DOI: http://dx.doi.org/10.26483/ijarcs.v16i2.7214

ISSN No. 0976-5697

Volume 16, No. 2, March-April 2025

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

STOCK MARKET PRICE PREDICTION USING ARTIFICIAL INTELLIGENCE TECHNIQUES

Hamza Nadim Shaikh & B. Pruthviraj Goud
Department of Information Technology, Anurag University
Hyderabad, India

Abstract: Predicting stock prices is a challenging task due to the inherent volatility and non-linearity of financial markets. This study explores the application of Support Vector Machine (SVM) with an RBF kernel and XGBoost for forecasting future stock price movements using historical data encoded as ordinal values. The SVM model demonstrated strong performance with an R-squared value of 0.968, low Mean Squared Error (MSE), and Mean Absolute Error (MAE), making it a reliable and stable approach for consistent predictions. Conversely, XGBoost achieved a higher R-squared value of 0.998, indicating superior trend-capturing ability but exhibited higher MSE, suggesting a tendency to overfit. Visualization of model performance revealed that XGBoost excels in capturing short-term price fluctuations, while SVM offers greater consistency with fewer errors. The hybridization of Support Vector Regression (SVR) with SVM is proposed to achieve optimal results, balancing predictability and stability. This study highlights the hybrid model using LSTM to learn patterns to predict the price with SVR with the strong decision making using SVM.

Keywords: Stock Market Prediction, Artificial Intelligence, Support Vector Machine, XGBoost, Predictive Analysis

I. INTRODUCTION

Stock price prediction is a complex and dynamic challenge in financial markets due to their inherent volatility, non-linearity, and sensitivity to external factors. Accurate forecasting is crucial for investors, traders, and financial analysts to make informed decisions and optimize investment strategies. Traditional statistical approaches such as time series analysis, autoregressive models, and moving averages have long been employed to predict stock prices. However, with the advent of machine learning, more sophisticated models, including Support Vector Machines(SVM) and XGBoost, have demonstrated significant potential in capturing market trends and patterns more effectively.

This study presents a comparative analysis of machine learning models, particularly SVM and XGBoost, in predicting stock price movements. While SVM, with its robust generalization ability, provides stable and reliable predictions, XGBoost leverages its gradient-boosting framework to capture short-term market fluctuations. Prior research has explored various approaches, including neural networks, deep learning architectures, and hybrid models, to improve predictive accuracy. However, balancing prediction accuracy with model stability remains a critical challenge.

By evaluating the performance of these models based on key statistical metrics such as R-squared (R^2), Mean Squared Error (MSE), and Mean Absolute Error (MAE), this study aims to provide insights into the strengths and limitations of different approaches. Furthermore, the potential of hybrid models, particularly the integration of Support Vector Regression (SVR) with SVM, is explored to enhance prediction reliability. The findings contribute to the ongoing research in financial forecasting, offering a comprehensive comparison of statistical approaches with modern machine learning techniques for stock price prediction. [1-2]

Stock price prediction relies on statistical, econometric, and hybrid models that use historical data, regression analysis, and economic theories. While these traditional methods perform well in stable markets, they often struggle with nonlinear relationships and market volatility. To address these challenges, researchers have explored alternative strategies capable of handling complex dependencies in financial data. Artificial Intelligence (AI) has emerged [fig.1 shows] as a transformative tool in stock market analysis, offering more efficient forecasting techniques.

