International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 12 Issue: 04 | Apr 2025 www.irjet.net p-ISSN: 2395-0072

A Review of Stock Price Prediction Using Machine Learning Techniques

Tan Chun Fui[1], Tan Lay Hong[2] , Ajay Kumar Singh[3]

[1] *Senior Lecturer, Faculty of Information Science Technology, Multimedia University-Jalan Ayer Keroh Lama, Melaka, Malaysia.*
[2] *Senior Lecturer, Universiti Teknikal Malaysia Melaka (UTeM), Fakulti Pengurusan Teknologi Dan Teknousahawanan (FPTT), Centre of Technopreneurship Development (CTeD), 75450 Ayer Keroh, Melaka, Malaysia.*
[3] *Professor, Electronics and Communication Engineering NIIT University, Alwar, Rajasthan India.*

Abstract - *This paper reviews existing literature on predicting stock prices using machine learning techniques, emphasizing the growing importance of accurate stock price predictions for making informed financial decisions. The primary goal of this study is to provide practical guidelines for beginners entering the field of machine learning for stock price prediction instead of just providing the knowledge of machine learning or comparing the advantages and disadvantage of the algorithm. The review encompasses various academic journals. For the selection of research papers, the study focuses on publications from 2010 to 2023, a period marked by significant advancements in machine learning. The criteria for choosing these 24 research papers are based on their implementation of different machine learning methods for stock price prediction, along with the presence of results, data processing processes, or algorithms. By examining various machine learning methods employed in stock price prediction and their implementation details, this review aims to distill actionable insights for newcomers. It summarizes key findings and extracts practical guidance, providing novice practitioners with a structured entry point into the world of machine learning for stock price prediction system. Additionally, the paper acknowledges the limitations of current research and suggests potential areas for future exploration, ensuring a comprehensive and informative resource for those venturing into stock price prediction using machine learning.*

Key Words: Stock price predictions ·Machine learning techniques · Real-time market data · Financial indicators · Predictive models.

1.INTRODUCTION

The stock market is vital for our economy, influencing how businesses are run and helping people manage their money. It can encourage companies to make better long-term decisions, but also carries significant risky [1]. Market volatility poses challenges for investors and companies, prompting researchers to develop predictive methods to aid wise investment decisions and minimize losses. This is an important and active area of research to minimize the financial risk [1].

The motivation behind this study lies in the recognition of the growing importance of precise stock price forecasts and the increasing role of machine learning in this endeavor. With rapid advancements in machine learning technology, there emerges an opportunity to not only understand the theoretical underpinnings of these techniques but also to provide practical guidance for aspiring practitioners. While previous research has contributed valuable insights into machine learning for stock price prediction, there remains a distinct need to distill this knowledge into a cohesive set of guidelines that can serve as a structured entry point for newcomers in the field. Therefore, the main objective of this paper is to provide practical guidelines for beginners entering the field of machine learning for stock price prediction instead of just providing the knowledge of machine learning theories or comparing algorithm advantages and disadvantages.

This research paper aims to bridge the gap between theoretical knowledge and practical implementation in the domain of stock price prediction using machine learning techniques. The primary objective is to conduct an review of 24 research papers published between 2010 and 2023, focusing on their implementation of diverse machine learning methods, data processing techniques, and their demonstration of results, algorithms, or data processing processes. These papers have been meticulously selected to provide a comprehensive understanding of the field.

The subsequent sections of this research paper will comprise a thorough exploration of the relevant literature, covering fundamental concepts in stock price markets, the intricacies of machine learning, evaluation metrics, and data processing techniques. Following the literature review, the methodology section will explain the approach taken to conduct this study, including detailed explanations of paper selection criteria and the process for extracting valuable information from each selected research paper.

The discussion section will provide the findings from the reviewed research papers, synthesising of their findings, insights, and methodologies. Finally, the conclusion section will summarize of key takeaways, highlight the limitations of existing research, and propose potential avenues for future exploration.

2. LITERATURE REVIEW

This session will explore the term that related to stock, normal method for analysising the stock price based on historical data, the machine learning techniques for predicting stock price and some techniques to analysis the accuracy of machine learning method.

A. Stock

The stock market is a place where shares of publicly held companies are bought and sold, involving both formal exchanges and over-the-counter marketplaces. It facilitates interactions between allows buyers and sellers, price discovery, and serves as an indicator of the economy's health [2].

The stock market performs several vital functions, such as ensuring transparency in prices, maintaining liquidity, and enabling fair dealings. It caters to various types of traders, including investors, traders, market makers, speculators, and hedgers [2].

The stock market holds great significance in a free-market economy. It allows companies to raise capital by offering shares to investors. Investors, in turn, get the opportunity to participate in a company's financial success, earning profits through capital gains and dividends. The stock market also plays a crucial role in channeling savings and investments into productive ventures, contributing to the overall economic growth of the country. Stockbrokers, portfolio managers, and investment bankers are essential in helping investors navigate the stock market by facilitating stock transactions and representing companies in various financial activities[2].

B. Stock Price Analysis

Stock analysis is a method used by investors and traders to make smart decisions about buying and selling stocks. It involves looking at past and present data to determine the real value of a stock and gain an advantage in the markets. Investors use things like financial statements, stock price movements, market indicators, and industry trends to help them make these decisions.

However, stock analysis has some limitations. First, it relies on historical information, and the future can be unpredictable, which makes projections uncertain. Also, some companies may not share all the important information, and analysts might have biases that affect their analysis. Plus, stock analysis is complex and time-consuming, requiring constant monitoring of changing factors [3].

There are two main methods for performing stock analysis: fundamental analysis and technical analysis. Fundamental analysis involves looking at financial statements, economic reports, company assets, and market share to see how healthy a company is and how much it might grow. On the other hand, technical analysis focuses on past and present price movements to predict future trends. This method uses charts and technical indicators to help with the predictions.

To answer the question which stock analysis technique is best, there's no one-size-fits-all answer. Different techniques work for different investors and situations. Some people like to use a mix of fundamental, technical, and quantitative analysis to make the best decisions [3].

Another question people often have is how to know if a stock's price will go up. It's tough to predict exactly what a stock will do. But investors can analyze information about the stock, like its fair value and how people feel about it in the market, to make smarter choices [3].

To start stock analysis, the way to begin is by gathering public information about a company, like its financial statements, news articles, and how it compares to other companies in the same industry. This can give insight on how the company is doing.

To research stocks before buying, it's essential to collect a lot of information. Some documents to research include government filings, news, what people are saying on social media, and, of course, the company's financial statements. It can also be helpful to check what other analysts are saying about the stock. This thorough research process will help form a well-informed investment decision.

C. Fundamental Analysis

Fundamental analysis is a comprehensive approach used to evaluate investments by examining publicly available financial data in order to determine whether a stock or security is fairly valued by the market. This analysis is conducted from a macro to micro perspective, beginning with an assessment of the overall state of the economy, followed by an evaluation of the strength of the specific industry, and culminating in a detailed examination of the financial performance of the company issuing the stock. The main objective of fundamental analysis is to determine a reasonable market value for the stock based on its underlying financial data and growth potential.

Investors employ fundamental analysis for several reasons. Firstly, it aids in the identification of stocks that may be undervalued or overvalued by the market, offering potential buying or selling opportunities. Secondly, it provides valuable insights into the financial health of a company and its growth prospects, enabling informed investment decisions. Moreover, fundamental analysis facilitates a comparison of a company's performance with that of similar companies in the industry [4].

When conducting fundamental analysis, analysts utilize a combination of quantitative and qualitative data.

Quantitative fundamentals involve numerical data such as revenue, earnings, profit margins, and various financial ratios. On the other hand, qualitative fundamentals encompass aspects such as the company's business model, competitive advantage, quality of management, corporate governance policies, and prevailing industry conditions.

To perform fundamental analysis, analysts rely heavily on financial statements like income statements, balance sheets, and cash flow statements. These financial documents provide crucial information about a company's financial performance over a specific period. Additionally, analysts may consult government agency reports on industries and the economy, as well as market reports, which serve as valuable tools in their analysis [4].

Below is a list of example tools used in fundamental analysis:

- Financial statements: Income statements, balance sheets, and cash flow statements.
- Financial ratios: Key metrics derived from financial statements, such as price-to-earnings ratio (P/E), return on equity (ROE), and debt-to-equity ratio.
- Government agency reports: Economic indicators like consumer price index, gross domestic product growth, and interest rates.
- Industry analysis: Reports and metrics specific to the industry in which the company operates.
- Corporate governance assessment: An evaluation of a company's policies and practices with a focus on transparency and shareholder interests.
- Company reports and press releases: Valuable insights into the company's activities, goals, and overall performance.

D. Stock Technical Analysis

Technical analysis involves using various tools and charting techniques to evaluate investments by analyzing statistical trends, such as price movement and trading volume. This method helps traders identify short-term trading opportunities and assess a security's strength or weakness in comparison to the broader market.

Professional analysts often combine technical analysis with other research methods, while retail traders may rely solely on price charts and statistics. Technical analysis is applicable to any security with historical trading data, including stocks, futures, commodities, fixed-income securities, and currencies [5].

Some examples of technical analysis indicators and their uses include:

- Price trends: Identifying upward, downward, or sideways movements in prices.
- Chart patterns: Recognizing formations like head and shoulders, double tops, or triangles.
- Volume and momentum indicators: Analyzing trading volume and price momentum to validate trends.
- Oscillators: Indicating overbought or oversold conditions in the market.
- Moving averages: Smoothing price data to reveal trends over specific periods.
- Support and resistance levels: Identifying price levels where securities tend to rebound or stall.

Fundamental analysis differs from technical analysis as it focuses on evaluating a company's financial statements, economic conditions, and management to determine the intrinsic value of a stock. In contrast, technical analysis primarily analyzes price and volume data, assuming that all known fundamentals are already reflected in the stock's price.

However, technical analysis has its limitations. Critics argue that it may not always provide actionable information, similar to the weak and semi-strong forms of the Efficient Market Hypothesis (EMH). Historical price patterns may not accurately predict future movements, and relying solely on technical analysis signals cannot influence the long-term price trajectory of an asset.

In conclusion, while technical analysis is a valuable tool for traders and analysts, it should be used in conjunction with other research methods to make well-informed investment decisions.

E. Short Term Investor

Short-term or day traders are individuals who take advantage of quick price movements in financial assets, such as stocks. They aim to profit from short-term swings in the market, typically holding positions for a brief period, often within the same trading day.

Day traders need to act swiftly and decisively. They closely monitor the market, execute trades promptly, and stay mindful of potential risks. Since their trades are short-term, they rely heavily on technical analysis using various indicators to make timely decisions [6].

