954	0.008849	0.008464
wartsila	0.015665	0.015661	0.008942

Table 8: R^2 comparation of models (daily timeframe)

	Hybrid	LSTM	MS
digia	0.203238	0.320578	0.38723
kone_b	0.915403	0.885765	0.919725
nokia	0.484257	0.588557	0.632109
upm_kymmene	0.768273	0.880344	0.88271
wartsila	0.479568	0.615688	0.749973

According to the results, the MS model demonstrates superior stock price prediction performance across all measures (lowest RMSE and MAPE values, and the highest R^2 value) and stocks, while the proposed hybrid model shows the poorest performance in stock price prediction.

To increase the validity of the results of the proposed model, we also test it in a weekly time frame. For this purpose, historical price data from January 2000 to May 2024 was used. This period contains 1274 data sets showing the weekly price. The first 1267 data sets form the training data, and the last 7 data sets were used as valuation data. Descriptive statistics are presented like the daily time frame and then the results of the model performance test and its comparison with MS and LSTM models are presented.

To increase the robustness and validity of our results, we also evaluated the proposed model in a weekly time frame. For this purpose, historical price data from January 2000 to May 2024 was used, which includes a total of 1,274 data points representing weekly prices. Of these, the first 1267 data points were used to train the model, while the remaining 7 data points were used for testing. As with the analysis of the daily time frame, we first present descriptive statistics and then the performance results of the proposed model and the comparison with MS and LSTM models.

Figure 19 shows the price prediction graphs for each of the models.

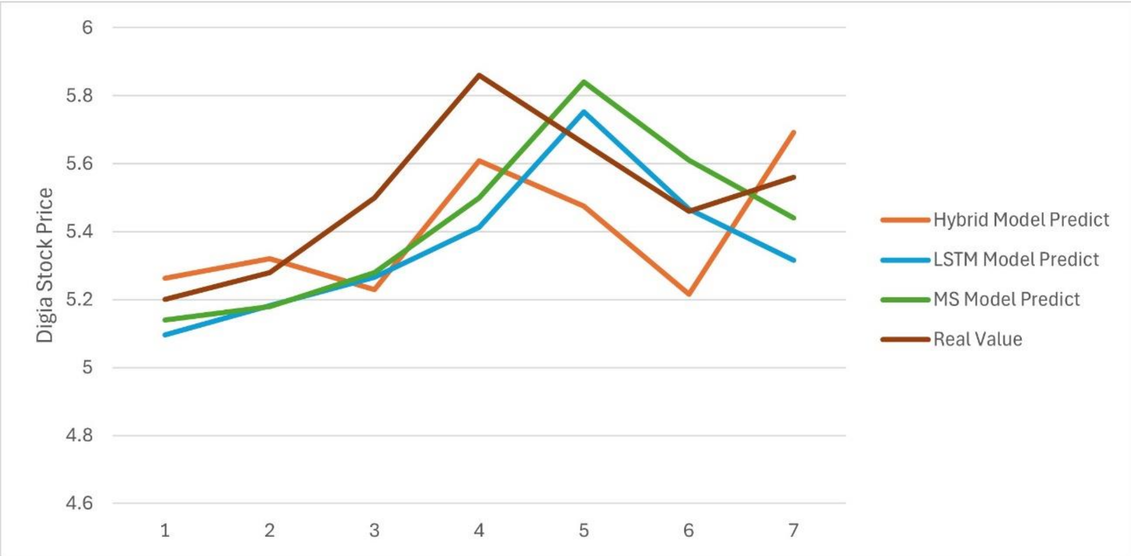

Figure 19: Models weekly predict

Table 9: RMSE comparation of models (weekly timeframe)

	Hybrid	LSTM	MS
digia	0.154576	0.206806	0.178543
kone_b	1.984365	1.988495	3.032185
nokia	0.054742	0.07589	0.055766
upm_kymmene	0.552314	0.82302	0.661891
wartsila	0.888655	1.04404	0.97258

Table 10: MAPE comparation of models (weekly timeframe)

	Hybrid	LSTM	MS
digia	0.022869	0.030136	0.028344
kone_b	0.034476	0.037834	0.048066
nokia	0.013664	0.01616	0.013432
upm_kymmene	0.015974	0.022563	0.016212
wartsila	0.023461	0.480425	0.038235

