# Comprehensive Analysis of Machine and Deep Learning Models for Stock Market Prediction

Hadi S. AlQahtani, Prof. Mohammed J. Alhaddad, Dr. Mutasem Jarrah
Department of Information Technology-Faculty of Computing and Information Technology,
King Abdulaziz University, Jeddah, Saudi Arabia

***Abstract*—Stock market prediction is a core task in financial engineering that requires sophisticated methods to extract subtle market and volatility trends. The increasing complexity of the stock market has led to the integration of advanced machine learning (ML) and deep learning (DL) techniques to improve accuracy beyond traditional statistical methods. This research provides a taxonomy of stock market prediction methods and reviews key regression-based models, including linear regression and advanced neural networks like recurrent neural networks (RNNs), long short-term memory (LSTM), and hybrid (CNN-LSTM) models. The study deploys and evaluates three specific models: Linear Regression, RNNs, and LSTMs. The models were trained and tested using modern data preprocessing procedures, including Z-score normalization and temporal sequencing. The findings show that the Linear Regression (LR) model performed better, with a Root Mean Square Error (RMSE) of 0.334 during training and 0.304 during testing, and a Mean Absolute Error (MAE) of 0.203 and 0.207, respectively. This contrasted with the deep learning models, which had higher error rates. The LSTM achieved a training RMSE of 0.355, while the RNN model had a training RMSE of 0.383. These results provide empirical evidence that increased model complexity does not necessarily translate into better forecasting accuracy in financial applications, and that model selection is both context-sensitive and data-driven. The findings mentioned the challenge of nonstationarity in stock market data and the need to periodically retrain models on recent data.**

***Keywords*—Deep learning; machine learning; prediction methods; stock market; regression; taxonomy**

## I. Introduction

Predicting stock prices accurately is crucial for investors, financial markets, and economy decision makers because of its profound implications for investment decisions. Two traditional approaches are used for forecasting stock trends, fundamental analysis and technical analysis. Fundamental analysis consists of the assessment of the actual value of securities based on the company’s financial reports and other macroeconomic factors [1]. This method is more suitable for strategic long-term investment. Technical analysis, on the other hand, involves the use of price movements, volumes, and charts to determine the prognosis of future prices. This method is applied for short-term trading and help determine tendencies and the right time for trading operations [2].

The financial forecasting industry, particularly stock market prediction, has been greatly impacted by AI technologies as it continues to evolve rapidly. Traditional financial time series and time series analysis techniques often fail to analyze the market's complex nonlinear data, which made researchers shift their focus to more sophisticated AI technologies [3]. These include Recurrent Neural Networks (RNNs), Long Short-Term Memory networks (LSTMs), and even hybrid models such as LSTM-CNNs [4]. RNNs provide the most basic method to model temporal dependencies in sequential data, while LSTMs improve upon RNNs by introducing memory cell architectures to mitigate the vanishing gradient problem [3]. More recently, LSTM-CNN models have been developed which combine LSTMs with Convolutional Neural Networks (CNNs), leveraging both temporal and spatial feature extraction capabilities. This research focuses on the stock price forecasting AI models and aims to analyze their performance comparatively and identify the most optimal model for practical usage [4].

The traditional forecasting methods struggle to capture the complexities and uncertainties inherent in stock markets. However, AI technologies are modern methods that offer promising ways for enhancing prediction accuracy [13]. The algorithms offer the ability to automatically detect patterns, adapt to changing market conditions, process large volumes of data, including historical prices, technical indicators, financial news, social media sentiment, leveraging large datasets and detecting intricate patterns, especially machine learning (ML) and deep learning (DL) techniques [5]. By employing ML and DL algorithms such as linear regression, logistic regression, RNNs, LSTM, and LSTM-CNN, researchers have attempted to exploit the vast amount of market data to uncover hidden relationships and forecast future price movements [6]. This paper mentions stock market prediction methods taxonomy, with recent advancements in stock market prediction achieved through possible methods that can assist in stock market prediction. Generally, this paper provides insights into the effectiveness and reliability of the most important approaches in stock market prediction, informing stakeholders about their potential benefits and challenges in real-world and prediction applications.

The reminder sections of this paper are: proposed previous work in Section III as a literature review. Section IV discussed methodology, results and findings are evaluated in Section V, Critical analysis is given in Section VI and Section VII is the conclusion.

## II. Stock Market Prediction Methods Taxonomy

Stock market prediction methods are divided into three categories: fundamental analysis, technical analysis, and algorithmic approaches.

## A. Fundamental Analysis

Fundamental analysis relies on economic indicators, company performance metrics, and financial reports to assess the intrinsic value of a stock [1].

## B. Technical Analysis

Technical analysis focuses on historical price patterns, trading volumes, and chart indicators to forecast future price movements [12].