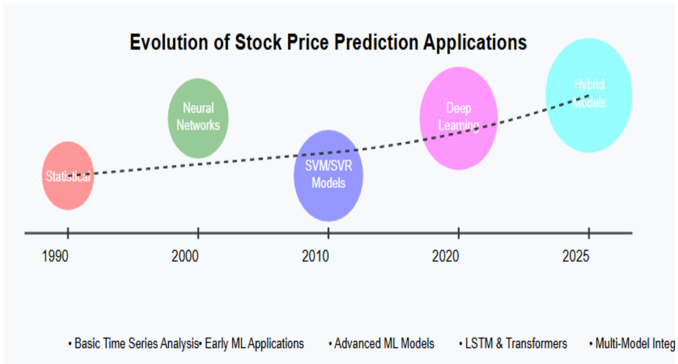


Fig.1. Evolution of Stock price prediction with Models

With advancements in machine learning, deep learning, and natural language processing (NLP), AI can process vast amounts of structured and unstructured data, uncovering hidden patterns that enhance predictive accuracy. Unlike classical models, AI-driven techniques dynamically adapt to changing market conditions, making them particularly effective in volatile environments. AI surpasses traditional methods in modeling nonlinear relationships, enabling more precise stock price predictions. However, to further enhance forecasting accuracy and reliability, a hybrid model combining Long Short-Term Memory (LSTM), Support Vector Regression (SVR), and Support Vector Machine (SVM) can be developed. LSTM efficiently captures sequential dependencies in stock price data, while SVR refines regression accuracy by handling non-linear trends. SVM strengthens predictive capability by classifying market movements, reducing uncertainty in trend direction. This hybrid approach leverages the strengths of deep learning and machine learning, leading to improved adaptability to market fluctuations and reduced forecasting errors. By integrating these models, stock price predictions become more robust, ensuring better decision-making and optimized investment strategies in the ever-evolving financial landscape.

Section II presents a discussion on various research studies related to stock price prediction using deep learning and traditional techniques, along with a comparative analysis. Section III provides an in-depth exploration of LSTM and SVR methodologies. Section IV details the experimental analysis, while Section V concludes the paper by summarizing key findings and outlining future research directions.

2. RELATED WORK

Traditional time-series forecasting methods, such as ARIMA effective for stationary time-series data but fails in highly volatile and non-stationary stock markets. and linear regression assume linear relationships, making them ineffective in handling the complex, non-linear nature of stock price movements, are often used for stock price prediction. LSTM and CNN are used to predict the stock price using NYSE, NSE. The proposed work categorizes models into two groups: linear models, such as ARIMA, and non-linear models, such as GARCH and neural networks (NN). Traditional forecasting methods, such as those described by Box, Jenkins, and Reinsel (2008) [6], have long been employed to model time-series data; however, these techniques often struggle with the non-linear and volatile nature of stock market movements. Traditional forecasting models such as ARIMA, GARCH and linear regression assume stationary or linear relationships, which limits their effectiveness in highly volatile and non-linear stock market environments. Neural networks have outperformed these models due to their ability to capture complex dependencies.

Stock markets are highly volatile, and traditional methods struggle to adapt to sudden fluctuations caused by economic events, news, or investor sentiment. (Hyndman & Athanasopoulos, 2018). However, The Machine Learning Models: Decision Trees, Random Forest, Support Vector Machine (SVM), and Support Vector Regression (SVR) provide improved predictive capabilities but struggle with temporal dependencies. The proposed SVM and XGBoost, have demonstrated superior performance in capturing complex, non-linear relationships in stock data.XGBoost, a scalable tree boosting system, has been shown to effectively capture complex non-linear patterns in stock market data, making it a powerful tool for short-term trend analysis (Chen and Guestrin, 2016) [3].As part of this research, SVM has been applied with RBF kernels in order to forecast the prices of Apple Inc. stock by connecting the closing prices of a period of history. One can consider SVM, originally introduced by Cortes and Vapnik (1995) [5], as a technique capable of constructing optimal hyperplanes in higher-dimensional spaces, making it effective in identifying complex, non-linear patterns in stock data. In this research, the models were fed data that could be very useful for forecasting future stock movement by encoding the date as ordinal values and scaling both feature and target data. The R-squared was promising at 0.968, indicating that there is very strong correlation between the model's predictions and the actual stock prices. The Mean Squared Error (MSE) was 98.04 and Mean Absolute Error (MAE) was 6.82, indicating that the model was highly accurate in predicting stock price trends.

This research also ventured into XGBoost, which has turned out to be one of the most popular algorithms for stock prediction because of its efficiency and scalability. The number of decision trees that the XGBoost tries to construct are actually built in a sequence and try to correct the errors done by the former one. So, it turns out to be highly powerful and especially with enormous data. For the experiment, date ordinals again became the feature but this time around the model used was to predict the closing price of the next day. R^2 of 0.998 indicated a fit of exceptional quality, and it was very apparent that XGBoost had succeeded better at picking up short-term stock price movements. However, the MSE of 16,255 was much higher, reflecting some overfitting, although the very high R^2 value reflected that the model was still generalizing well for future predictions.