Table 11: R^2 comparation of models (weekly timeframe)

	Hybrid	LSTM	MS
digia	0.618505	0.426663	0.483581
kone_b	0.395068	0.340172	0.265193

nokia	0.994957	0.757119	0.88208
upm_kymmene	0.720162	0.67848	0.783018
wartsila	0.872096	0.671247	0.813946

According to the tables 8-10, the results show that the proposed model outperforms the others in all key metrics RMSE, MAPE and R^2. A possible explanation for this better performance could be the higher volatility observed in the weekly time frame. Due to the proximity of the daily prices to each other and the low volatility, the real prices are closer to their trend line, which leads to a better performance of the MS model. In contrast, in the weekly time frame, the variance between the real prices and the trend line is larger. As a result, a significant part of the difference between the predicted and actual prices is predicted more accurately by the LSTM model, which helps to improve the performance of the proposed model in predicting the weekly prices.

These results show that the proposed model is particularly effective in high volatility environments. In markets with high volatility, such as forecasts in higher time frames (monthly, annually) or markets such as cryptocurrencies, which generally have higher fluctuations than the stock market, the proposed model is a suitable choice for these applications.

5 Conclusions

In the modern world, the capital market is one of the most important markets for investments. Although investing in the capital market can be profitable for many investors, it has caused significant losses for many stockholders due to sharp price fluctuations. Therefore, predicting the performance of stock prices is always one of the main concerns of stockholders and capital market participants. Participants and stockholders always try to find ways to increase the profit of their capital as much as possible by increasing the accuracy of forecasts. To achieve this, it seems necessary to consider appropriate, correct and principle-based methods for determining the future stock price. However, predicting stock price performance in the stock market is very complicated and difficult due to its complexity and dynamics as well as the existence of many factors and variables that affect the stock price.

The efficient market hypothesis states that all available information is fully and directly reflected in the price of an asset, so that it is not possible to make systematic profits by predicting prices. The efficiency of capital markets can be examined at various levels.

The theory of weak market efficiency states that all information from the past, such as previous stock prices, is reflected in the current stock price and therefore does not contribute to higher returns on stock investments.

According to the semi-strong market efficiency, all public information available to investors is reflected in the current stock price, so that technical and fundamental market analysis is ineffective for higher stock returns. According to the strong market efficiency, all information, whether public or private, is reflected in the current stock price, and in this case even private information within companies cannot lead to higher stock returns.

Shaker (2013) examined market efficiency at a weak level in the Finnish and Swedish stock markets. The results show that the weak form of market does not exist in the stock markets of these countries. Therefore, it is possible to make predictions about future stock prices.

Statistical methods based on time series such as ARIMA, regression or a combination of these were popular approaches for predicting stock prices. By their very nature, these methods can provide a good approximation in environments with small changes, but in cases such as stock market prediction, where environmental conditions are constantly changing, they cannot estimate a good approximation of the environmental changes. To predict these periods, new tools and models need to be used. Nowadays, intelligent systems and methods such as neural networks, genetic algorithms, fuzzy logic, etc. are used. They are used in many areas of science. Recently, LSTM have become very attractive for the stock market due to their nonlinear nature in predicting stock prices. Therefore, a hybrid model of MS and LSTM model was presented in this study to capture both the linear and nonlinear properties of stock price time series.

As mentioned in the literature review section, many studies have shown that hybrid models outperform individual models in predicting stock prices (Yu and Yan, 2020, Ji et al. 2021). According to the studies, the LSTM model has shown better performance in stock prediction compared to ANN (Ma,2020, Ho et al. 2021). Moreover, by exploiting the ability to change the regime, the MS model allows us to recognize the presence of different regimes in time series and based on that, predict the stock price based on different regimes.

In the model proposed in this study, the linear part of the time series (trend) was predicted by the MS model and the non-linear part (volatility of difference between the actual value and the trend line) was predicted by the LSTM model. To evaluate the performance of the model, five companies (Digia, Kone B, Nokia, UPM Kymmene and Wartsila) were selected from the Finnish Stock Exchange. The data used for the model included historical prices from January 2019 to December 2023.