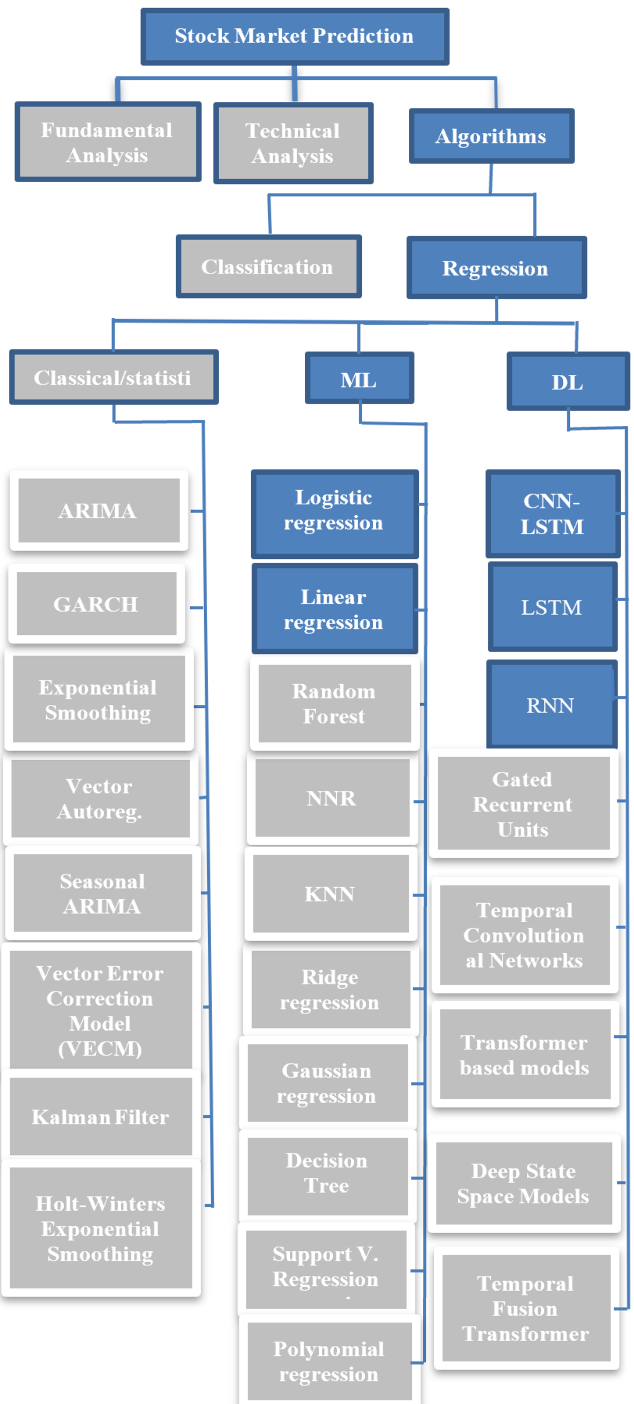


Fig. 1. Taxonomy of stock market prediction methods.

## C. Algorithmic Approaches

Algorithmic approaches employ computational models and artificial intelligence techniques to identify patterns and generate predictions. Within algorithmic approaches, methods are typically organized into two main tasks: classification (e.g., predicting whether a stock will rise or fall) and regression (e.g., forecasting the exact stock price or return value) [6].

Since the present study is concerned with continuous stock price forecasting [15], the focus is placed on regression methods. These can be further classified into three different categories [14]:

- Classical/Statistical models (e.g., ARIMA, GARCH), which provide simple yet interpretable baselines.
- Machine Learning (ML) models (e.g., Linear Regression, Logistic regression, Support Vector Regression, Random Forests), which capture nonlinear relationships and improve predictive performance.
- Deep Learning (DL) models (e.g., RNN, LSTM, CNN-LSTM), which are designed to learn complex temporal dependencies and deliver state-of-the-art results for sequential financial data.

This hierarchical taxonomy provides a structured view of stock prediction methods, showing how approaches range from traditional econometric techniques to advanced AI-driven models. It also clarifies the scope of this study, which emphasizes regression-based methods across statistical, ML, and DL Models. These three categories contain more techniques, as shown in Fig. 1.

# III. Literature Review

Predicting the stock market has attracted much interest from the scientific community, financial experts, and computer scientists. Future market prices and movements are likely to show complex nonlinear relationships with time that differ. Due to their availability and increased computational power, most technological tools are incredibly precise and elaborate.

The reviewed studies in this paper are organized into a taxonomy that highlights how stock market prediction methods are structured along the key dimensions: algorithm type

The discussed literature is limited to important machine learning (ML) approaches and deep learning (DL) approaches:

## A. Machine Learning

*1) Linear regression:* Linear regression is treated as a supervised learning algorithm for regression tasks. It is an elementary method of predictive modeling, a subset of artificial intelligence, used to establish a relationship between an outcome variable and one or several predictor variables. It assumes that the best fit is a straight line that best predicts the errors between the predicted and actual values. The conditional assumptions are linearity, the absence of multicollinearity, and the normality and homoscedasticity of residuals. Research in [7] discussed regression methods used for prediction models in machine learning. The researchers overviewed definitions of the models, metrics for model evaluation, assumptions, and methods of attribute selection. Some of the issues that were raised in the study include issues related to overfitting and the fact that while regression is a very powerful tool, it is, however, quite restricted in its ability to solve very complicated problems. Reference [8] designed a website that employed linear regression for stock price prediction. The researchers concluded that the performance on two years of historical prices provided reasonable accuracy. Authors of study [16] also noted that it is crucial to consider nonstationarity when dealing with stock data for production use. The model must be periodically retrained using more recent data to enhance its predictive ability. Research in [9] discussed how the linear regression model is trained for stock prediction using Python. The researchers explored exploratory data analysis, formulating a model, evaluating it, and incorporating it into trading strategies. Generally, they proved the existence of its kind but pointed out the dangers of oversimplification and the use of predictive modeling. Moreover, regression coefficients introduce interpretability measures to identify useful indicators [9]. Nonstationarity is one of the challenges faced by models and may be addressed by retraining on more recent sliding stock data. In general, literature supports the application of regression for stock prediction but simultaneously raises concerns about using it without proper precaution [8]. Important features are cross-validation, integration with other techniques, repetitiveness with retraining, interpretability and integration with human decision-making [14]. The greatest value is achieved when prescriptive analytics include predictions and insights into the trading systems.