Here are some common technical indicators used by day traders:

- Moving Averages: Day traders often use Moving Averages (MA) to guide their trading decisions. MAs help identify trends by smoothing out price fluctuations over a specific period. Combining different types of MAs, like simple, exponential, weighted, or smoothed, can offer valuable insights into potential entry and exit points [6].
- Relative Strength Index (RSI): RSI is a momentum oscillator used to measure the speed and change of price movements. It ranges from 0 to 100, with readings above 70

indicating overbought conditions and readings below 30 indicating oversold conditions. Day traders use RSI to spot possible buying or selling opportunities based on overbought and oversold levels [6].

- Stochastics: This momentum oscillator assesses the closing price's location relative to the high-low range over a set number of periods. It helps day traders determine overbought and oversold conditions, similar to RSI. Readings above 80 suggest overbought, while readings below 20 suggest oversold conditions [6].

- Average Directional Movement Index (ADX): ADX consists of plus and minus directional indicators. It helps determine whether a trend is forming, which is crucial for identifying potential trading opportunities during breakouts [6].

- Bollinger Bands: Bollinger Bands consist of bands placed above and below the moving average. These bands expand and contract with changing market volatility. A move outside the bands is significant and can signal potential trading opportunities [6].

However, it's important to recognize that these indicators have limitations. They cannot predict future price movements with certainty, and solely relying on them may lead to missed information or false signals. Day traders must also consider external factors like news events and market sentiment, which can influence short-term price movements. Additionally, high-frequency trading can increase transaction costs and may not be suitable for all traders due to the need for constant monitoring and quick decision-making. As with any trading approach, there are inherent risks, so traders should exercise caution and proper risk management.

F. Long Term Investor

A long-term investor is someone who holds onto their investments, like stocks, for many years. They focus on the potential growth and performance of their investments over time rather than short-term market fluctuations.

When choosing investments, long-term investors look for strong companies with good growth potential. They consider the company's financial health, business model, management team, competitive advantage, and industry trends before making a decision. Short-term price movements are less important to them as they focus on the company's long-term prospects and value.

Three simple technical indicators are commonly used by long-term investors:

- Bollinger Bands: These are trendlines drawn above and below a 20-day average of a security's price. When the price touches the bottom line, it is considered oversold, and when it touches the top line, it is considered overbought. Long-term investors use the 20-day average to decide when to buy (when the price goes below) or sell (when it goes above). It helps them visualize the "buy low, sell high" principle, and they find an oversold strong company more attractive for long-term investment [7].

- 200-Day Simple Moving Average: This is a crucial indicator for long-term investors as it represents a strong support level for a security price. If the price falls below the 200-day moving average, it may indicate potential risks with the company's financial health or undervaluation. This indicator helps long-term investors assess a company's overall strength and make informed decisions [7].

- Relative Strength Index (RSI): RSI measures recent price changes and shows if a security is oversold (RSI below 30) or overbought (RSI above 70). Long-term investors use RSI along with Bollinger Bands to plan their trades better. Buying shares of a strong company when both indicators show it is oversold can be a good long-term investment opportunity [7].

These indicators provide valuable information to long-term investors. Bollinger Bands help identify entry and exit points based on price volatility, while the 200-day moving average acts as a crucial support level. RSI helps evaluate if a security is oversold or overbought, aiding in well-timed investment decisions. Combining these indicators offers a more comprehensive view of a company's potential for long-term investors.

However, it's essential to know that technical indicators have limitations. They cannot predict future price movements with certainty. Long-term investors should use these indicators along with fundamental analysis for well-rounded decisions. Relying solely on indicators may lead to overlooking critical information or misinterpreting market signals. External factors like economic events or sudden news can also influence the stock market, making it challenging to rely solely on indicators for long-term investing.

G. Machine Learning

Machine Learning refers to computer learning by studying data, analyzing data and predicting outcomes. The way of achieving this is to use data and algorithms to imitate the way of human learning [8].

Machine learning algorithms are used to make predictions based on input, which is labelled or unlabeled data, algorithms will produce estimates about pattern of data [8]. Some common machine learning algorithms include:

1. Neural network: simulate human brain work. Good in recognizing pattern and play important role in application such as language transaction and image recognition [8]

2. Linear regression: predict numerical value based on linear relationship [8].

3. Decision trees: can be used to predict value and classify data into categories [8].

H. Regression

Regression is a predictive modeling technique that predicts continuous outcomes based on relationships between features and outcomes. It is widely used in supervised machine learning for various purposes, including forecasting trends and outcomes. Representative labeled training data is crucial for accurate predictions. Common uses include predicting house prices, stock prices, and analyzing datasets for insights [9].

I. Linear Regression

Linear regression is a popular and straightforward machine learning technique used for making predictions. It establishes a straight-line relationship between a target variable (y) and one or more input variables (x).

The main goal of linear regression is to find the best-fitting line that reduces the difference between predicted and actual values. This is achived using a cost function, usually Mean Squared Error (MSE), to measure how well the model performs. The model's coefficients (a0 and a1) are adjusted through Gradient Descent to optimize the cost function.

To assess the model's accuracy, R-squared is employed, which indicates how well the line fits the data points and the strength of the relationship between the variables. To ensure reliable results, linear regression relies on certain assumptions, such as a linear relationship between variables, minimal multicollinearity (high correlation between input variables), uniform error distribution (homoscedasticity), normal distribution of error terms, and no autocorrelations in error terms. Meeting these assumptions is vital for creating an effective linear regression model [10].

However, Linear regression has both advantages and disadvantages. On the positive side, it is a simple and computationally efficient model for expressing the relationship between predictor variables and the predicted variable. The output of linear regression is interpretable, allowing us to understand the relative influence of predictors on the target variable when predictors are independent [11].

However, there are limitations to consider. Linear regression is overly simplistic and may struggle to capture complex real-world relationships. It assumes a linear relationship between predictor and predicted variables, which may not always hold true. Outliers can significantly impact the model, leading to less reliable results. Additionally, linear regression assumes independence among predictor variables, which can be challenging to meet in practice. High multicollinearity among predictors can result in unreliable model weights and makes it difficult to determine feature importance accurately [11].

J. Polynomial Regression

Polynomial Regression is a type of linear regression used when the relationship between variables is not a straight line but shows a curved pattern. It models this curved relationship by using higher-order polynomial terms of the independent variable.

We use polynomial regression in scenarios when the straight-line model inadequately fits the data due to its curved nature. When applying a linear model to curved data, the scatter plot of residuals shows patterns of positive and negative residuals, indicating a non-linear model might be better. The assumption of independence among independent variables is violated [12].

In Python, polynomial regression can be implemented using libraries like NumPy, Pandas, Matplotlib, and Scikit-learn. By adding higher-order terms of the independent variable in the feature space, we construct a polynomial model suited to non-linear data patterns.

Polynomial regression finds application in various real-world cases where data is non-linear, like modeling growth rates, disease progression, and distribution patterns [12].

To fit the polynomial regression model, we use Scikit-learn's PolynomialFeatures class to transform the input data into polynomial features, followed by Linear Regression to fit the model [12].

However, caution is necessary to avoid overfitting, where the model becomes too complex and doesn't perform well on new data. Regularization techniques like Lasso and Ridge regression can help penalize model complexity and prevent overfitting.

K. Multiple regression

Multiple linear regression is a statistical method used to predict the outcome of a dependent variable by considering two or more independent variables [13]. It helps analysts understand how each independent variable impacts the overall variance of the model. Multiple regression can accommodate both linear or non-linear relationship.

In multiple linear regression, the formula involves the dependent variable (yi), regression coefficients (β0, β1, β2, βp) representing the effect of each independent variable, and a random error term (ϵ). The main objective is to establish a relationship between the dependent variable and the independent variables.

There are several assumptions that must be met for accurate results in multiple linear regression:

- Linear Relationship: It assumes a linear relationship between the dependent and independent variables, which can be checked using scatterplots.
- No Multicollinearity: The independent variables should not be highly correlated, as this can create difficulties in identifying the specific variable affecting the dependent variable.
- Homoscedasticity: The error in the residuals should have a consistent variance across the linear model.

- Independence of Observations: Each observation should be independent of others, and the residuals' values should not be correlated.
- Multivariate Normality: The residuals should follow a normal distribution, which can be checked using histograms or normal probability plots [14].

Following these assumptions ensures the reliability of the multiple linear regression model and helps in making accurate predictions.

L. Bias-Variance Tradeoff in Regression: Lasso, Ridge, and Elastic Net

Lasso, Ridge, and Elastic Net are three distinct techniques used in machine learning to address the bias-variance tradeoff. Bias represents the model's underlying assumptions that simplify the target function, while variance pertains to the model's sensitivity to small fluctuations in the data.

A model with high bias tends to make more assumptions, leading to underfitting, while high variance causes overfitting by capturing noise and outliers from the training data.

To address this tradeoff, we have three regression techniques:

Ridge Regression: Ridge Regression introduces a penalty term equal to the square of the coefficients into the cost function. By controlling this penalty through a parameter lambda, the model can reduce the magnitude of coefficients to zero. This results in higher bias but lower variance, making it suitable for decreasing model complexity without reducing the number of variables [15].

Lasso Regression: Lasso Regression incorporates a penalty term equal to the absolute sum of the coefficients into the cost function. As the coefficient values increase, this penalty encourages the model to shrink certain coefficients to absolute zero. Lasso is particularly useful for feature selection, as it can set some coefficients to zero, effectively disregarding less important features. However, it may encounter challenges with collinear variables [15].

Elastic Net: Elastic Net combines the regularization of both Lasso and Ridge. It proves useful when Lasso introduces a slight bias, making the model too reliant on specific variables. By utilizing Elastic Net, we can harness the benefits of both Lasso and Ridge without their respective limitations [15].

Each of these regression techniques helps strike a balance between bias and variance, thereby enabling the construction of more robust and accurate regression models.

M. ARIMA

ARIMA, which stands for Autoregressive Integrated Moving Average, and it is a statistical model used for time series forecasting. Time series data is a series of data points collected at successive time intervals, such as stock prices over days, months, or years.

The ARIMA model consists of three main components:

Autoregressive (AR) Component (p): This component uses past values of the time series to predict future values. It assumes that the future value of the time series is linearly dependent on its past values. The "p" in ARIMA(p, d, q) represents the number of lagged observations used in the model.

Integrated (I) Component (d): This component involves differencing the time series to make it stationary. A stationary time series has constant statistical properties over time, making it easier to predict. The "d" in ARIMA(p, d, q) represents the number of times the differencing is performed.

Moving Average (MA) Component (q): This component uses past forecast errors in a regression-like model to predict future values. It assumes that the future value of the time series is related to the past forecast errors. The "q" in ARIMA (p, d, q) represents the number of lagged forecast errors used in the model.

When combined, these three components form the ARIMA(p, d, q) model, which is capable of handling different types of time series data and making accurate predictions based on historical patterns.

ARIMA models find extensive applications across industries for demand forecasting, stock price prediction, economic analysis, and more. They are effective for short-term predictions and can handle non-stationary time series data.

However, ARIMA models also have limitations. They may struggle to predict turning points in the data, and determining the appropriate values of "p," "d," and "q" often involves some trial and error or expert judgment. Additionally, ARIMA models may not perform well for long-term forecasts or time series data with seasonal patterns [16].