The prediction steps were performed in such a way that first the MS model estimated the parameters of the model using the training data and then the individual values (difference between the actual and estimated values) were utilized as inputs to the LSTM model to perform its learning phase. Before starting the training, the optimal values of the parameters of the LSTM model were determined using the Taguchi method. In the next step, the future price was predicted by the MS model and the error value was also predicted by the LSTM model. Finally, the resulting values were combined linearly to obtain the final price predicted by the proposed model.

The results of the proposed model were then compared with the MS and LSTM models separately using the RMSE, MAPE and R^2 criteria. The results showed that the MS model performed the best and the proposed model performed the worst. To increase the robustness and reliability of the results obtained, the model was also tested with weekly time frame data. The results showed that the use of weekly data leads to an improvement in the performance of the proposed model, so that the proposed model has the best performance. The reason for this could be due to the higher volatility of weekly price data. Therefore, the proposed model is recommended for price prediction in markets and stocks with high volatility. In response to the study question "How does the integration of MS and LSTM models improve the accuracy of stock price prediction compared to single models?" It should be noted that the proposed hybrid model works well under conditions of extreme fluctuations compare to single MS and LSTM models. By predicting trends and residual values more accurately, this model can lead to more accurate forecasts and help investors make better decisions.

The findings of this study and the proposed model will help other researchers to develop new models for stock price prediction using the new approach introduced in this research. Using this innovative method, researchers can increase the accuracy of their forecasts and contribute to the development of financial forecasting techniques.

Additionally, the model's predictive capabilities can be useful as a risk management tool for market participants who prioritize capital preservation and aim to maximize profits during periods of high market volatility. By integrating this model into investment strategies, market participants can better anticipate market fluctuations, reduce potential losses, and improve overall portfolio performance. This dual utility—supporting further research and helping practical risk management—underscores the importance and versatility of the model in the financial domain.

Future research can be done by considering the wide range of different models for predicting the future price, comparing these models with each other can provide valuable insights. For example, different neural networks such as MLP and SVM can be used in combination with time series trend prediction models. In addition, examining different markets such as energy, forex, cryptocurrencies, etc. in different time frames can help identify the conditions under which the model performs better.

These studies can help to identify the strengths and weaknesses of hybrid models and show how these models can work more effectively under different market conditions and with different data. In this way, it is possible to improve and optimize further models and find wider application in predicting prices in financial markets and other areas.

Finally, developing hybrid models that benefit from multiple techniques and methods simultaneously can be an effective solution to address the complexity and fluctuations of financial markets. This versatile approach can increase the accuracy of forecasts and help investors and analysts make more informed decisions.

6 References

Afeef, M., Ihsan, A., & Zada, H. (2018). Forecasting stock prices through univariate ARIMA modeling. NUML International Journal of Business & Management, 13(2), 130-143.

Allen, F., Brealey, R., Myers, S. (2011). Principles of Corporate Finance. New York: McGraw-Hill/Irwin.

Ariyo, A. A., Adewumi, A. O., & Ayo, C. K. (2014, March). Stock price prediction using the ARIMA model. In *2014 UKSim-AMSS 16th international conference on computer modelling and simulation* (pp. 106-112). IEEE.

Armstrong, J. S. (Ed.). (2001). *Principles of forecasting: a handbook for researchers and practitioners* (Vol. 30, pp. 1-2). Boston, MA: Kluwer Academic.

Banerjee, S., & Mukherjee, D. (2022). Short term stock price prediction in indian market: A neural network perspective. *Studies in Microeconomics*, *10*(1), 23-49.

Berglund, T., Wahlroos, B., & Örnmark, A. (1983). The weak-form efficiency of the finnish and scandinavian stock exchanges: A comparative note on thin trading. *The Scandinavian Journal of Economics*, *85*(4), 521-530.

Bildirici, M., & Ersin, Ö. (2014). Modeling Markov switching ARMA-GARCH neural networks models and an application to forecasting stock returns. *The Scientific World Journal*, *2014*(1), 497941.

Billah, M., Waheed, S., & Hanifa, A. (2015). Predicting closing stock price using artificial neural network and adaptive neuro fuzzy inference system (anfis): the case of the dhaka stock exchange. *International Journal of Computer Applications*, *129*(11), 1-5.

Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.

Chung, H., & Shin, K. S. (2018). Genetic algorithm-optimized long short-term memory network for stock market prediction. *Sustainability*, *10*(10), 3765.

Couillard, M., & Davison, M. (2005). A comment on measuring the Hurst exponent of financial time series. *Physica A: Statistical Mechanics and its Applications*, *348*, 404-418.

Chen, S., & Zhou, C. (2020). Stock prediction based on genetic algorithm feature selection and long short-term memory neural network. *IEEE Access*, *9*, 9066-9072.

De Oliveira, F. A., Nobre, C. N., & Zárate, L. E. (2013). Applying Artificial Neural Networks to prediction of stock price and improvement of the directional prediction index–Case study of PETR4, Petrobras, Brazil. Expert systems with applications, 40(18), 7596-7606.

Degutis, A., & Novickytė, L. (2014). The efficient market hypothesis: A critical review of literature and methodology. Ekonomika, 93(2), 7-23.

Devadoss, A. V., & Ligori, T. A. A. (2013). Stock prediction using artificial neural networks. *International Journal of Data Mining Techniques and Applications*, *2*(1), 283-291.

Dimson, E., & Mussavian, M. (1998). A brief history of market efficiency. *European financial management*, *4*(1), 91-103.

Dutta, G., Jha, P., Laha, A. K., & Mohan, N. (2006). Artificial neural network models for forecasting stock price index in the Bombay stock exchange. *Journal of Emerging Market Finance*, *5*(3), 283-295.

Esfahanipour, A., & Mardani, P. (2011, June). An ANFIS model for stock price prediction: The case of Tehran stock exchange. In *2011 International Symposium on Innovations in Intelligent Systems and* Applications (pp. 44-49). IEEE.

Fama, E. F. (1970). Efficient capital markets. *Journal of finance*, *25*(2), 383-417.

Fama, E. F. (1976). Efficient capital markets: reply. *The Journal of Finance*, *31*(1), 143-145.

Fama, E. F. (1991). Efficient capital markets: II. *The journal of finance*, *46*(5), 1575-1617.

Frennberg P., and Hansson B. 1993. Testing the random walk hypothesis on Swedish stock prices: 1919- 1990. *Journal of Banking and Finance, 17*(1), 175-191.

Ganesan, A., & Kannan, A. (2021). Stock price prediction using ARIMA model. *International* Research Journal of Engineering and Technology (IRJET), 8(08), 226-234.

Gebali, F., & Gebali, F. (2015). Reducible Markov Chains. Analysis of Computer Networks, 157-189.

Gopinathan, K. N., Murugesan, P., & Jeyaraj, J. J. (2024). Stock price prediction using a novel approach in Gaussian mixture model-hidden Markov model. *International Journal of Intelligent Computing and Cybernetics*, *17*(1), 61-100.

Göçken, M., Özçalıcı, M., Boru, A., & Dosdoğru, A. T. (2016). Integrating metaheuristics and artificial neural networks for improved stock price prediction. *Expert Systems with Applications*, *44*, 320-331.

Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., & Schmidhuber, J. (2008). A novel connectionist system for unconstrained handwriting recognition. *IEEE transactions on pattern analysis and machine intelligence*, *31*(5), 855-868.

Intelligence Magazine, Vol. 31, No. 5, pp. 24-38, May 2009. Grossman, S. J., & Stiglitz, J. E. (1980). On the impossibility of informationally efficient markets. *The American economic review*, *70*(3), 393-408.

Gupta, B. K., Mallick, M. K., & Hota, S. (2021). Survey on stock price forecasting using regression analysis. In *Intelligent and Cloud Computing: Proceedings of ICICC 2019, Volume 2* (pp. 147-156). Springer Singapore.

Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. Econometrica: Journal of the econometric society, 357-384.

Hamilton, J. D. (1994). Time Series Analyses. Princeton, New Jersey: Princeton University Press.