*2) Logistic regression:* In machine learning and for classification tasks, logistic regression is a supervised learning algorithm. It is a specific feature of neural networks is that they require many training data points for accurate predictions. Neural networks can be used when making predictions based on binary outcomes. Logistic regression is one of the most frequently used in machine learning and can be utilized to predict a binary outcome point given predictor variable [14]. The study in [10] evaluated the abilities of logistic regression as a machine learning algorithms, including random forest and neural networks. This study revealed that logistic regression was not inferior to other algorithms after the method of choosing the number of features through the area under the ROC curve was applied [10]. This is where logistic regression proves to be a helpful interpretable baseline model. The study in [11] employed logistic regression and logitboost algorithms to build churn prediction models. Logit had a slightly better performance than logistic regression, although this came at the cost of performing hyperparameter tuning. Logistic regression provides an excellent and less complex solution to the model. Authors of [17] considered the rationale for employing logistic regression or linear regression analysis when analyzing treatment effects in experimental research with binary data. The study in [11] concluded that logistic regression was reasonable for disentangling treatment impact from risk profiles relevant to the outcome, even in situations of imbalanced randomization. Generally, the reviewed research proves the suitability of the logistic models for causal conclusions. Logistic regression has several benefits, namely, interpretability, efficiency, and reliability of the results. Coefficients of logit show how each predictor is associated with the outcome variable [11]. Fewer tuning parameters are needed than in other complex machine learning methods. Logistic regression also performs well when analyzing small datasets and has no overfitting issue. In fact, research [10] has demonstrated that it can do so to estimate treatment effects while also accounting for confounding factors. However, the assumptions accompanying most logistic regression models, such as linearity and multicollinearity, may be violated in real-world data [10]. The algorithm cannot, in the process, model multiplicative complex nonlinear effects or high-order interactions as with tree-based methods. Another disadvantage is that logistic regression is very sensitive to the choice of the classes where the classes are imbalanced. Even if more variables are added to the model, the algorithm may not increase its predictive accuracy if the relationship is not linear or, in other words, if it is nonlinear in nature [11]. These limitations justify the need for superior sampling and variable selection techniques in the study designs. Machine Learning techniques reviewed sources of Linear Regression and Logistic Regression are summarized in Table I.

TABLE I. EVALUATION SUMMARY OF MACHINE LEARNING TECHNIQUES

| Ref. | Methodology | Performance metrics | Strengths | Limitations | Result |
|---|---|---|---|---|---|
| [7] | Support vector polynomial regression algorithm | Mean absolute error (MAE) | Used number of transactions missing to calculate its performance | Used algorithms may not increase its predictive accuracy and need test it on other algorithms | SVM accuracy: 93% |
| [8] | Linear regression, basic but fundamental | Mean absolute error (MAE), root mean square error (RMSE) | Simplicity, easy to implement | Limited complexity in modeling | Exceeded 85% |
| [9] | Regression analysis | MSE, RMSE, R-squared | Practical application | Limited to simple models | Accuracy 76% |
| [10] | Comparative study of logistic regression and ML | Recall, F1 Score, AUC | Simple and Direct comparison | Algorithm not work as expected with nonlinear models, in addition to that, it is very sensitive to the choice of the classes | N/A |
| [11] | Logistic regression and Logit boost | Recall, F1 Score, Area Under the Curve (AUC) | Focused on binary classification | The algorithm may not increase its predictive accuracy if the relation is not linear | Accuracy more than 85% |

*B. Deep Learning*

*1) Recurrent Neural Networks (RNN):* RNNs are a type of deep neural network that is very useful when processing data in a sequence, such as time series data such as stock prices [17]. The recurrent connections allow RNNs to capture temporal dependencies within the series of observations. The study in [18] discussed deep learning stock forecasting models and noted that RNNs/LSTMs can identify temporal dependencies that are not extractible by other models. The inherent recurrent structure of RNNs allows the design of a model of sequential correlations unavailable to feedforward networks. Nonetheless, it is difficult for vanilla RNNs to handle longer sequences because these models have issues with exploding/vanishing gradients [19]. RNNs with memory solve this problem using gated memory cells that help with long-term dependency. In conjunction with recent techniques in deep learning, RNNs offer the best performance in modeling most sequence data. Due to their ability to identify temporal relationships, RNNs with memory models are useful for predicting the next values of time series such as stock prices. RNNs with memory networks for stock price prediction are rooted in enhanced capacity to handle time series data appropriately. The study in [20] compared RNN LSTM with other models, such as linear models and random forests, especially from the perspective of stock price forecasts. The source takes up the problem of a lack of effective tools for risk forecasting in the context of the fluctuating financial market, as well as implementing sound research methods to determine the efficiency of the models [20]. A weakness with all recurrent neural networks is the vanishing gradient problem, but LSTMs are especially lauded for their performance with long-term dependencies in data, which is useful since stock price data are a sequence. This model is even better than regular RNNs because it has memory cells that control the flow of data, and the cells have gates that control the addition and accumulation of data, which tend to solve problems such as vanishing gradients, which are common with regular RNNs.