N. KNN

K-Nearest Neighbor (KNN) is a straightforward and widely used machine learning algorithm that operates on the principle of Supervised Learning. It is particularly useful for classification tasks, as it categorizes a new data point based on its similarity to the available categories. One significant advantage of KNN is its simplicity in implementation. It requires minimal parameter tuning and can be quickly applied to various datasets.

Another advantage of KNN is its robustness to noisy training data. Since KNN relies on the proximity of data points to make predictions, isolated noisy data points are less likely to influence the overall classification. This characteristic makes KNN suitable for dealing with datasets containing outliers or noise.

Moreover, KNN can be effective when the training data is large. Since it stores all available data, it doesn't require a lengthy training process and can quickly adapt to new data points without retraining the model. This feature makes KNN efficient in scenarios where the dataset is continuously growing or being updated.

However, there are certain considerations to keep in mind when using the KNN algorithm. One crucial factor is selecting the value of "K." The appropriate choice of "K" is essential to achieving optimal performance. Setting "K" too low, such as "K=1" or "K=2," might lead to overfitting, making the model sensitive to noise and outliers in the data. On the other hand, large "K" values could lead to underfitting, where the model may lose important patterns and result in inaccurate predictions.

Another disadvantage of KNN is the high computation cost, especially with large datasets. To classify a new data point, KNN needs to calculate the Euclidean distance to all training samples and select the "K" nearest neighbors. This process becomes computationally expensive as the size of the dataset increases, potentially making the algorithm inefficient for real-time or resource-constrained applications [17].

O. Moving Average

The Moving Average (MA) method is a widely used time series forecasting technique that plays a crucial role in smoothening out fluctuations and identifying long-term trends while reducing the impact of short-term variations. Its applications span in diverse fields, such as stock price prediction, economic forecasting, and pandemics analysis like COVID-19 [18].

The method works by using a sliding window of fixed width (w) that moves with a specified stride over the time series data. Within this window, the average of data points is computed, and the original data points are replaced with their respective average values, resulting in a new series with reduced fluctuations and noise.

There are several types of moving averages exist. The Simple Moving Average (SMA) calculates the standard mean of the values within the sliding window, while the Weighted Moving Average (WMA) assigns weights to each data point, giving more importance to recent values. The Exponential Moving Average (EMA) is a special case of WMA that applies smaller exponential weights to older values, thus prioritizing recent trends.

One significant advantage of the Moving Average method is its speed and computational efficiency, making it suitable for handling large datasets and real-time forecasting applications. Additionally, it is easy to update the model with new data points without complicating the prediction process. Moreover, MA models are interpretable and explainable, enabling stakeholders to understand the model's functioning and customize it to suit specific business needs [18].

However, there are certain considerations and limitations when using the Moving Average method. To provide accurate forecasts, a sufficient number of samples are needed to establish a reliable trend. The method may not effectively capture pattern-based long-term trends, limiting its ability to predict far into the future without retraining the model. Unlike other machine learning models, MA cannot identify relationships between variables and assign custom weights to features based on their importance [18].

P. RNN

A Recurrent Neural Network (RNN) is a specialized type of neural network designed to handle sequential data like time-series and text data. Unlike traditional neural networks where inputs and outputs are treated independently, RNNs incorporate a hidden layer that enables them to retain information from previous steps in the sequence [19].

The architecture of an RNN is similar to other deep neural networks, consisting of input and output layers. However, the key distinction lies in how information is processed and flows from input to output. In an RNN, the same set of weights is used across all time steps, and the hidden state at each step is updated based on the current input and the previous hidden state [19].

The hidden state (ht) at a given time step is computed using the formula: $h = \sigma(UX + Wh\text{-}1 + B)$, where h represents the current hidden state, U is the weight matrix for the current input (X), W is the weight matrix for the previous hidden state (h-1), and B is the bias term.

The output (Y) at each time step is calculated using: $Y = O(Vh + C)$, where Y denotes the output at the current time step, V represents the weight matrix for the output layer, and C is the bias term.

The distinctive advantage of RNNs lies in their hidden state, which enables them to remember information from past inputs, making them adept at handling sequential data effectively. Moreover, the model's parameters (W, U, V, B, C) are shared across all time steps, reducing the complexity compared to other neural networks [19].

RNNs are trained using Backpropagation Through Time (BPTT), a variation of the Backpropagation algorithm that computes and updates gradients over all previous time steps.

There are different types of RNNs based on the number of inputs and outputs in the network:

- One to One: Similar to a simple feedforward neural network, it has one input and one output.
- One to Many: It generates multiple outputs based on one input, such as image captioning.
- Many to One: It produces a single output based on multiple inputs, commonly used in sentiment analysis.

- Many to Many: Both multiple inputs and multiple outputs are utilized, frequently used in language translation.

To overcome challenges like vanishing and exploding gradients, advanced versions of RNNs have been introduced. Notable variations include Bidirectional Neural Networks (BiNN) and Long Short-Term Memory (LSTM). BiNN allows information to flow in both directions, valuable for tasks where context is crucial, like natural language processing. LSTM incorporates gates to selectively read, write, and forget information, effectively handling long-term dependencies [19].

Q. Graph Neural Networks

A Graph is a data structure that represents a set of objects (nodes) and the connections between them (edges). It serves as a powerful tool to model complex relationships and interactions among different entities [22].

Graph Neural Network (GNN) is a specialized deep learning technique designed to process non-Euclidean structured data. Non-Euclidean data refers to data that lacks fixed size or dimensionality, making it challenging to analyze using traditional deep learning methods that work well with Euclidean data, such as images with fixed dimensions.

The key concept behind GNN is to work with graphs using deep learning principles. GNNs exploit the inherent graph structure to perform computations and make predictions based on the relationships between the nodes.

Several mainstream models of Graph Neural Networks include:

- Graph Convolutional Network (GCN): GCN is built upon spectral methods, which are closely related to graph signal processing. It leverages the convolution theorem to transform signals between the time and spectral domains, enabling computations on the graph. Nonlinear activation functions are applied to the aggregated results, and multiple layers are stacked to form a neural network [22].

- Graph Recurrent Network (GRN): GRN transforms the graph data into a sequence and allows nodes to exchange information with neighboring nodes iteratively until reaching a stable state [22].

- Graph Attention Network (GAT): GAT is suitable for sequential tasks and excels in handling graphs with varying sizes. It focuses on the most crucial elements of input data and uses attention mechanisms to emphasize relevant information from neighboring nodes [22].

R. LSTM

LSTM stands for Long Short-Term Memory, and it belongs to the category of recurrent neural networks (RNNs) [20]. The primary purpose of LSTM is to overcome the limitations of conventional RNNs, which struggle with learning long-term dependencies due to issues like vanishing or exploding gradients.

In contrast, LSTM networks are specifically designed to handle long-term dependencies and accurately represent sequences in chronological order. The distinguishing feature of LSTM is its internal cell design, comprising three logistic sigmoid gates and a Tanh layer. These gates control the flow of information, enabling the network to decide what information to retain and what to discard.

The architecture of an LSTM includes a hidden layer with a gated unit or cell. Each LSTM cell takes three inputs: the present information, the previous hidden state, and the previous cell state. It produces two outputs: the hidden state and the cell state. The forget gate, one of the sigmoid layers, plays a crucial role in determining how much information from the previous cell state should be retained for the current step.

LSTM networks have found applications in various fields, including text generation, image processing, speech and handwriting recognition, music generation, and language translation. Before using LSTM models in real-world applications, they need to be trained on appropriate datasets [20].

However, LSTMs come with certain drawbacks. They can be computationally intensive and demand high memory bandwidth. Researchers are actively working on developing models capable of storing past data for even longer durations. Overfitting is another challenge with LSTMs, making it difficult to implement dropout effectively to address this issue [20].

S. Measure of the accuracy of the machine learning

Accuracy is a widely used metric in Machine Learning for evaluating classification models. It measures the percentage of correct predictions made by a model out of the total number of predictions. The accuracy formula is calculated by dividing the number of correct predictions by the total number of predictions made [21].

In simpler cases, accuracy is easy to understand and implement, making it a popular choice for model evaluation. However, in real-life scenarios, machine learning problems are often more complex. Issues like imbalanced datasets, multiclass or multilabel classification, and differing objectives can make accuracy less suitable as the sole evaluation metric.

The Accuracy Paradox illustrates a common problem with accuracy when dealing with imbalanced datasets [21] . A high accuracy score may be misleading if the model performs poorly on minority classes. For instance, in medical diagnosis, misclassifying serious illnesses can have severe consequences, even if the overall accuracy seems high.

To address these limitations, alternative metrics such as precision, recall, F-score, and confusion matrix can be utilized. These metrics provide insights into the model's performance at a class level, helping to identify weaknesses in specific areas. In multiclass and multilabel problems,

different accuracy formulas account for the complexities inherent in these scenarios.

For multilabel problems, metrics such as Hamming Score and Hamming Loss are relevant metrics where classes can have multiple labels and may not be mutually exclusive. Subset Accuracy is another metric that requires all labels to match exactly for a given sample, making it suitable for strict classification tasks.

Ultimately, selecting appropriate metrics should align with the specific problem, business requirements, and workflow to effectively measure the model's performance.

T. Accuracy score

An accuracy score serves as a metric to assess the performance of a classification model in machine learning [23]. It denotes the proportion of correct predictions made by the model on a given dataset. The simplicity of the accuracy score's calculation and interpretation has led to its widespread usage, providing a single numerical value that reflects the model's ability to make accurate predictions.

To determine the accuracy score, you require two essential components:

- Ground Truth Classes: These correspond to the actual class labels assigned to the data points within the dataset, representing the true values [23].
- Predictions Made by the Model: These are the class labels predicted by the model for the corresponding data points in the dataset [23].

The formula for computing accuracy is straightforward:

Accuracy = Number of correct predictions / Total number of predictions

Alternatively, a more formal representation involves using True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) values from the confusion matrix:

Accuracy = (TP + TN) / (TP + FP + TN + FN)

U. Logarithmic Loss

Logarithmic loss, also called log loss in some of the research papers, is a widely employed error metric in the realm of applied machine learning. Its purpose is to assess the accuracy of a model's predictions by comparing the predicted probabilities to the actual labels [24]. The log loss values are confined to the range of zero to one, with zero signifying a perfect alignment between predictions and true labels. When dealing with multi-class problems, log loss typically exhibits higher tolerance levels compared to binary classification tasks.

This evaluation metric is particularly well-suited for binary classifiers, which are systems designed to distinguish between two outcomes, such as distinguishing spam from non-spam emails. In this context, lower log loss values indicate more accurate predictions, whereas higher log loss values suggest an elevated risk of misclassification.

To calculate log loss accurately, users must define the probabilities associated with each class before applying the log loss function. The formula entails computing the logarithm of the corrected probabilities and subsequently determining the negative average of these logarithms [24].