Ho, M. K., Darman, H., & Musa, S. (2021, July). Stock price prediction using ARIMA, neural network and LSTM models. In *Journal of Physics: Conference Series* (Vol. 1988, No. 1, p. 012041). IOP Publishing.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

Hossain, M. A., Karim, R., Thulasiram, R., Bruce, N. D., & Wang, Y. (2018, November). Hybrid deep learning model for stock price prediction. In *2018 ieee symposium series on computational intelligence (ssci)* (pp. 1837-1844). IEEE.

Islam, M. R., & Nguyen, N. (2020). Comparison of financial models for stock price prediction. *Journal of Risk and Financial Management*, *13*(8), 181.

Jennergren L., and Korsvold P. 1974. Price Formation in the Norwegian and Swedish Stock Markets: Some Random Walk Tests. *Swedish Journal of Economics, 76*(2), 171.

Ji, Y., Liew, A. W. C., & Yang, L. (2021). A novel improved particle swarm optimization with long-short term memory hybrid model for stock indices forecast. *Ieee Access*, *9*, 23660-23671.

Jones, S. L., & Netter, J. M. (2008). Efficient capital markets. *Concise Encyclopedia of Economics*, *138*, 139-141.

Khan, Z. H., Alin, T. S., & Hussain, M. A. (2011). Price prediction of stock market using artificial neural network (ANN). *International Journal of Computer Applications*, *22*(2), 42-47.

Kabir, M. H., Sobur, A., & Amin, M. R. (2023). Stock Price Prediction Using The Machine Learning. International Journal of Computer Research and Technology (IJCRT), 11(7).

Khanderwal, S., & Mohanty, D. (2021). Stock price prediction using ARIMA model. International Journal of Marketing & Human Resource Research, 2(2), 98-107.

Khare, K., Darekar, O., Gupta, P., & Attar, V. Z. (2017, May). Short term stock price prediction using deep learning. In *2017 2nd IEEE international conference on recent trends in electronics, information & communication technology (RTEICT)* (pp. 482-486). IEEE.

Khashei, M., & Hajirahimi, Z. (2020). A comparative study of series arima/mlp hybrid models for stock price forecasting. *Communications in Statistics-Simulation and Computation*, *48*(9), 2625-2640.

Khuat, T. T., & Le, M. H. (2017). An application of artificial neural networks and fuzzy logic on the stock price prediction problem. *JOIV: International Journal on Informatics Visualization*, *1*(2), 40-49.

Kobiela, D., Krefta, D., Król, W., & Weichbroth, P. (2022). ARIMA vs LSTM on NASDAQ stock exchange data. *Procedia Computer Science*, *207*, 3836-3845.

Kumar Chandar, S. (2021). Grey Wolf optimization-Elman neural network model for stock price prediction. Soft Computing, 25, 649-658.

Kuwornu, J. K., & Owusu-Nantwi, V. (2011). Macroeconomic variables and stock market returns: Full information maximum likelihood estimation. Research journal of finance and accounting, 2(4), 49-63.

Ma, Q. (2020). Comparison of ARIMA, ANN and LSTM for stock price prediction. In *E3S Web of Conferences* (Vol. 218, p. 01026). EDP Sciences.

Machová, V., & Vochozka, M. (2019). Using Artificial Intelligence in Analyzing and Predicting the Development of Stock Prices of a Subject Company. *Sustainable Growth and Development of Economic Systems: Contradictions in the Era of Digitalization and Globalization*, 235-245.

Mahadik, A., Vaghela, D., & Mhaisgawali, A. (2021, August). Stock price prediction using lstm and arima. In 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 1594-1601). IEEE.

Malkiel, B. G. (2011, October). The efficient-market hypothesis and the financial crisis. In Rethinking finance: perspectives on the crisis (Proceedings of a conference). Russel Sage Foundation.

Merh, N., Saxena, V. P., & Pardasani, K. R. (2010). A comparison between hybrid approaches of ANN and ARIMA for Indian stock trend forecasting. *Business Intelligence Journal*, *3*(2), 23-43.

Metghalchi M., Chang Y., and Marcucci J. 2008. Is the Swedish stock market efficient? Evidence from some simple trading rules. *International Review of Financial Analysis, 17*(3), 475-490.

Mishkin, F. S., & Eakins, S. G. (2006). Financial markets and institutions. Pearson Education India.