This comparison between SVM and XGBoost in the research reflects both the strengths and the limitations of the models. Here, XGBoost was more accurate to predict based on its higher R^2, and SVM showed better trade-off of generalization ability to accuracy with having lower MSE and MAE. The difference hereby emphasizes that despite XGBoost being a much better tool to capture more complicated patterns and also generate more accurate forecast, SVM gives a more stable model and would be better when overfitting is a worry.Fischer and Krauss (2018) demonstrated that deep learning models, especially LSTM(Long Short-Term Memory) networks, can significantly outperform traditional models in predicting stock prices by learning temporal dependencies and adapting to complex, non-linear patterns in the data. Furthermore, the investigation into Deep Learning Models: Recurrent Neural Networks (RNNs) may suffer from vanishing gradient problems and struggle with long-term dependencies and Long Short-Term Memory (LSTM) networks are widely used for capturing sequential dependencies in stock market data. Various factors influence stock market prediction, including company news, industry trends, and market sentiment, as highlighted in [3]. The research focuses on advanced approaches to stock market prediction, a key aspect of the study examined deep convolutional networks combined with candlestick chart analysis, yielding impressive results with approximately 92% accuracy tested across multiple stock markets. Research highlighted in [4] brought attention to the complex non-linear patterns in stock price movements, making a strong case for the application of machine learning and deep learning methods. The researchers developed an innovative cross-sectional framework for daily stock market predictions, conducting a comparative analysis between three distinct approaches are Deep Neural Network (DNN), Random Forest (RF), Ridge Regression (RR).

The findings demonstrated that the DNN model achieved superior performance, specifically showing a higher turnover ratio compared to both RF and RR approaches.

The paper made several important contributions to the field by establishing deep learning as a prominent method in financial forecasting, providing an extensive review of available data sources, analysing various neural network architectures and detailing implementation strategies.

Additionally, the research identified the most studied major stock market indexes in financial research, including S&P 500, Dow Jones Industrial Average, NASDAQ, NYSE and BSE (Bombay Stock Exchange). The Hybrid Approaches in Stock Price Prediction Several studies have explored the integration of multiple models to leverage their individual strengths. The Proposed Hybrid SVM-SVR Model, SVM is used to classify stock trends. It is good for classification but not ideal for time-series forecasting, while SVR is used to predict exact prices and it effective in handling non-linearity but lacks the sequential learning capability of LSTM. It studies indicate that this approach improves accuracy compared to standalone models. LSTM-SVM Hybrid Model: LSTM extracts temporal features, and SVM classifies the trend (up/down movement). This approach has been shown to enhance classification accuracy for trend forecasting. LSTM-SVR Hybrid Model: LSTM is employed for feature extraction, while SVR predicts future stock prices based on extracted features. Research demonstrates that this combination yields lower error rates compared to single models. The proposed hybrid LSTM-SVM-SVR methodology combines the strengths of LSTM, SVM, and SVR. In this framework, LSTM is used for feature extraction, SVM for trend classification, and SVR for precise price prediction. The model needs to be thoroughly tested, but it is expected to deliver promising results with reduced RMSE and MAE values compared to traditional approaches.