While log loss holds substantial importance for binary classifiers, it may not be the most appropriate metric for complex multiclass classification tasks. This is due to its label-dependent nature, rendering it less precise in such scenarios [24]

The efficacy of machine learning is also reliant on data processing methods, which have the potential to simplify the classification process and minimize log loss [24].

V. Confusion Matrix

The confusion matrix is a tool used in machine learning to evaluate the performance of classification models. It represents a table with four different combinations of predicted and actual class labels [25]

To understand the confusion matrix, let's consider an analogy related to pregnancy:

- True Positive (TP):

Interpretation: The model predicted positive, and it's true.

Analogy: The model predicted that a woman is pregnant, and she is.

- True Negative (TN):

Interpretation: The model predicted negative, and it's true.

Analogy: The model predicted that a man is not pregnant, and he is not.

- False Positive (FP) - Type 1 Error:

Interpretation: The model predicted positive, but it's false.

Analogy: The model predicted that a man is pregnant, but he is not.

- False Negative (FN) - Type 2 Error:

Interpretation: The model predicted negative, but it's false.

Analogy: The model predicted that a woman is not pregnant, but she is.

In the confusion matrix, the predicted values are described as Positive or Negative, while the actual values are described as True or False.

The confusion matrix is valuable for computing various performance metrics, such as:

- Recall: It calculates how many of the positive class instances were predicted correctly.

- Precision: It calculates how many of the predicted positive class instances were actually positive.
- Specificity: Another term for True Negative Rate, measuring how well the model identifies negative samples.
- Accuracy: It measures the overall correctness of the model's predictions.
- AUC-ROC (Area Under the Receiver Operating Characteristic) curve: It provides an aggregate measure of model performance across all classification thresholds.

To calculate the confusion matrix for a 2-class classification problem, actual class labels and the predicted class labels are needed to be able to compare them to determine the TP, TN, FP, and FN [25].

W. AUC-ROC curve

The AUC-ROC (Area Under the Receiver Operating Characteristic) curve is a performance metric used in machine learning to evaluate classification models [26]. It represents the model's performance at different threshold values by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR). TPR is the ratio of correctly predicted positive instances, while FPR is the ratio of incorrectly predicted positive instances [26].

AUC, which stands for Area Under the ROC Curve, calculates the two-dimensional area under the entire ROC curve, ranging from (0,0) to (1,1). It measures the model's performance across different thresholds and provides an aggregate measure of its predictive power. A higher AUC value indicates better model performance, with a value close to 1 suggesting a good ability to distinguish between positive and negative instances.

AUC-ROC is useful in cases where the ranking of predictions matters more than their absolute values, making it scale-invariant. Additionally, it evaluates model performance without considering the specific classification threshold used, making it classification-threshold-invariant.

However, AUC-ROC is not recommended when we require calibrated probability outputs from the model or when there are significant imbalances in the costs of false negatives and false positives. In such cases, other evaluation metrics might be more appropriate.

Although AUC-ROC is primarily used for binary classification problems, it can be adapted for multi-class classification using the One vs. All approach [26]. This method involves constructing separate AUC-ROC curves for each class against the rest, enabling effective evaluation of the multi-class model's performance.

X. Mean Absolute Error

Mean Absolute Error (MAE) is a statistical metric utilized to evaluate the accuracy of predictions in regression models. It computes the average magnitude of errors by measuring the absolute difference between predicted values and actual values. MAE assesses errors without regard to their direction, which enhances its robustness, particularly for datasets containing outliers [27].

The formula for Mean Absolute Error is as follows:

$MAE = (1/n) \Sigma (i=1 \text{ to } n) | y_i - \hat{y}_i|$

Where:

n is the number of observations in the dataset.

y_i is the true value of the target variable (the actual value).

$\hat{y}_i$ is the predicted value by the regression model.

MAE is a linear score, giving equal weight to all errors, facilitating model comparison and interpretation. It is widely used in various disciplines like finance, engineering, and meteorology due to its simplicity and ability to provide valuable information about prediction errors.

Y. Mean Square Error

Mean Squared Error (MSE) is a statistical metric used to evaluate the performance of a regression model [27]. It calculates the average of the squared variances between the actual values and the model's predictions. To compute MSE, one subtracts the actual values from their corresponding predicted values, squares these differences, computes their mean, thus obtaining a single numerical representation.

The formula for MSE is as below:

$MSE = \Sigma [(Actual - Predicted)^2] / N,$

where :

Σ denotes the sum of squared differences

"Actual" is the actual value

"Predicted" is the predicted value

N is the total number of data points

MSE serves several purposes: it assesses forecast accuracy, handles positive and negative errors equally, and is sensitive to large deviations or outliers. A lower MSE indicates a more accurate model. In the given example of ice cream demand forecasts, the calculated MSE was approximately 4.67, indicating the forecast model's performance.

Additionally, Root Mean Squared Error (RMSE), the square root of MSE, is often used for easier interpretation since it shares the same unit as the original data. In regression analysis, other metrics like Mean Absolute Error (MAE) and R-squared (R2) are also used for model evaluation, each having their own strengths and weaknesses. The choice of the appropriate metric depends on the specific dataset and the problem at hand [27].

Z. F1 Score

The F1 score is a widely used evaluation metric in machine learning that combines precision and recall measuring a model's accuracy [28]. Unlike accuracy, which evaluates

overall correct predictions, the F1 score focuses on class-wise performance, making it valuable for class-imbalanced datasets. Precision measures the proportion of true positive predictions among positive predictions, while recall measures the proportion of correctly identified positive class samples.

The F1 score is calculated as the harmonic mean of precision and recall, giving equal importance to both metrics. This makes it suitable for situations where maximizing both precision and recall simultaneously is essential. The F1 score ranges from 0 to 100%, with higher values indicating better classifier performance.

To calculate the F1 score, a confusion matrix with True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN) is required. The formula for the F1 score is: F1 Score = 2 * (Precision * Recall) / (Precision + Recall).

For multi-class datasets, there are different approaches to compute the F1 score:

Macro-averaged F1 Score: Simple average of class-wise F1 scores for datasets with equal class samples.

Micro-averaged F1 Score: Metric for multi-class data distributions using net TP, FP, and FN values.

Sample-weighted F1 Score: Ideal for class-imbalanced datasets, calculating a weighted average based on class samples.

Additionally, there's the Fβ score, a generalized version of the F1 score where β is a user-defined weighting coefficient, allowing prioritization of precision or recall [28].

In Python, you can easily calculate the F1 score using the "f1_score" function from scikit-learn. The "classification_report" function provides a comprehensive list of metrics, including class-wise and average metrics. The F1 score is a valuable tool for evaluating classifier performance and is commonly used in classification tasks.

AA. Data analysis and machine learning

Data processing is the essential task of converting data from one form into a more useful format. This process involves cleaning, transforming, and preparing data for analysis. Machine learning, math, and statistics are used to automate this process. The results can be shown in various forms like graphs, videos, and tables[29].

Data processing is crucial in machine learning because it makes data ready for building models. The main steps include collecting data, cleaning it up, analyzing it, making sense of the analysis, and storing it securely. Finally, the results are presented in an easy-to-understand way[29].

To get good results in machine learning, you need high-quality and accurate data. Collecting data can be expensive and time-consuming. Organizations and researchers must decide what data they need carefully [29].

Data preparation involves getting data from different sources, analyzing it, and creating a new dataset for further work. Sometimes, data is turned into numbers for faster learning by models [29].

The data might not be easy for machines to read, so xleaning, filtering, and transforming the data to make it suitable for analysis is required[29].

Processing data involves using algorithms and machine learning techniques to follow instructions over a large amount of data with accuracy and efficiency [29].

The output stage provides meaningful results that are easy for users to understand. These results can be in the form of reports, graphs, or videos [29].

Data cleaning is a critical part of machine learning. It helps ensure the data is accurate and consistent. Removing errors and inconsistencies is important because they can affect model performance [30].

Data cleaning steps include inspecting the data's structure, checking for duplicates, and handling missing values. It's important to remove unnecessary or irrelevant observations [30].

Handling missing data is a common challenge. There are two main ways to deal with it: removing observations with missing values or imputing missing values from past data [30].

Outliers, which are extreme values that differ significantly from the majority of data, need to be addressed as well. They can negatively impact analysis and model performance [31].

Data transformation means converting data into a format that's suitable for analysis. Techniques like normalization and scaling are used to transform data [31].

Normalization and scaling are important to ensure that features with different scales do not affect the model's performance. This step helps make sure all features are treated equally [31]..

3. RESEARCH METHODOLOGY

The methodology section outlines the approach used in this research to review existing literature on stock price prediction using machine learning techniques. The objective of this study is to distill valuable insights and generate relevant guidelines as a starting point for practitioners entering the field of machine learning for stock price prediction. The selection of 24 research papers from IEEE, published between 2010 and 2023, forms the basis of this methodology.

To ensure the quality and relevance of the selected research papers, a stringent set of criteria were employed during the paper selection process. The primary inclusion criteria were as follows:

- Research papers must be peer-reviewed and published in IEEE journals or conference proceedings.
- The papers must focus on stock price prediction using machine learning methodologies.
- Papers must include some or all the practical implementations, data processing details, algorithms, and results related to stock price prediction as verification of the effectiveness of the research done by the authors.

The selection of 24 research papers was motivated by the need to compile a comprehensive and diverse set of literature that encompasses various approaches, data sources, data processing methods, and algorithms. This selection strategy ensures that the final guideline reflects a broad spectrum of machine learning techniques and practical applications.

The final step of the methodology involved synthesizing the information obtained from the analyzed research papers to generate a comprehensive guideline. This guideline serves as a valuable resource for individuals looking to initiate their journey in machine learning for stock price prediction. It includes practical recommendations, best practices, and potential pitfalls to help beginners navigate the complexities of this field.

4. RESULTS AND DISCUSSIONS

This session will discuss several case studies of existing papers on using machine learning to predict the stock price, its benefits and weaknesses and accuracy.

1. This research [32] named 'Stock Prices Prediction Using Machine Learning' focuses on predicting stock prices using machine learning techniques, particularly Support Vector Regression (SVR) and Long-Short Term Memory (LSTM). The study involves collecting daily stock price data from five companies (Amazon, Google, Tesla, Netflix, Facebook) between 2015 and 2020. The data is organized, cleaned, and used to construct and test SVR and LSTM models with a 100-day training period. Model performance is evaluated using Root Mean Squared Error (RMSE). The results indicate that LSTM generally outperforms SVR in predicting stock prices, although SVR with the radial basis function (RBF) kernel performs best for Google. The research does not explicitly discuss the suitability of these methods for short-term or long-term investors, but due to the daily prediction focus and data range, they appear more relevant for short-term investment strategies.
2. This research [33], which name " Stock Prediction and analysis Using Supervised Machine Learning Algorithms" focuses on using Supervised Machine Learning algorithms to predict stock prices, especially in the context of the pandemic and its impact on the Indian stock market. They've tried different methods like Random Forest, Decision Tree, and Logistic Regression to make these predictions.