Mondal, P., Shit, L., & Goswami, S. (2014). Study of effectiveness of time series modeling (ARIMA) in forecasting stock prices. *International Journal of Computer Science, Engineering and* Applications, 4(2), 13.

Naseer, M., & Bin Tariq, D. Y. (2015). The efficient market hypothesis: A critical review of the literature. The IUP journal of financial risk management, 12(4), 48-63.

Nguyen, N. (2017). An analysis and implementation of the hidden Markov model to technology stock prediction. Risks, 5(4), 62.

Pai, P. F., & Lin, C. S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *Omega*, *33*(6), 497-505.

Pan, W. T. (2010). Performing stock price prediction use of hybrid model. Chinese Management Studies, 4(1), 77-86.

Rapach, D. E., Strauss, J. K., & Zhou, G. (2010). Out-of-sample equity premium prediction: Combination forecasts and links to the real economy. *The Review of Financial Studies*, *23*(2), 821-862.

Santoso, E. B., & Ikhsan, M. (2020, March). Efficient market hypothesis in Indonesia stock exchange 2019. In Annual International Conference on Accounting Research (AICAR 2019) (pp. 51-53). Atlantis Press.

Sattayatham, P., Sopipan, N., & Premanode, B. (2012). Forecasting volatility and price of the SET50 index using the Markov regime switching. *Procedia Economics and Finance*, *2*, 265-274.

Sebastian, St. B. J. (1989). The Role of the Securities Market in Mobilizing Resources for the Region. Social and Economic Studies, 38(4), 115–132. http://www.jstor.org/stable/27864906

Senapati, M. R., Das, S., & Mishra, S. (2018). A novel model for stock price prediction using hybrid neural network. *Journal of the Institution of Engineers (india): Series B*, *99*(6), 555-563.

Shaker, A. T. M. (2013). Testing the weak-form efficiency of the Finnish and Swedish stock markets. European Journal of Business and Social Sciences, 2(9), 176-185.

Shleifer, A. (2000). Inefficient markets: An introduction to behavioural finance. Oup Oxford.

Taguchi, G. (1995). Quality engineering (Taguchi methods) for the development of electronic circuit technology. IEEE transactions on reliability, 44(2), 225-229.

Van Eyden, R. J. (1996). The Application of Neural Networks in the Forecasting of Share Prices (Finance and Technology Publishing, Haymarket, VA).

Vijh, M., Chandola, D., Tikkiwal, V. A., & Kumar, A. (2020). Stock closing price prediction using machine learning techniques. *Procedia computer science*, *167*, 599-606.

Vui, C. S., Soon, G. K., On, C. K., Alfred, R., & Anthony, P. (2013, November). A review of stock market prediction with Artificial neural network (ANN). In *2013 IEEE international conference on control system, computing and engineering* (pp. 477-482). IEEE

Wang, J. Z., Wang, J. J., Zhang, Z. G., & Guo, S. P. (2011). Forecasting stock indices with back propagation neural network. *Expert Systems with Applications*, *38*(11), 14346-14355..

Wang, L., Wu, J., Cao, Y., & Hong, Y. (2022). Forecasting renewable energy stock volatility using short and long-term Markov switching GARCH-MIDAS models: Either, neither or both?. *Energy Economics*, *111*, 106056.

Wijaya, Y. B., Kom, S., & Napitupulu, T. A. (2010, December). Stock price prediction: comparison of Arima and artificial neural network methods-An Indonesia Stock's Case. In *2010 Second International Conference on Advances in Computing, Control, and* Telecommunication Technologies (pp. 176-179). IEEE.

Wijesinghe, G. W. R. I., & Rathnayaka, R. M. K. T. (2020, December). Stock Market Price Forecasting using ARIMA vs ANN; A Case study from CSE. In 2020 2nd International Conference on Advancements in Computing (ICAC) (Vol. 1, pp. 269-274). IEEE.

Yu, P., & Yan, X. (2020). Stock price prediction based on deep neural networks. *Neural Computing and Applications*, *32*(6), 1609-1628.

Zhu, X., & Zhu, J. (2013). Predicting stock returns: A regime-switching combination approach and economic links. *Journal of Banking & Finance*, *37*(11), 4120-4133.