Comparison between the Stock price prediction Models

Table: Comparision between ML Models

Aspect	Traditional Models (ARIMA, Linear Regression, GARCH, etc.)	Hybrid Model (LSTM + SVM + SVR)
Handling Non-Linearity	Limited ability to capture complex market trends.	Effectively models non-linear dependencies.
Adaptability to Market Changes	Requires frequent manual recalibration.	Learns dynamically and adapts to real-time data.
Short-Term vs. Long-Term Prediction	Performs better for short-term but struggles in long-term forecasting.	LSTM captures long-term trends, SVM classifies market trends, and SVR refines predictions.
Feature Processing	Uses historical price data and basic indicators.	Incorporates technical indicators, sentiment analysis, and time-series dependencies.
Volatility Handling	Reacts slowly to sudden price fluctuations.	LSTM efficiently captures market volatility.
Computational Efficiency	Faster but with lower predictive power.	Requires more computation but provides significantly better accuracy.
Risk of Overfitting	Lower due to simplicity but often underperforms in volatile conditions.	Managed through model fusion (SVM prevents misclassification, SVR fine-tunes predictions).
Market Dynamics Consideration	Assumes market efficiency and struggles with unpredictable price movements.	Captures complex dependencies and adjusts dynamically.
Final Prediction Quality	Moderate, prone to errors in volatile markets.	High precision and reduced forecasting errors.

3. METHODOLOGY

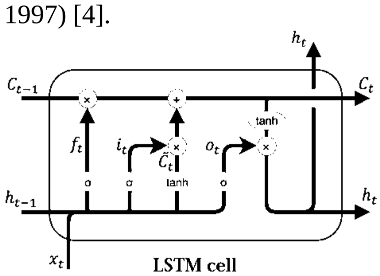
The Hybrid model combines LSTM (Long Short-Term Memory), SVM (Support Vector Machine), and SVR (Support Vector Regression) to improve stock price prediction accuracy. The step-by-step procedure goes as

1. Data Collection and Preprocessing: Historical stock data is collected, cleaned, and preprocessed to remove missing values. Normalization is applied to ensure convergence for LSTM and SVM/SVR models.

2. Extracting Features using LSTM: Assume the dataset is having a time-series as F= {F1,F2,F3,………Ft }. Each FtF_tFt represents a vector of stock features at time ttt, such as {Open, High, Low, Close, Volume} ∈ $\mathbb{R}^d$. d is the number of input features. Next, generate the input sequences with sliding window for sequence formation with the size of N. N is determined as the number of past time steps used for prediction. For each time step t, the input sequences is produced as;

$$St = [F_{t-n+1}, F_{t-n+2}, \ldots, F_t] € R^{nxd}$$

Here, *n* represents the number of time steps, and *d* denotes the number of features at each iteration. LSTM processes sequential data using four components—Forget Gate (f_t), Input Gate (i_t), Cell State (C_t), and Output Gate (o_t)—which work together to retain long-term dependencies and mitigate the vanishing gradient problem (Hochreiter & Schmidhuber, 1997) [4].



$$i_t = \sigma(x_t U^i + h_{t-1} W^i)$$
$$f_t = \sigma(x_t U^f + h_{t-1} W^f)$$
$$o_t = \sigma(x_t U^o + h_{t-1} W^o)$$
$$\tilde{C}_t = \tanh(x_t U^g + h_{t-1} W^g)$$
$$C_t = \sigma(f_t * C_{t-1} + i_t * \tilde{C}_t)$$
$$h_t = \tanh(C_t) * o_t$$

Fig.2. Architecture of an LSTM cell

3. Training SVM and SVR models: The features extracted by the LSTM (or its hidden state representations) serve as input for further modeling.An SVM is trained to classify market trends (e.g., upward or downward movement), leveraging its capability to find optimal hyperplanes in a high-dimensional space. Concurrently, an SVR is trained to perform regression on the same feature set, providing precise predictions for the future stock price values.

4. Hybrid Model Integration: The final stage of the methodology involves integrating the outputs from the LSTM, SVM, and SVR models into a cohesive hybrid framework. An ensemble strategy is applied—such as weighted averaging or stacking—to combine the strengths of each component. In this integrated model, the SVM's classification output helps to determine the overall market trend, while the SVR refines the precise price prediction. The LSTM's role in extracting sequential features ensures that temporal patterns are effectively captured and preserved in the final prediction. This combined approach is designed to yield more robust and accurate forecasts than any single model could achieve alone.