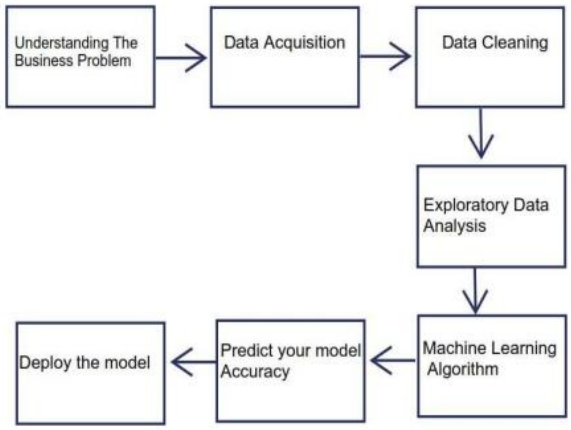

Fig -1: Step of model development in research [33]

The step of research includes, first, they got a bunch of data from Kaggle, which had information about stock prices, like when they opened, their highest and lowest points, how much they were traded, and more. Then, they used these algorithms to teach the computer how to predict stock prices. Figure 1 shows the flow chart of the step taken, which includes understanding the business problem, data acquisition, data cleaning, exploratory data analysis, machine learning algorithm, predict the model accuracy and deployment of model. The results: one of the methods they used (Logistic Regression) got an accuracy score of 52%, and the other one (Decision Tree) did better with 83%. But they didn't give more detailed figures about how well these predictions worked. However, detailed performance metrics beyond accuracy were not provided. The study did not specify whether these methods are suitable for short-term or long-term stock investment strategies. Additionally, it did not clarify the time period covered by the predictions or the timing of the research itself. The research also did not conclusively determine the effectiveness of these methods in predicting stock prices, leaving several questions unanswered.

3. This research [34] named 'Analysing the Trend of Stock Marketand Evaluate the performance of Market Prediction using Machine Learning Approach' focuses on predicting stock market values with different method with different stock. Firstly, data is gathered from different sources, including a database called Quandl, to obtain historical stock information. Then, the historical stock data is processed to create a dataset suitable for analysis. Next, the data is divided into categories like High Open, High Close, and Average Movement, making it easier to work with. Different prediction methods, including Support Vector Machine, Random Forest, and Neural Network, are tested and compared for accuracy. The results show that the Neural Network using the Levenberg-Marquardt method is the most accurate, with a reported accuracy of 94.17%.

4. The research paper [35] named 'Analysis of Stock Price Prediction using Machine Learning Algorithms' aims to predict stock prices for Reliance Industries Limited (RIL) using machine learning and deep learning techniques. The study followed several steps: first, they collected stock price data from November 11th, 2020, to November 10th, 2021, from the National Stock Exchange of India. Next, they cleaned the data by removing any missing information and focused solely on the closing prices. To evaluate their predictions, they split the data into two parts: one for training the models (80%) and the other for testing their accuracy (20%). They employed three prediction models: Linear Regression, Auto-ARIMA, and LSTM (Long Short-Term Memory). Linear Regression was used to predict stock prices based on various features. Auto-ARIMA helped with time series forecasting and calculated the Root Mean Squared Error (RMSE) to assess accuracy. LSTM, a deep learning model, aimed to capture long-term patterns in the data. The researchers found that the LSTM-based model, which used one-week historical data, was the most accurate for predicting RIL's closing prices over a 25-day period. While this research primarily focuses on short-term predictions, covering the mentioned date range, it does not provide explicit guidance for long-term investors.
5. The research [36] named 'Prediction of Stock Prices using Machine Learning (Regression,Classification) Algorithms' focuses on predicting stock prices using machine learning techniques, specifically employing regression and classification algorithms. The step-by-step process includes data collection from Yahoo Finance, where historical stock data for companies within the S&P500 index was obtained. Data preprocessing was performed to extract relevant features which is momentum and volatility. The dataset was split into training and test sets for model evaluation. Various models, including Simple Linear, Polynomial, Support Vector Regression, Decision Tree Regression, and Random Forest Regression, were implemented for stock price prediction. Accuracy results were provided for these models, with Random Forest Regression achieving the highest accuracy of 99.57%. In the classification task, Logistic Regression achieved a mean accuracy of 68.622%, and confusion matrix values were presented. The detail result of regression are presented in figure 2

Model	Accuracy	Time(in seconds)
Simple Linear Regression	81.52	0.77
Polynomial Regression	91.45	0.98
Support Vector Regression (SVR)	87.41	1.16
Decision Tree Regression	98.09	0.79
Random Forest Regression	99.57	1.06

Fig -2: Result of accuracy of Regression algorithm. [36]

For classification, the results of prediction are presented in Figure 3, which shows that SVM get the highest accuracy result.

Model	Acc.	Time(s)
Support Vector Machine (linear)	68.41	158.48
Support Vector Machine (poly)	64.80	195.38
Support Vector Machine (rbf)	67.86	201.15
Support Vector Machine (sigmoid)	58.65	160.81
K – Nearest Neighbors	61.50	19.02
Logistic Regression	68.27	10.51
Naïve Bayes	67.10	10.14
Decision Tree Classification	57.99	198.57
Random Forest Classification	63.33	202.54

Fig -3: Result of accuracy of classification algorithm. [36]

However, the author mentioned that accuracy does not represent the power of algorithms as it still depends on the data fed in. Also, the author did not provide justification about the timing and how it balances.

6. The research [37] named 'Stock price prediction based on multifactorial linear models and machine learning approaches' is focused on predicting the closing prices of nine different stocks using various prediction models and considering the impact of 18 different factors. The research process involves selecting nine stocks from different industries and collecting daily market data for these stocks from January 1, 2019, to December 31, 2021. However, the paper doesn't mention specific steps for data cleaning. It then divides the data into a training set (80%) and a test set (20%) to train and evaluate the models. However, the paper does not specifically describe what these 18 factors are. It primarily focuses on technical factors such as KDJ, RSI, Bollinger Bands (Boll), Moving Average Convergence Divergence (MACD), as well as price-related data (e.g., opening price, highest price, lowest price, etc.) as part of the 18 factors. However, the specific details or definitions of these factors are not provided in the paper. Four prediction models are used: Multiple Linear Regression (MLR), Exponentially Weighted Moving Averages (EWMA), Extreme Gradient Boosting (XGBoost), and Long Short Term Memory Network (LSTM). Accuracy is assessed using Mean Square Error (MSE) and Coefficient of Determination (R-squared) for each model. The results indicate that MLR and EWMA models show good accuracy, while XGBoost and LSTM models perform less effectively, especially when data is limited.
7. The research [38] named 'Short Term Stock Price Prediction Using Deep Learning' focuses on predicting short-term stock price movements using deep learning algorithm for ten different stocks listed on the New York Stock Exchange. It involves the use of two different neural network models, namely Multilayer Perceptron (MLP) and Long Short-Term Memory (LSTM), applied to minute-by-minute stock price data collected over a one-

year period. Data normalization techniques, which is min-max scalar were employed to ensure consistent data ranges. The study selected various technical financial indicators including trend, oscillator, and momentum indicators as features to capture different aspects of stock price movements. Both models underwent training and validation on the same dataset to prevent overfitting. The accuracy of the models was evaluated using Root Mean Squared Error (RMSE). The results of the comparison of MLP and LSTM are shown in figure 4.

LSTM		MLP	
Average Case	Best Case	Average Case	Best Case
$4.8 * 10^{-2}$	$1.88 * 10^{-2}$	$2.5 * 10^{-3}$	$9.37 * 10^{-4}$

Fig -4: RSME value of LSTM and MLP algorithm. [38]

The research concludes that MLPs outperformed LSTM in accurately predicting short-term stock prices as it had lower RMSE value. However, the applicability of these models to long-term investment strategies was not explored, given the short-term nature of the dataset. While the research suggests the potential of neural networks in predicting short-term stock prices, it does not offer a universally effective prediction method for investors and traders. Further refinement and validation may be needed to adapt these models for practical trading applications.

8. The primary focus of this research paper [54] named 'Stock Price Prediction using Machine Learning' is stock price prediction using machine learning methods, specifically LSTM and Regression models. The dataset used in the study is sourced from www.nseindia.com, covering 50 stocks from January 1st, 2000, to July 31st, 2020. While the paper describes two models - a Regression-Based Model and an LSTM Network-Based Model - it does not explicitly detail any data cleaning steps.

   The experimental results section presents the outcomes of using the LSTM model for stock price prediction, with a particular emphasis on different training epochs. Visual representations illustrate how the LSTM model's predictions compare to actual trends. However, specific accuracy values are not provided. The author concluded that increasing the epoch value can increase the precision.

9. The research [39] named 'Prediction of the Stock Adjusted Closing Price Based On Improved PSO-LSTM Neural Network.' primarily focuses on predicting stock prices, particularly the adjusted closing price, by introducing an improved model known as IPSO-LSTM. This model combines a Long Short-Term Memory (LSTM) neural network with an improved Particle Swarm Optimization (PSO) algorithm. The study involves several key steps, starting with data preprocessing, where historical stock data is divided into training and test sets and standardized to facilitate analysis. Subsequently, the model's parameters are initialized, including those for the IPSO-LSTM model. The LSTM network is trained using the training data, while the PSO algorithm optimizes crucial hyperparameters during this process. The model's accuracy is assessed using various evaluation metrics like RMSE, MAPE, MAE, and R^2, with IPSO-LSTM consistently outperforming other baseline models as shown in figure 5, the IPSO-LTSM has lower error rate and higher R-square value.

Model	RMSE	MAPE(%)	MAE	R^2
ARIMA	18.05	0.80	12.70	0.9947
LSTM	30.42	1.33	21.25	0.9848
PSO-LSTM	18.20	0.75	12.03	0.9948
IPSO-LSTM	**16.75**	**0.67**	**11.12**	**0.9958**

Fig -5: The model's accuracy is assessed using various evaluation metrics

The predicted range of the dataset is 100 days. The research demonstrates the model's robustness when applied to different stock indexes, which are the Dow Jones Industrial Average Index (DJI) and Nasdaq Composite Index (IXIC). The results are shown in figure 6, which demonstrate the gap between actual and prediction is very near.