*2) Long Short-Term Memory (LSTM):* LSTM is a type of recurrent neural network that might be most effective in modeling time series data. Since their introduction, LSTMs have been implemented in the financial field, where they have been used in stock price prediction, algorithm trading, risk control, and fraud identification. Reference [19] discussed various prediction methods employed in stock markets, including LSTMs. They discovered that LSTMs are more effective for variable financial time series where the data contain nonlinear trends and are nonstationary. They enhance the algorithm to increase its accuracy. Authors of study [21] introduced a four-layer architecture of a dual-LSTM model using past prices and technical indicators to forecast multiple-day-ahead stock prices. The author's approach offers one of the best solutions for benchmark datasets. Accordingly, LSTMs have significant advantages and are suitable for modeling financial time series. It can remember long-range dependencies missing in feedforward networks through their memory cells. Some of the main issues that are particularly relevant to LSTM-based financial applications include data noise, model interpretability, and the model's ability to generalize to distribution shifts with respect to the time series. Research has provided a new mixed option comprising long short-term memory (LSTM) and an adaptive genetic algorithm (AGA) for the prediction of stock indices, further stressing innovation in the spheres of financial forecasting [12]. The study in [22] motioned progressive developments in predictive analytics in financial markets and, thus, considered the current state of the markets. From a methodological perspective, the integration of LSTM with AGA is focused on optimizing the networks' structures and parameters to improve the forecast's accuracy accordingly [22]. Performance metrics such as the mean square and absolute percentage errors are used to check the model's effectiveness. One of the advantages of this study is the authors' effective method of tracking changes in data from one period to another and adjusting for such changes in the analysis, thus enhancing the forecast accuracy. However, the requirements of the components to work out the model result in a more complex design and higher processing needs, which may become an issue of practical application [22]. The results show that it has better accuracy than traditional models do, making it possible to fine-tune investment portfolios [22].

*3) CNN-LSTM:* The combination of CNNs and LSTM produced a new model that uses both a convolutional neural network (CNN) and LSTM networks to benefit from each other's strengths and make a proper model for time series forecasting. The CNN down samples the input time series to extract local spatial relations, whereas the LSTM learns long-term temporal relations. The CNN is used to extract features, and the output sequence is taken from the CNN and passed to the LSTM at the back end, which is considered for temporal analysis. The CNN features in the network act as translation invariants, and the LSTM capability is used to store the temporal state. A few recent studies have implemented the CNN-LSTM architecture across different fields, such as stock market prediction and electricity load forecasting [20]. Authors of study [23] used a CNN-LSTM to input multivariate financial time series data and established that this improves the performance of a pure LSTM.

*4) Analysis of the CNN-LSTM method* shows that the CNN-LSTM method for financial time series forecasting takes advantage of convolutional neural networks and long short-term memory networks [23]. It can identify spatial correlations in the data and temporal correlations to make the best prediction. One advantage is that the weights of a CNN are made regular before feeding the LSTM layer, which increases its ability to resist variations [24]. Long-term dependencies may remain a task for a model that can still be challenging compared with more straightforward methods of statistics. Thus, CNN-LSTM has the potential to refine the accuracy of forecasts; however, further studies are still needed in terms of interpretability, versatility for different datasets, and optimization of feature extraction strategies. Further tweaking and testing it on live data would help achieve its real-life applicability. Simple modifications in the structural design of models and training algorithms could effectively unlock the benefits of this integrated deep learning approach to modeling financial series. Deep Learning techniques reviewed sources of RNN, LSTM, and CNN-LSTM are summarized in Table II.

TABLE II. EVALUATION SUMMARY OF DEEP LEARNING TECHNIQUES

| Ref. | Methodology | Performance Metrics | Strengths | Limitations | Result |
|---|---|---|---|---|---|
| [21] | Dual-LSTMs, advanced for real-time data | RMSE, Mean Absolute Deviation (MAD) | Capable of handling complex patterns | Complexity in tuning and training | Best average prediction errors of LSTM = 0.05 |
| [19] | Etaheuristic Optimization on RNN–LSTM | MAE, R-squared, Accuracy, Precision | Wide scope by using more datasets | Lacks specific practical insights | Accuracy after using RNN–LSTM increment of 4–6% |
| [23] | Multivariate CNN-LSTM model | MSE, RMSE, MAE | Handles multiple data streams | Complexity and implementation difficulty | CNN-LSTM Average value of RMSE = 0.0162 |
| [22] | Uses LSTM with sentiment analysis from social media. (TLSTM) | RMSE, MAE, Accuracy, Precision | Combines technical analysis with sentiment analysis for better predictions | Model Complexity | RMSE= 2.147, 82.19 and F-measure = 89% |

## IV. METHODOLOGY

### A. Research Design and Framework

The study employs a quantitative experimental design to determine the performance of machine learning and deep learning techniques for stock market prediction. The methodology outlines the procedure as a six phase process: data collection, preprocessing, models implementation, model training/testing, and results evaluation. This study will only consider the performance of the Saudi Aramco (2222.SR) stock. Saudi Aramco is the largest oil company in the world, and it has considerable influence over the regional and global financial markets [1].

The experimental framework was structured to allow reproducibility and statistical reliability due to a comparison of three modeling approaches. The first Linear Regression model was used as a baseline model. The second, RNN, was a first-generation model based on basic sequential learning. The third, LSTM, was a first-generation model based on sequential temporal dependencies. This multi-model approach allows for assessing performance at different levels of complexity and computational overhead. In addition, it provides a better understanding of how machine learning methods perform compared to the more up-to-date deep learning architectures in forecasting applications within finance.