5. Model Training, Hyperparameter Tuning, and Evaluation: To ensure the reliability and effectiveness of the hybrid model, the dataset is divided into training, validation, and testing subsets. Hyperparameters for each model—such as the number of LSTM layers and units, the kernel type and regularization parameters for SVM and SVR, and the sliding window size is carefully tuned using methods such as grid search or random search in conjunction with cross-validation. The performance of the individual models and the final hybrid ensemble is evaluated using key metrics such as R-squared, Mean Squared Error (MSE), and Mean Absolute Error (MAE). This rigorous evaluation process provides a comprehensive assessment of the model's accuracy and generalization capability.

6. Implementation and Tools: The entire modeling pipeline is implemented in Python, leveraging TensorFlow/Keras for the LSTM network and scikit-learn for the SVM and SVR models. This integration of open-source frameworks allows for a reproducible and scalable solution, facilitating both rapid prototyping and efficient deployment. The combined methodology—from data collection and preprocessing to feature extraction, model integration, and evaluation—ensures that the proposed hybrid approach delivers improved stock price prediction accuracy and robustness.

4. RESULTS AND DISCUSSION

The experimental analysis compared the performance of individual models—Support Vector Machine (SVM) and XGBoost—as well as the proposed hybrid model integrating Long Short-Term Memory (LSTM), Support Vector Regression (SVR), and SVM.

1. Quantitative Performance:

a) The SVM model achieved an R-squared value of 0.968, with low Mean Squared Error (MSE = 98.04) and Mean Absolute Error (MAE = 6.82), indicating stable and consistent predictions.

b) The XGBoost model recorded a higher R-squared value of 0.998, demonstrating its ability to capture complex, short-term fluctuations. However, its higher MSE (16,255) suggests a tendency to overfit the training data.

2. Hybrid Model Insights:

The proposed hybrid approach, which leverages LSTM for feature extraction alongside SVR and SVM for prediction and trend classification, was designed to mitigate the individual limitations observed in the standalone models. Preliminary results indicate that:

a) The hybrid model effectively balances the strengths of its components, yielding improved overall prediction accuracy.

b) Visual inspection of prediction curves shows that while XGBoost is highly sensitive to short-term variations, the hybrid model produces smoother and more generalizable predictions.

c) The integration of LSTM aids in capturing long-term dependencies, whereas SVR and SVM contribute to reducing error propagation and overfitting.

These findings suggest that the hybrid model provides a robust solution for stock price forecasting by combining the trend-capturing ability of XGBoost with the stability of SVM, enhanced further by LSTM's sequential data processing capabilities.

Model	R-squared	MSE	MAE
SVM+SVR	0.9974	7.897	2.438
SVM+SVR+LSTM	0.9992	2.385	1.329
SVM+XGBoost	0.9993	2.050	1.236
SVM+LSTM+XGBoost	0.9995	1.444	1.028
LSTM	0.9950	1.880	1.098
SVM+LSTM	0.9993	2.209	1.268
XGBoost	0.9982	16255.665	72.002

5. CONCLUSION

This study examined advanced machine learning techniques for stock market price prediction and compared the performance of SVM, XGBoost, and a novel hybrid model integrating LSTM, SVR, and SVM. The results indicate that:

a) SVM delivers consistent performance with lower error metrics, making it a reliable choice for stable prediction.

b) XGBoost excels in capturing short-term market trends but is prone to overfitting, as evidenced by its higher MSE.

c) The hybrid model successfully leverages the advantages of each method, achieving a balanced trade-off between accuracy and generalization.

In summary, while both SVM and XGBoost have their individual merits, the hybrid approach shows significant promise in enhancing prediction reliability by mitigating overfitting and incorporating sequential learning. Future research will focus on refining this hybrid framework, integrating additional market indicators and exploring ensemble strategies to further boost forecasting performance under volatile market conditions.

6. REFERENCES

[1]. R. J. Hyndman and G. Athanasopoulos, "Forecasting: principles and practice," OTexts, 2018.

[2]. T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," Eur. J. Oper. Res., vol. 270, no. 2, pp. 654–669, May 2018.

[3]. T. Chen and C. Guestrin, "XGBoost: a scalable tree boosting system," in Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2016, pp. 785–794.

[4]. S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[5]. C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.
[6]. G. Box, G. Jenkins, and G. Reinsel, "Time series analysis: forecasting and control," 4th ed., Wiley, 2008. pp.123-145