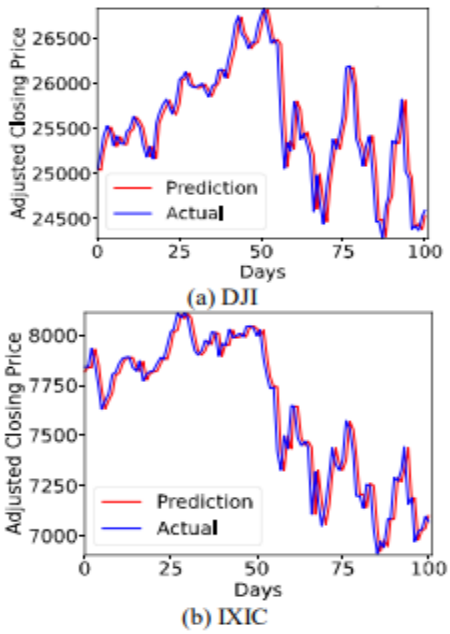

Fig -6: IPSO-LTSM performance on DJI and IXIC. [39]

10. The research [40] named 'Stock Price Prediction and Recommendation Approach Based on Machine Learning' using machine learning, specifically LightGBM, to predict and recommend stocks within the Taiwan stock market. The primary goal is to establish a systematic approach for selecting stocks and determining the best times to buy and sell them. The study draws a comparison between the performance of machine learning-driven stock recommendations and two well-known Taiwan ETFs, ETFs 0050 and 0056. The research process starts with data collection from the Taiwan Stock Exchange website, encompassing fundamental, technical, and chip-related data. Subsequently, the data undergoes preprocessing to eliminate any erroneous values. It's then segmented into training and test datasets for further analysis. Machine learning enters the scene, with the LightGBM model being trained using the prepared datasets. Following this, model settings are defined, and stocks are ranked based on the model's predictions. A specified number of the top-ranked stocks are chosen, and their performance is evaluated through back testing. As for results, it suggests that the machine learning-based approach outperforms Taiwan ETFs 0050 and 0056 in terms of annualized return and volatility.
11. The research [41] named 'Stock Price Forecasting on Telecommunication Sector Companies in Indonesia Stock Exchange Using Machine Learning Algorithms' Focuses on predicting stock prices for five telecommunications companies in Indonesia using machine learning techniques, particularly Gaussian Process and SMOreg. The study's steps include collecting historical stock price data spanning from January 1, 2017, to December 31, 2019, from Yahoo Finance and converting it into a usable CSV format. While data cleaning is mentioned, specific details on the cleaning process are absent. The dataset is then divided into training and testing sets in a 70:30 ratio. Metrics like RMSE, MAPE, and MBE are used to test accuracy. The results concluded that SMOreg outperforms the Gaussian Process.

12. The research [42] named 'Prediction of Trends in Stock Market using Moving Averages and Machine Learning.' focuses on using machine learning to improve trading signals generated by moving averages in the stock market. It aims to improve the accuracy and timeliness of these signals, particularly in relation to moving average crossovers. The data preprocessing steps of this research are illustrated in figure 7.

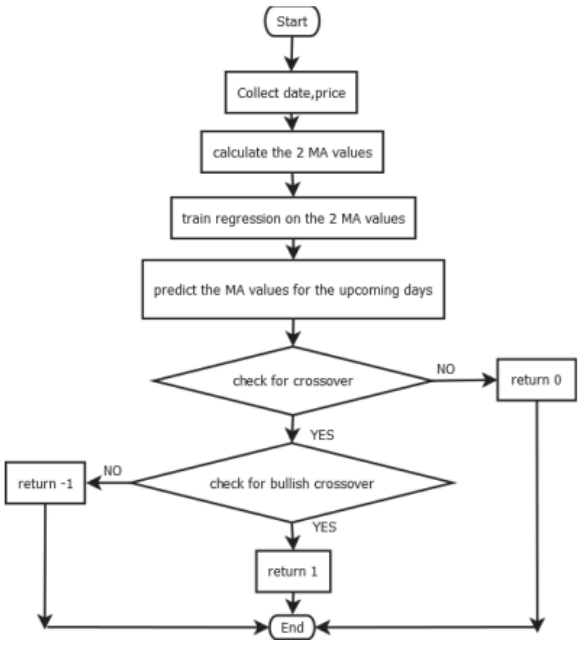

Fig -7: Flow chart of proposed model. [42]

From figure 7, it illustrates that the model will only consider the closing price and the date. Then the moving average is calculated and the data that is used to train is moving average. After that the crossover is checked. The result of research demonstrates that the proposed machine learning model can improve the accuracy and timing of trading signals based on moving average crossovers.

13. The research [43] 'A Novel Approach to Improve Accuracy in Stock Price Prediction using Gradient Boosting Machines Algorithm compared with Naive Bayes Algorithm' focuses on assessing the accuracy of stock price predictions using two machine learning algorithms: Gradient Boosting Machines (GBM) and Naive Bayes. The study involves several key steps, starting with data collection from the National Stock Exchange. Subsequently, data preprocessing is performed to clean the dataset by removing null and missing values and converting text-based data into a suitable format. The dataset is then split into a 25% training set and a 75% test set for evaluation. Both the GBM and Naive Bayes algorithms are applied to predict stock prices, and the study employs statistical analysis using IBM SPSS and Google Colab. The accuracy of the prediction methods is a pivotal aspect of the research. GBM achieves an accuracy rate of 92.3%, whereas Naive Bayes lags slightly behind with an accuracy of 87.7%. The results clearly show that GBM outperforms Naive Bayes in terms of prediction accuracy.
14. The research[44] paper named 'Enhanced Extreme Learning Machine Algorithm with Deterministic Weight Modification for Investment Decision on Indian Stocks' focuses on predicting stock prices in the Indian stock

market and introduces a new machine learning algorithm called DELM, designed to improve prediction accuracy and convergence rates. The study involves steps like using benchmark stock market datasets, including Nifty 50, S&P BSE Sensex, State Bank of India (SBIN), and ICICI Bank (ICICI). Technical indicators are used to extract relevant information from financial data, but the paper doesn't specify data cleaning steps. The accuracy of prediction methods is evaluated using metrics like RMSE, MAE, and DS. The results indicate that DELM outperforms other algorithms in terms of prediction accuracy.

15. The research [45] named 'Prediction of Stock Price Direction with Trading Indicators using Machine Learning Techniques' Focuses on predicting stock price directions using machine learning and trading indicators. The step-by-step process involved data collection from Yahoo! Finance, followed by feature extraction, labeling stocks as "upwards," "downwards," or "neutral" based on certain criteria, and balancing the dataset to ensure an equal number of records for each class. Feature elimination was also performed to improve model efficiency. Six different classification models were applied, with the Random Forest model yielding the best performance. The whole process is illustrated in figure 8.

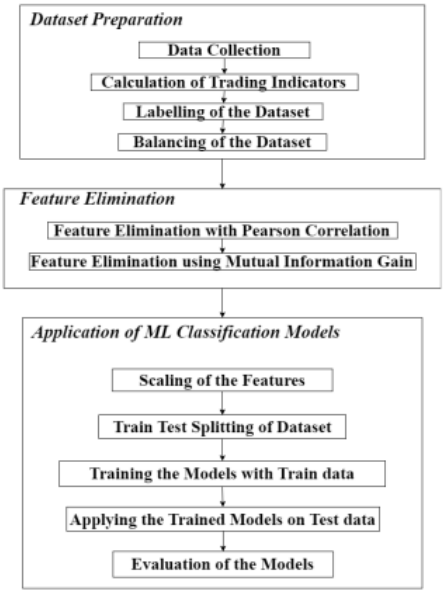

Fig -8: Flow chart of proposed model. [45]

The research conclude that all the machine learning models performed well, with Random Forest performing the best.

16. The research [46], named 'Stock Price Prediction Based On Lstm And Bert' primarily revolves around predicting stock prices for three Chinese listed companies, namely PingAn Bank, ZTE, and MuYuan. The researchers adopt a step-by-step approach in their study. Initially, they gather essential stock market data, including factors like opening and closing prices, volume, and trading amounts, spanning from January 2, 2019, to September 24, 2021. Simultaneously, they collect a substantial amount of text data from various online sources, totaling 67,981 post titles for PingAn Bank, 398,198 for ZTE, and 109,956 for MuYuan. The dataset is then divided into two parts, with 85% earmarked for training and the remaining 15% set aside for testing the model. To refine the data for analysis, the researchers employ sentiment analysis using the BERT model. This involves categorizing the sentiment expressed in the collected text data into three categories: positive, neutral, or negative. The research hinges on the implementation of a specific machine learning model called LSTM (Long Short-Term Memory) to predict stock prices. The LSTM model consists of two layers, and various standard practices, such as dropout layers, the Adam optimizer, and the Mean Squared Error (MSE) loss function, are applied in its configuration. The model's performance is assessed using several key evaluation metrics, including the Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Accuracy. These metrics provide a comprehensive understanding of the model's predictive capabilities. As results, the research concludes that BERT-LSTM model which includes sentiment analysis, will improve the performance of LTSM model.
17. The research [47] named 'A Hybrid Model for Stock Price Prediction using Machine Learning Techniques with CNN' focuses on predicting stock prices using a hybrid model that combines LSTM and CNN techniques, aiming to benefit both short-term and long-term investors. The research follows several key steps, beginning with the collection of historical stock price data from the Yahoo Finance API, which includes various indicators like opening and closing prices, high and low values, and trading volumes. While the paper mentions data scaling and dataset splitting for training and testing, it does not explicitly detail data cleaning processes, such as addressing missing values or outliers. In terms of model construction, the research builds neural network structures for both LSTM and CNN. The paper concluded that the CNN-LSTM model achieved the highest R-squared (R2) value of 0.90, which is an indicator of its accuracy. This represents an improvement of 2.8% and 0.55% relative to the other two approaches.
18. The research [48] named 'Preliminary Investigation in the use of Sentiment Analysis in Prediction of Stock Forecasting using Machine Learning' focuses on improving stock price forecasting by combining sentiment analysis with machine learning models. It conducts three experiments, testing the impact of sentiment analysis scores, historical stock price data,

and their combination on prediction accuracy. The results show that combining sentiment analysis with historical data enhances accuracy, with artificial neural network (ANN) classifiers performing best.

19. The research [1] named 'An Ensemble Learning Model Integrating Short-term Trend and Long-term Trend Used in Stock Price Forecasting' focuses on developing a stock price prediction model that enhances accuracy by combining short-term and long-term trends using the Support Vector Regression (SVR) model. The process begins with collecting historical stock price data, including opening and closing prices, high and low prices, and trading volume. After cleaning the data by removing invalid entries, a target variable is added to represent the closing price of the next day. The SVR model is then trained on this data for each company. An ensemble learning approach is introduced to combine the short-term and long-term SVR models for more accurate stock price predictions. The algorithm of combination are shown in figure 9.

Algorithm 1 Ensemble Learning Algorithm

Input: The historical statistics of the stock prices; The widow N of the short-term learning; The weight W.

Output: The prediction of the closing prices; The evaluation of the model performance.

1: *stock_df* ← Preprocess the statistics;
2: *model_l* ← Use *stock_df* to train the SVR model
3: **for** *i = N* ← *i = len (stock_df)* **do**
4: *train* ← The historical statistics of the first N days;
5: *test* ← The closing price of the N+1th day;
6: *model_s* ← Use *train* to train the SVR model;
7: predict_l ← use *model_l* to predict *test*;
8: predict_s ← use *model_s* to predict *test*;
9: predict ← *W * predict_l + (1 - W) * predict_s*;
10: **end for**
11: Evaluate the performance of the algorithm according to the predicted value and the true value.