### B. Data Collection and Acquisition

Historical stock data for Saudi Aramco was collected systematically using Yahoo Finance API (yfinance) Python library over a total of last five years. The dataset captures daily trading activity which includes opening price, closing price, high-low range for the day, volume of trades and is therefore complete in terms of transaction history. Since the target stock symbol was 2222.SR, it ensured that the data collected and returned was specifically from the Saudi Stock Exchange (Tadawul). Generally, error handling and data validation were built into the data collection process to capture concerns regarding completeness and integrity of the data returned. Missing values were identified and subsequently addressed using forward and backward interpolation as appropriate, when generating daily averages from weekly averages. Outliers were identified through statistical measures and determined whether to keep/discard outliers based on trading volumes and more importantly, where the trades were made with respect to the market, bid-ask and consensus price at that point in time. Overall, the comprehensive dataset provides a great way to train a model, and based on the extensive and complete dataset provided, all empirical results should reflect the actual market conditions and trading behaviors observed in the Saudi Arabian equities market.

### C. Preprocessing

The preprocessing stage included many data standardization methods to improve how well the model converges with optimal performance. The data standardization techniques included standardization of the input features utilizing Z-score normalization method to standardize all input features using the equation, as shown in Eq. (1):

$$Z = (X - \mu) / \sigma \tag{1}$$

Where Z is the standardized value of X, the original value, μ is the mean, and σ is the standard deviation of the feature. The standardization of the input features is important in optimizing a neural network as it allows the converging of the cost function to occur more efficiently [14]. This ultimately reduces the required number of epochs to train the model and improves model reliability. The preprocessing pipeline developed the temporal sequences for the sliding window techniques in RNN and LSTM models. The construction of temporal sequence input patterns using sliding windows, enabled the models to learn from historical price movements. All numerical features underwent feature scaling to develop an ability to separate features within differing magnitude therefore preventing bias to larger magnitude features. The dataset was divided into a training and testing dataset, the training dataset consisted of 80% and the testing dataset consisted of 20% of the total data, used chronological splitting methods to prevent any leakage of future data which ensured the temporal characteristics of the dataset remained intact throughout the inception of the models.

### D. Model Performance Evaluation Metrics

The model performance was assessed using two main evaluation metrics. The first evaluation metric was Root Mean Square Error (RMSE), which was calculated as shown in Eq. (2):

$$RMSE = \sqrt{}(\Sigma(y_i - \hat{y}_i)^2 / n \tag{2}$$

The second evaluation metric was Mean Absolute Error (MAE), calculated as shown in Eq. (3):

$$MAE = \Sigma|y_i - \hat{y}_i| / n \tag{3}$$

where $y_i$ represents the actual values, $\hat{y}_i$ represents the predicted values, and n is the number of observations that were captured each day. After evaluating all models, meaningful differences in predictive performance were observed across models, with results systematically recorded to analyze prior to determining the best approach for predictive modeling.

### *E. Model Implementation and Architecture*

The Linear Regression model is the baseline with which everything else can be compared. It implements the simplest relationship in the literature, which is shown in Eq. (4):

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n + \varepsilon, \tag{4}$$

has easy-to-interpret coefficients, and is easy to compare with this work. The Linear Regression model was implemented using scikit-learn's Linear Regression class with all parameters set to the default (that is, no user-supplied input / adjusted parameters for this baseline model) to standardize the evaluation conditions. The implementation of the RNN architecture was carried out using the TensorFlow/Keras framework. It defines sequential data with the appropriate number of timesteps in the input layer, an RNN layer that has 50 hidden units, a dense layer that contains the predicted closing price, an Adam optimizer with a default learning rate set to 0.001, and finally a Mean Squared Error loss function. The RNN model used a basic architecture which was capable of looking at (and capturing) some short-range temporal dependencies, which may still have inherent issues with long range dependencies due to vanishing gradient issues before the learner reaches the next timestep [21]. The LSTM structure uses memory cells and gating mechanisms to solve RNN issues. It has an LSTM layer containing 50 memory units, a dropout layer (0.2) to regularize, and a dense final output layer for prediction. It also has the same optimizer and loss function as RNN. The LSTM design allows for learning long-term dependencies using a more sophisticated gating architecture, including forget gates, input gates, and output gates [21].

### *F. Statistical Validation, Robustness Testing, and Computational Environment*

This study employed a rigorous method to implement and test machine learning models, including linear regression, with machine learning versus strictly statistical. The intention was to determine models that yield the most stable predictive accuracy to be used specifically in the Saudi Arabian equity market. Cross-validation tested the generalization capacity of each model by maintaining training and test data distinct to avoid leakage and attain a realistic performance estimation. Sensitivity analysis tested model stability against small to moderate perturbations in hyperparameters. Statistical significance testing determined whether differences observed in performance were statistically significant, for example, linear regression versus other models. Robustness testing assessed the performance of models under different market conditions and volatility regimes to assess generalizability. The performance of the models were assessed using the same computational resources; all experiments used Python 3.8+ with the same libraries (TensorFlow 2.x, scikit-learn, pandas, and numpy). Random seeds were established and hyperparameter values were recorded as well as calculated to ensure reproducibility. Each model had the same hardware specifications, details of all training modalities were finalized and recorded to allow full transparency and facilitate future model improvements.

## V. RESULTS AND EVALUATION

The investigation findings provide a comparative evaluation among three predictive methods for forecasting Saudi Aramco stock prices: Linear Regression, Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN). The training and testing datasets were evaluated using two relevant regression measures: Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). In the training phase, Linear Regression was again the best-performing method with the lowest RMSE (0.334) and MAE (0.203). For these metrics, linear regression was performed comparably on a direct comparison basis with the actual price. LSTM at RMSE 0.355 and MAE 0.266 for the remaining predictive methods, and RNN at RMSE 0.383 and MAE 0.263. Training Results prediction performance for Aramco Stock Market is summarized in Table III.