Fig -9: Algorithm of ensemble Kearning Model. [1]

The model's performance is evaluated using Root Mean Squared Error (RMSE) and determination coefficient (R2) for various companies. The results indicate improvements in accuracy with the ensemble learning model. However, the text does not specify whether the model is better suited for long-term or short-term investors, and it does not provide details on the prediction date range or data processing date range. Additionally, the research does not conclude on specific effective prediction methods for short-term or long-term stock price forecasting.

20. The research [49] named 'Analysis and Prediction of Stock Price Using Hybridization of SARIMA and XGBoost' focuses on predicting publicly traded stock prices using machine learning techniques, specifically SARIMA and XGBoost, based on historical data from Yahoo Finance. The study proceeds through several steps: data collection from Yahoo Finance, data pre-processing involving the removal of NULL values to ensure data quality, time series decomposition to understand trend, seasonality, and noise in the data, modeling with SARIMA with some adjustments, testing the model using a separate dataset, and analyzing results for accuracy. All the step are shown in figure 10.

Step 1: Collection of historical data of the given stock value from the Yahoo Finance website. (Collected the data of Microsoft (MSFT))
Step 2: Preprocessing of data; Cleansing up of the data by removing unwanted values by checking if there's any NULL values and discarding them
Step 3: Plotting the graph based on the Adjusted closing price, Trend, Seasonality and Residuals.
Step 4: Split the data about 60% data for Training and remaining 40% for the Testing (on sequential basis).
Step 5: Build the model with the S-ARIMA initially by using the training dataset, later tune it with the XGBoost.
Step 6: Test the built model with help of the Testing Data set.
Step 7: Plot the graph with a future forecast and display the results.

Fig -10: Algorithm step of SARIMA-XGBoost hybrid model. [49]

The SARIMA-XGBoost hybrid model achieves an accuracy rate of 89.48%, with a Mean Absolute Error (MAE) of 15.612 and Mean Absolute Percentage Error (MAPE) of 10.52%. However, the research does not explicitly specify whether this predictive model is better suited for short-term or long-term investors, nor does it provide information regarding the specific date range for stock price prediction.

21. The research [50] named 'Stock Market Prediction Using Hidden Markov Model' focuses on predicting stock market fluctuations using machine learning techniques like Neural Networks, Support Vector Machine (SVM), and Hidden Markov Model (HMM). It begins with an introduction emphasizing the importance of stock market prediction, followed by a literature review that discusses the various methods used in this field. The researchers collect and prepare specific stock datasets for ICICI, SBI, and IDBI, specifying training and testing periods. They implement the HMM for stock prediction and evaluate its accuracy using Mean Absolute Percentage Error (MAPE), presenting MAPE values for the selected stocks. While the research doesn't explicitly mention data cleaning, it highlights the HMM's better accuracy compared to traditional techniques.
22. The research [51], named 'A Stock Prediction Method Based on Fake Information Identification and Machine Learning' focuses on a stock prediction method that combines fake information identification with machine learning techniques. It involves a series of steps, beginning with data collection from Twitter comments

and stock price data from the Tushare Data API. The gathered data is then pre-processed, which includes tasks like tokenization and stemming, while ensuring data cleanliness. Feature selection methods such as bag-of-words, POS tagging, and word2vec are employed to prepare the textual data for classification. Fake news classification is carried out using five different classifiers, with Logistic Regression being chosen as the best-performing model for distinguishing between true and false news based on the selected features. Additionally, the study explores the accuracy of two stock price prediction models, LSTM and GRU, with the latter demonstrating superior performance in terms of accuracy metrics. However, no detailed algorithms are provided in this research. This approach highlights the potential of integrating fake news detection with machine learning for stock prediction, showcasing GRU as an effective model for accurate stock price forecasting.

23. The research [52] named 'Recursive Stock Price Prediction With Machine Learning And Web Scrapping For Specified Time Period' focuses on using Machine Learning, specifically the Random Forest Regression algorithm, to predict stock prices. It incorporates factors such as open, high, low, and close rates, trading volume, Price to Earning Ratio, Moving Average (MA), and Moving Average Convergence Divergence (MACD) to enhance prediction accuracy. Additionally, web scraping is utilized to gather current market data. The methodology includes data collection from the National Stock Exchange, data pre-processing for cleaning and preparation, model training with Random Forest Regression, and a recursive approach for forecasting long-term future stock prices.

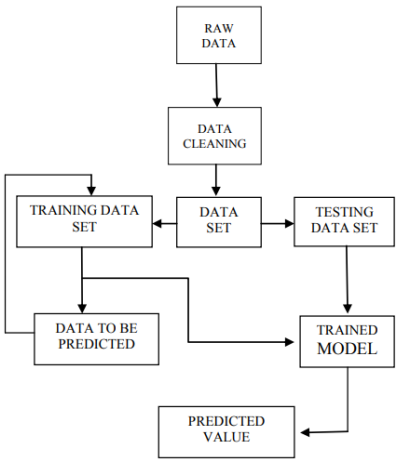

Fig -11: Structure to set up the recursive model dataset. [52]

The implementation details of the recursive model are shown in figure 11. However, the paper is only focused on implementation but not provided the accuracy of the data in long term.

The research [24] named 'Application of Singular Spectrum Analysis and Kernel-based Extreme Learning Machine for Stock Price Prediction' focuses on the application of Singular Spectrum Analysis (SSA) combined with Kernel-based Extreme Learning Machine (KELM) for stock price prediction. It aims to tackle the challenge of accurately predicting stock prices, with a particular focus on improving the speed of prediction. The study uses three different stock price datasets, including the Stock Exchange of Thailand index (SET), the Standard & Poor's 500 return index (S&P 500), and the stock market return index of Japan (Nikkei 225). In terms of data preprocessing, the stock price data is first normalized and subjected to SSA to reduce noise and improve data quality. The SSA helps in detrending the data and creating lagged matrices, which are then transformed into singular values. These reconstructed series are subsequently used for stock price prediction. For the prediction phase, the research employs the Kernel-based Extreme Learning Machine (KELM). Various parameters such as the kernel type, regularization parameter (C), and kernel parameter (σ) are selected through a grid search algorithm. All models are trained with the same parameter settings. The study evaluates the performance of different models, including SSA-KELM, SSA-LSSVM, SSA-SVM, KELM, LSSVM, and SVM, using metrics included root mean squared error (RMSE), mean absolute percentage error (MAPE), mean absolute deviation (MAD), directional symmetry (DS), and training time. The experimental results demonstrate that the SSA-based models outperform non-SSA models in terms of accuracy. Specifically, SSA-KELM exhibits the highest accuracy and the shortest training time among the SSA-based models, making it an efficient model for stock price prediction. This research indicates SSA-KELM's capability as a swift and accurate tool for stock price forecasting, although specific implementation details like algorithmic flow charts are not provided.

4. CONCLUSIONS

Drawing from the research findings and the theoretical framework of data analysis and machine learning data processing discussed in the literature review, the following conclusions can be made:

Data Collection:

Data for stock price prediction can be obtained from various sources, as observed in the reviewed research papers:

- Yahoo Finance API: This source is frequently used for collecting historical stock price data, encompassing parameters such as opening and closing prices, high and low values, and trading volumes in several research papers.

- Online Text Data: In certain studies, researchers have gathered textual data from diverse online sources. This textual data is valuable for sentiment analysis and acquiring supplementary information related to the stocks under consideration.
- Twitter Comments: In one specific research paper, Twitter comments were collected and harnessed for tasks like identifying fake information and conducting sentiment analysis in the context of stock price prediction.
- Tushare Data API: Another research paper adopted the Tushare Data API to acquire stock price data.
- Web Scraping: In the context of recursive stock price prediction, web scraping was employed as a method for data collection and preprocessing. This approach facilitated the acquisition of up-to-date market data from online sources.

Data Preprocessing:

After download the data from data source, prior to modeling, data preprocessing is imperative and includes the following steps, which were identified in the reviewed research papers:

- Data Scaling: Some papers emphasize data scaling as an initial preprocessing step. This entails normalizing or standardizing the data to ensure that all features are on the same scale, which can enhance the performance of machine learning models.
- Data Cleaning: While not explicitly detailed in some papers, data cleaning is foundational. It encompasses tasks such as handling missing values, eliminating duplicates, and addressing outliers to ensure the quality of the dataset.
- Time Series Decomposition: One research paper highlighted time series decomposition as part of data preprocessing. This step helps in understanding the underlying trend, seasonality, and noise components within time series data, which is beneficial for accurate forecasting models.
- Tokenization: In the context of sentiment analysis, tokenization was mentioned as part of text data preprocessing. Tokenization involves segmenting text into individual words or tokens, making it amenable to further analysis.
- Stemming: Stemming, as mentioned in one paper, is a text preprocessing technique that reduces words to their root or base form. This standardizes text data and reduces dimensionality.
- Feature Selection: In a research paper focused on fake news identification, feature selection methods such as bag-of-words, POS tagging, and word2vec were applied. These methods are employed to prepare textual data for classification tasks.
- Data Transformation: In the case of Singular Spectrum Analysis (SSA), a research paper employed data transformation techniques to enhance data quality. This involved detrending the data and generating lagged matrices.
- Feature engineering is a critical step in the data preprocessing phase. It involves selecting and engineering relevant features that can contribute to improved prediction accuracy.
- Techniques such as sentiment analysis, technical indicators, and news sentiment can be explored to gain additional insights.

Model Selection:

After done the data preprocessing, selecting an appropriate machine learning or deep learning model is pivotal for accurate stock price prediction. The choice of model depends on the purpose as mentioned in table 1.

Model Training and Evaluation:

Following model selection, the following matrices are critical to evaluate the performance:

- Mean Absolute Error (MAE): Used to evaluate the accuracy of stock price prediction models.
- Mean Absolute Percentage Error (MAPE): Employed to assess prediction accuracy as a percentage.
- Root Mean Squared Error (RMSE): Used to measure the accuracy of prediction models, especially in regression tasks.
- Accuracy: Evaluated in some research papers, especially when the problem involves classification of stock trends (e.g., up, down, or neutral).
- R-squared (R2): Used to measure the goodness of fit in regression models, indicating how well the model fits the data.
- Directional Symmetry (DS): Mentioned in one research paper as an evaluation metric for stock price prediction.
- Incorporating recursive techniques in the analysis can further enhance the predictive capabilities of the model. While the research papers reviewed in the previous conversation provide valuable insights into the application of machine learning and data analysis techniques for stock price prediction, there are several limitations that should be acknowledged. Firstly, the performance of the proposed models may be highly dependent on the quality and quantity of data used for training and testing. Variability in data sources and data preprocessing methods could impact the generalizability of the models. Additionally, the research papers often do not explicitly address issues related to data stationarity and non-stationarity, which are crucial considerations in time series forecasting tasks. Furthermore, the prediction horizons and investment strategies targeted by the models are not consistently defined, making it challenging to assess their suitability for short-term or long-term investment decisions. Also, users need to further study to have a deep understanding of algorithms to use it.

Future research in the field of stock price prediction using machine learning can address some of these limitations. Research can be done by using standardized datasets and evaluation metrics to enable more meaningful comparisons between different models and enhance the reliability of their findings.

REFERENCES

[1] An ensemble learning model integrating short-term trend and long-term trend used in stock price forecasting. (2020, December 1). Retrieve 8 September 2023, from IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/9532165

[2] Chen, J. (2023b). What is the stock market, what does it do, and how does it work? Investopedia. Retrieve 8 September 2023, from https://www.investopedia.com/terms/s/stockmarket.asp

[3] Chen, J. (2023a). Stock analysis: Different methods for evaluating stocks. Investopedia. Retrieve 8 September 2023, from https://www.investopedia.com/terms/s/stock-analysis.asp#:~:text=Stock%20analysis%20is%20a%20method,markets%20by%20making%20informed%20decisions.

[4] Segal, T. (2023). Fundamental analysis: principles, types, and how to use it. Investopedia. Retrieve 8 September 2023, from https://www.investopedia.com/terms/f/fundamentalanalysis.asp#:~:text=Fundamental%20analysis%20is%20a%20valuation,in%20and%20its%20financial%20performance.

[5] Hayes, A. (2022). Technical analysis: What it is and how to use it in investing. Investopedia. Retrieve 8 September 2023, from https://www.investopedia.com/terms/t/technicalanalysis.asp

[6] Staff, D. (2023, June 6). 5 Best Short Term Trading Indicators for Technical Analysis - DTTWTM. Day Trade The WorldTM. Retrieve 8 September 2023, from https://www.daytradetheworld.com/trading-blog/short-term-trading-indicators/

[7] Onigbanjo, T. (2021, December 31). 3 Simple technical indicators for long-term investing. Medium. Retrieve 8 September 2023, from https://medium.datadriveninvestor.com/3-simple-technical-indicators-for-long-term-investing-a100f02b9bed

[8] What is machine learning? IBM. (2022) Retrieve 8 September 2023, from https://www.ibm.com/cloud/learn/machine-learning

[9] Castillo, D. (2023). Machine learning regression explained. Seldon. Retrieve 8 September 2023, from https://www.seldon.io/machine-learning-regression-explained#:~:text=Machine%20Learning%20Regression%20is%20a,used%20to%20predict%20continuous%20outcomes.

[10] Linear Regression in Machine learning - Javatpoint. (2023). Retrieve 8 September 2023, from www.javatpoint.com. https://www.javatpoint.com/linear-regression-in-machine-learning

[11] Satyavishnumolakala. (2021, December 14). Linear Regression -Pros & Cons - Satyavishnumolakala - Medium. Medium. Retrieve 8 September 2023, from https://medium.com/@satyavishnumolakala/linear-regression-pros-cons-62085314aef0

[12] GeeksforGeeks. (2023). Python implementation of polynomial regression. GeeksforGeeks. Retrieve 8 September 2023, from https://geeksforgeeks.org/python-implementation-of-polynomial-regression/

[13] Hayes, A. (2023). Multiple Linear Regression (MLR) definition, formula, and example. Investopedia. Retrieve 8 September 2023, from https://www.investopedia.com/terms/m/mlr.asp

[14] Statistics Solutions. (2021, August 11). Assumptions of multiple linear regression - Statistics solutions. Retrieve 8 September 2023, from https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/assumptions-of-multiple-linear-regression/

[15] GeeksforGeeks. (2023a). Lasso vs Ridge vs Elastic Net ML. GeeksforGeeks. Retrieve 8 September 2023, from https://www.geeksforgeeks.org/lasso-vs-ridge-vs-elastic-net-ml/

[16] Science, N. B. P. a. D. (2021). Understanding ARIMA models for Machine learning. Capital One. Retrieve 9 September 2023, from https://www.capitalone.com/tech/machine-learning/understanding-arima-models/

[17] K-Nearest Neighbor(KNN) algorithm for machine learning - JavatPoint. (2023). www.javatpoint.com. Retrieve 9 September 2023, from https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning#:~:text=K%2DNN%20algorithm%20assumes%20the,point%20based%20on%20the%20similarity.

[18] A practical introduction to moving average time series model. (2023b, July 15). ProjectPro. Retrieve 9 September 2023, from https://www.projectpro.io/article/moving-average-time-series-model/716

[19] GeeksforGeeks. (2023c). Introduction to recurrent neural network. GeeksforGeeks. Retrieve 9 September 2023, from https://www.geeksforgeeks.org/introduction-to-recurrent-neural-network/

[20] What are LSTM Networks - Javatpoint. (2023). www.javatpoint.com. Retrieve 9 September 2023, from https://www.javatpoint.com/what-are-lstm-networks

[21] Bressler, N. (2023, March 23). How to check the accuracy of your machine learning model. Deepchecks. Retrieve 9 September 2023, from https://deepchecks.com/how-to-check-the-accuracy-of-your-machine-learning-model/#:~:text=Accuracy%20score%20in%20machine%20learning%20is%20an%20evaluation%20metric%20that,the%20total%20number%20of%20predictions.

[22] Zhang, W., Chen, Z., Miao, J., & Liu, X. (2022). Research on Graph Neural Network in stock market. Procedia Computer Science, 214, 786–792. Retrieve 9 September 2023, from https://doi.org/10.1016/j.procs.2022.11.242

[23] Parashar, N. (2023, January 11). What is an Accuracy Score and How to Check it? - Nilesh Parashar - Medium. Medium. Retrieve 9 September 2023, from https://medium.com/@niitwork0921/what-is-an-accuracy-score-and-how-to-check-it-13b23eeed6a3

[24] Odmark, J. (2022). What is log loss in machine learning? Pandio. Retrieve 9 September 2023, from https://pandio.com/what-is-log-loss-in-machine-learning/

[25] Narkhede, S. (2021, June 15). Understanding Confusion Matrix - towards Data science. Medium. Retrieve 9 September 2023, from https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

[26] AUC-ROC curve in machine learning - Javatpoint. (2023). www.javatpoint.com. Retrieve 9 September 2023, from https://www.javatpoint.com/auc-roc-curve-in-machine-learning

[27] Great Learning Team. (2022, November 18). Mean squared Error: Definition, applications and examples. Great Learning Blog: Free Resources What Matters to Shape Your Career! Retrieve 9 September 2023, from https://www.mygreatlearning.com/blog/mean-square-error-explained/

[28] Kundu, R. (2023, April 20). F1 Score in Machine Learning: Intro & Calculation. V7. Retrieve 10 September 2023, from https://www.v7labs.com/blog/f1-score-guide#:~:text=The%20F1%20score%20can%20be,%2Fmacro%2Fweighted%2Fnone.

[29] GeeksforGeeks. (2023b). ML Understanding Data Processing. GeeksforGeeks. Retrieve 10 September 2023, from https://www.geeksforgeeks.org/ml-understanding-data-processing/

[30] GeeksforGeeks. (2023f). ML Overview of data cleaning. GeeksforGeeks. Retrieve 10 September 2023, from https://www.geeksforgeeks.org/data-cleansing-introduction/

[31] GeeksforGeeks. (2021). ML Feature Scaling Part 1. GeeksforGeeks. Retrieve 10 September 2023, from https://www.geeksforgeeks.org/ml-feature-scaling-part-1/

[32] Stock Prices Prediction Using Machine Learning. (2021, September 23). IEEE Conference Publication | IEEE Xplore. Retrieve 10 September 2023, from https://ieeexplore.ieee.org/document/9617222

[33] Stock Prediction and analysis Using Supervised Machine Learning Algorithms. (2021, November 26). IEEE Conference Publication | IEEE Xplore. Retrieve 10 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9697162

[34] Analysing the Trend of Stock Marketand Evaluate the performance of Market Prediction using Machine Learning Approach. (2022, January 28). IEEE Conference Publication | IEEE Xplore. Retrieve 10 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9752616

[35] Analysis of Stock Price Prediction using Machine Learning Algorithms. (2022, January 21). IEEE Conference Publication | IEEE Xplore. Retrieve 10 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9725888

[36] Prediction of Stock Prices using Machine Learning (Regression, Classification) Algorithms. (2020, June 1). IEEE Conference Publication | IEEE Xplore. Retrieve 10 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9154061

[37] Stock price prediction based on multifactorial linear models and machine learning approaches. (2022, December 11). IEEE Conference Publication | IEEE Xplore. Retrieve 10 September 2023, from https://ieeexplore.ieee.org/document/10016086

[38] Short term stock price prediction using deep learning. (2017, May 1). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8256643

[39] Prediction of the Stock Adjusted Closing Price Based On Improved PSO-LSTM Neural Network. (2022, September 9). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023 , from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9941330

[40] Stock price prediction and recommendation approach based on machine learning. (2022, October 28). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10042922

[41] Stock price Forecasting on telecommunication sector companies in Indonesia Stock Exchange using machine learning algorithms. (2020, October 27). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9320758

[42] Prediction of Trends in Stock Market using Moving Averages and Machine Learning. (2021b, April 2). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9418097

[43] A Novel Approach to Improve Accuracy in Stock Price Prediction using Gradient Boosting Machines Algorithm compared with Naive Bayes Algorithm. (2022,

December 16). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10074387

[44] Enhanced Extreme Learning Machine Algorithm with Deterministic Weight Modification for Investment Decision on Indian Stocks. (2022, October 20). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9951899

[45] Prediction of Stock Price Direction with Trading Indicators using Machine Learning Techniques. (2022, December 30). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10150983

[46] Stock Price Prediction Based On Lstm And Bert. (2022, September 9). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9941293

[47] A Hybrid Model for Stock Price Prediction using Machine Learning Techniques with CNN. (2021, October 22). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9702382

[48] Preliminary Investigation in the use of Sentiment Analysis in Prediction of Stock Forecasting using Machine Learning. (2020, March 28). IEEE Conference Publication | IEEE Xplore. Retrieve 11 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9368258

[49] Analysis and prediction of stock price using hybridization of SARIMA and XGBOOST. (2022, March 10). IEEE Conference Publication | IEEE Xplore. Retrieve 12 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9767868

[50] Stock market prediction using Hidden Markov Model. (2014, December 1). IEEE Conference Publication | IEEE Xplore. Retrieve 12 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7065011

[51] A stock prediction method based on fake information identification and machine learning. (2022, October 1). IEEE Conference Publication | IEEE Xplore. Retrieve 12 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10129429

[52] Recursive Stock Price Prediction With Machine Learning And Web Scrapping For Specified Time Period. (2019, December 1). IEEE Conference Publication | IEEE Xplore. Retrieve 12 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8995080

[53] Application of singular spectrum analysis and kernel-based extreme learning machine for stock price prediction. (2016, July 1). IEEE Conference Publication | IEEE Xplore. Retrieve 12 September 2023, from https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7748873

[54] Stock Price Prediction using Machine Learning. (2022, March 16). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9752248