Thus, although the RNN and LSTM are deep learning computational models that model temporal dynamics such as time series data and patterns over time, and deeper network connections will provide better mathematical representation, they did not outperform the Linear regression-based training performance.

More importantly, during the testing phase, which indicates how the model performs on unseen data, Linear Regression (LR) remained ahead of the pack, with an RMSE of 0.304 and MAE of 0.207, as shown in Table IV and drawn in Fig. 2. LSTM's RMSE was 0.346 and MAE of 0.237, as shown in Table IV and drawn in Fig. 3, while RNN performed the worst with an RMSE of 0.415 and MAE of 0.329, as shown in Table IV and drawn in Fig. 4. Testing results analysis for Aramco stock market are summarized in Table IV.

TABLE III. TRAINING RESULTS PREDICTION PERFORMANCE OF ARAMCO STOCK MARKET

| Model Name | RMSE | MAE |
|---|---|---|
| Linear Regression | 0.334 | 0.203 |
| LSTM | 0.355 | 0.266 |
| RNN | 0.383 | 0.263 |

TABLE IV. TESTING RESULTS ANALYSIS OF ARAMCO STOCK MARKET

| Model Name | RMSE | MAE |
|---|---|---|
| Linear Regression | 0.304 | 0.207 |
| LSTM | 0.346 | 0.237 |
| RNN | 0.415 | 0.329 |

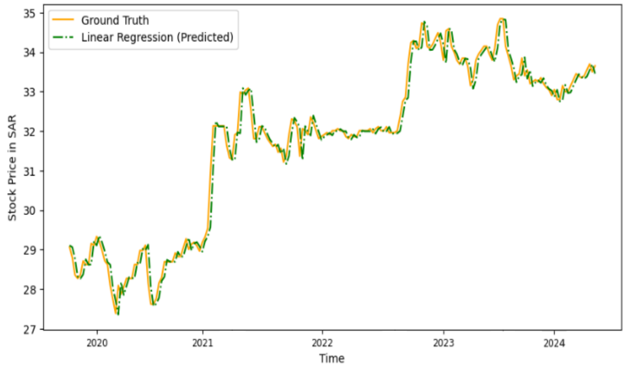


Fig. 2. LR prediction performance of Aramco stock market.

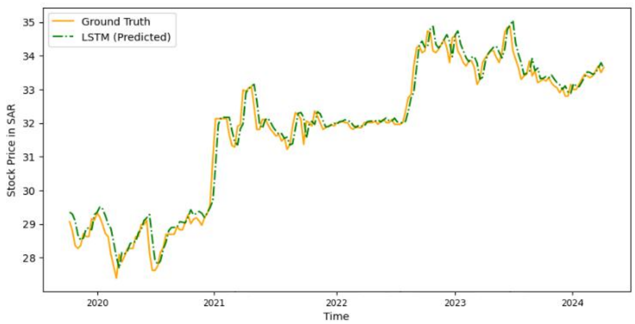


Fig. 3. LSTM prediction performance of Aramco stock market.

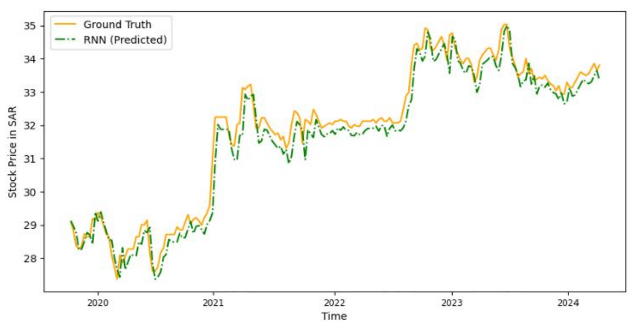


Fig. 4. RNN prediction performance of Aramco stock market.

### A. Discussion

The study evaluated three specific models: Linear Regression (LR) from Machine learning, Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks from Deep learning.

*1) Linear Regression (LR):* The study found that LR performed better than the deep learning models. This demonstrates that a less complex model can offer better forecasting accuracy, which is a significant finding. However, LR is limited in its ability to solve very complicated problems.

*2) Long Short-Term Memory (LSTM) Networks:* LSTMs are an advanced form of RNNs that address the vanishing gradient problem by using memory cells. The study found that LSTMs had a higher RMSE error compared to LR and performed better than RNNs. Its issues in this experiment are related to its complexity.

*3) Recurrent Neural Networks (RNNs):* The study's evaluation showed RNNs with the highest RMSE. A key weakness of standard RNNs is their difficulty in handling longer sequences due to issues with exploding modeling time series data, especially with non-linear and non-stationary trends.

Ultimately, this indicates that when assumed as a machine learning model and thus trained and treated as such, Linear Regression will outperform more complex Deep learning models - like RNNs and LSTMs - on stock market prediction tasks particularly when sequential patterns are not overly prevalent in the data or when the dataset is smaller in size. In conclusion, the findings continue to emphasize how model selection should be based on understanding the data. DNNs can include hidden layers meant to learn complex patterns in data, but models (assumptions of the data/structure) can still reveal the best predictive accuracy. While there is an excellent opportunity for deep learning models, their performance is not always superior, particularly in cases where the simpler model reveals the best outcomes.

## VI. Critical Analysis and Synthesis of Key Findings, Gaps, and Trends

A central finding of the research is that a simpler model like Linear Regression outperformed more complex deep learning models (RNN and LSTM) in terms of lower error rates. This contradicts the general trend in machine learning, where deep learning models are often lauded for their superior ability to handle complex, non-linear data. This finding suggests that while DL models are powerful, their effectiveness in stock market prediction is not guaranteed and is highly context-sensitive and data-driven. It could be that the specific dataset or preprocessing methods used in the study were more suited to a linear model, or that the deep models were not sufficiently optimized.

### A. The Dominance of Deep Learning for Non-linear Data

Despite the specific findings of the study, the broader literature review consistently highlights the shift from traditional statistical methods to more sophisticated AI technologies (ML and DL) to analyze the market's complex and non-linear data. Deep learning models, particularly LSTMs, are specifically praised for their capacity to handle time series data and mitigate the vanishing gradient problem, making them suitable for the long-term dependencies inherent in financial data. The trend is also moving toward hybrid models, like CNN-LSTM, which aim to combine the strengths of different architectures to improve prediction accuracy by capturing both spatial and temporal features.

### B. Data-Centric and Preprocessing Trends

The study explicitly mentions the use of "modern data preprocessing procedures," including Z-score normalization and temporal sequence, to ensure effective model training. The literature also notes the increasing use of large, heterogeneous datasets that go beyond historical prices to include technical indicators, financial news, and social media sentiment. This trend towards incorporating diverse data sources is a key advancement, as it enriches the models with more context and helps them adapt to changing market conditions.

### C. Gaps and Underexplored Areas

The study identifies several limitations and areas for future research:

*1) Model interpretability:* While models like Linear Regression offer interpretability, deep learning models often have complex functions. The paper notes that model interpretability is a key issue, especially for LSTM-based financial applications.

*2) Generalizability and robustness:* The study highlights that a model's ability to generalize to "distribution shifts" over time is a significant issue. The non-stationary nature of stock data means models must be periodically retrained on recent data.

*3) The "Human Element":* This study mentions incorporating predictions and insights into trading systems for "prescriptive analytics" and "integration with human decision-making". This suggests a gap in current research, which often focuses solely on model accuracy, without fully exploring the integration of these AI systems with human expertise and trading strategies.

*4) Real-world applicability:* The paper mentions that further tweaking and testing on "live data" is needed to achieve real-life applicability for models like CNN-LSTM. This points to a gap between theoretical model performance and practical, real-world deployment.

The findings of this research extend beyond the performance of individual models, offering insights into core, ongoing challenges in computer science and its applications.

*1) The most striking finding,* where the simple Linear Regression (LR) model outperformed the more complex deep learning models (RNNs and LSTMs), the study provides experimental evidence that for financial forecasting, which is both context-sensitive and data-driven, a complex model is not always the optimal choice. This highlights the ongoing challenge for computer scientists to select, not just develop, the right algorithm for a specific problem and dataset rather than universally favoring the latest or most sophisticated architecture.

*2) Adaptive systems for dynamic environments:* The document repeatedly mentions the challenge of nonstationarity in stock data and the need to periodically retrain models on recent data. This is a central problem in computer science applications that deal with dynamic, evolving systems, from robotics to network security. The need for models to adapt to distribution shifts over time without manual intervention is a major research area. The findings from this study confirm that for financial data, which is highly sensitive to real-world events and changes, the ability of a model to generalize and remain robust is a critical, and often unfulfilled, requirement.

*3) Hybrid architectures for complex problems*: The research's focus on hybrid models like the CNN-LSTM directly reflects a major trend in modern computer science: the move toward multi-modal or hybrid architectures. By combining different types of neural networks, researchers aim to leverage the unique strengths of each component to solve more complex problems than a single model could alone. The CNN-LSTM architecture, for example, is designed to extract both spatial features and temporal relationships. This approach represents a shift from finding a single "best" algorithm to building composite systems that are better suited for the multifaceted nature of real-world data.

## VII. Conclusion

From a practical perspective, the research has important guidance to provide traders and financial analysts on linear and non-linear models. A linear model has features such as interpretability, rapid calculation speed, and reliable performance that make it a practical choice for real-time trading systems. The work did reveal that some models used in the experiments produced reliable predictions in empirical tests and supported context-based model selection.

This research investigated the capacity of machine learning and deep learning models to predict prices for the consumed stock of Saudi Aramco. The results demonstrated that the simpler models such as the Linear Regression model were able to achieve lower RMSE and MAE errors than LSTM and RNN models. This suggests that simpler models can capture stock movements, which seem to be relatively linear. The lower performance of the deep-learning models suggests that they likely overfit the data, leading to the notion that not all complexities lead to better forecasting models. This study highlighted the relevance of regression models in financial forecasting. This is also highly applicable to emerging markets, where data characteristics are not conducive to complex neural architectures.

Future work should apply to a wider diversity of stocks, incorporate outside economic variables, and investigate ensemble models containing both linear and deep learning components. Future research also needs to focus on making complex models more transparent so that implementers can understand why a particular prediction was made. Further research is needed to create models that are more robust and adaptable to sudden and unexpected market changes. This is crucial for building trust and enabling human decision-making. All these then provide the basis for smarter and more efficient specific decision-making in algorithmic trading, risk management, and portfolio optimization

## References

[1] Y. C. Chen and W. C. Huang, "Constructing a stock-price forecast CNN model with gold and crude oil indicators," *Appl. Soft Comput.*, vol. 112, p. 107760, Jan. 2021.

[2] Li, X., Wu, P., & Wang, W., "Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong," *Inf. Process. Manag.*, vol. 57, no. 5, 102212, Sep. 2020.

[3] A. M. Pattanayak, A. Swetapadma, and B. Sahoo, "Exploring different dynamics of recurrent neural network methods for stock market prediction—A comparative study," *Appl. Artif. Intell.*, vol. 38, no. 1, 2371706, Jan. 2024.

[4] F. M. P. Fozap, "Hybrid machine learning models for long-term stock market forecasting: Integrating technical indicators," *J. Risk Financial Manag.*, vol. 18, no. 4, p. 201, Apr. 2025.

[5] J. Ayala, M. García-Torres, J. L. V. Noguera, F. Gómez-Vela, and F. Divina, "Technical analysis strategy optimization using a machine

learning approach in stock market indices," *Knowl.-Based Syst.*, vol. 225, 107119, Jul. 2021.

[6] D. R. Khan, A. B. Patankar, and A. Khan, "An experimental comparison of classic statistical techniques on univariate time series forecasting," *Procedia Comput. Sci.*, vol. 235, pp. 2730–2740, Apr. 2024.

[7] E. Gambhir, R. Jain, A. Gupta, and U. Tomer, "Regression analysis of COVID-19 using machine learning algorithms," in *Proc. Int. Conf. Smart Electron. Commun. (ICOSEC)*, 2020, pp. 65–71.

[8] P. Yu and X. Yan, "Stock price prediction based on deep neural networks," *Neural Comput. Appl.*, vol. 32, no. 6, pp. 1609–1628, Jun. 2020.

[9] C. Thakar and S. Ratnaparkhi, "Machine learning – Predict stock prices using regression," *Quantitative Finance & Algo Trading Blog by QuantInsti*, Apr. 18, 2023. [Online]. Available: https://blog.quantinsti.com/machine-learning-trading-predict-stock-prices-regression/

[10] S. Nusinovici, Y. C. Tham, M. Y. C. Yan, D. S. W. Ting, J. Li, C. Sabanayagam, T. Y. Wong, and C. Y. Cheng, "Logistic regression was as good as machine learning for predicting major chronic diseases," *J. Clin. Epidemiol.*, vol. 122, pp. 56–69, Jun. 2020.

[11] H. Jain, A. Khunteta, and S. Srivastava, "Churn prediction in telecommunication using logistic regression and logit boost," *Procedia Comput. Sci.*, vol. 167, pp. 101–112, Mar. 2020.

[12] I. Paliari, A. Karanikola, and S. Kotsiantis, "A comparison of the optimized LSTM, XGBOOST and ARIMA in time series forecasting," in *Proc. Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, 2021, pp. 1–7.

[13] A. O. Aseeri, "Effective RNN-based forecasting methodology design for improving short-term power load forecasts: Application to large-scale power-grid time series," *J. Comput. Sci.*, vol. 68, p. 101984, Mar. 2023.

[14] D. K. Mademlis and N. Dritsakis, "Volatility forecasting using hybrid GARCH neural network models: The case of the Italian stock market," *Int. J. Econ. Financial Issues*, vol. 11, no. 1, pp. 49–60, 2021.

[15] L. Rubio, A. P. Pinedo, A. M. Castaño, and F. Ramos, "Forecasting volatility by using wavelet transform, ARIMA and GARCH models," *Eurasian Econ. Rev.*, vol. 13, no. 3–4, pp. 803–830, 2023.

[16] B. Lin and T. Lan, "The time-varying volatility spillover effects between China's coal and metal market," *J. Futures Mark.*, vol. 44, no. 5, pp. 699–719, May 2024.

[17] R. Gomila, "Logistic or linear? Estimating causal effects of experimental treatments on binary outcomes using regression analysis," *J. Exp. Psychol. Gen.*, vol. 150, no. 4, pp. 700–709, Apr. 2021.

[18] W. Jiang, "Applications of deep learning in stock market prediction: Recent progress," *Expert Syst. Appl.*, vol. 184, 115537, Dec. 2021.

[19] K. Kumar and M. T. U. Haider, "Enhanced prediction of intra-day stock market using metaheuristic optimization on RNN–LSTM network," *New Gener. Comput.*, vol. 39, no. 1, pp. 231–272, Jan. 2021.

[20] W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, "A CNN-LSTM-based model to forecast stock prices," *Complexity*, vol. 2020, pp. 1–10, Aug. 2020.

[21] K. Gajamannage, Y. Park, and D. I. Jayathilake, "Real-time forecasting of time series in financial markets using sequentially trained dual-LSTMs," *Expert Syst. Appl.*, vol. 223, 119879, Jan. 2023.

[22] A. Peivandizadeh, S. Hatami, A. Nakhjavani, L. Khoshsima, M. R. C. Qazani, M. Haleem, and R. Alizadehsani, "Stock market prediction with transductive long short-term memory and social media sentiment analysis," *IEEE Access*, vol. 12, pp. 87110–87130, 2024.

[23] H. Widiputra, A. Mailangkay, and E. Gautama, "Multivariate CNN-LSTM model for multiple parallel financial time-series prediction," *Complexity*, vol. 2021, no. 1, 9903518, 2021.

[24] Ş. Öztürk and U. Özkaya, "Residual LSTM layered CNN for classification of gastrointestinal tract diseases," *J. Biomed. Inform.*, vol. 113, 103638, Jan. 2021.