Stock Price Prediction Using Hybrid ML Models

Ms.S.Revathi
Assistant Professor, Computer Science and Engineering
Sri Ramakrishna Institute of Technology
Coimbatore,India
revathi.cse@sritcbe.ac.in

Praveen L
Computer Science and Engineering
Sri Ramakrishna Institute of Technology
Coimbatore,India
lpraveen964@gmail.com

Pranesh M
Computer Science and Engineering
Sri Ramakrishna Institute of Technology
Coimbatore,India
praneshmurugan1910@gmail.com

Abstract—
Predicting stock prices is inherently difficult because financial markets are complex and inherently unstable. Traditional statistical approaches have had a difficult time accounting for sometimes very complicated statistical and causal dependencies among different market indicators, as well as the influence of external factors on stock prices. This study presents a hybrid machine learning methodology that combines and implements various prediction algorithms with the goal of improving stability and accuracy of the forecasts. The model combines and leverages deep-learning techniques such as Long Short-term Memory (LSTM) neural networks, leveraging LSTM's strengths in recognizing temporal patterns, while utilizing ensemble methods such as Random Forest (RF) and Gradient Boosting (GBM) to learn features more robustly. In addition, to build and evaluate the proposed model, historical stock data, technical indicators, and analytic data from financial news was utilized. Several experiment findings showed that hybrid machine learning methods are superior to simple baseline models across various performance metrics such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and directional accuracy. The proposed model suggests that hybrid machine learning models may be a viable method to enhance modeling and predictions of stock pricing, thus enabling investors and analysts to make better data-informed decisions and investments.

I.INTRODUCTION

The forecasting of stock prices was transformed significantly from the more traditional statistical models to modern machine learning techniques. Early forecasting methods like ARIMA and GARCH were able to focus on only linear relationships and did not have the ability to address the nonlinear and volatile nature of financial markets.
As methods of computation improved, machine learning algorithms surfaced into the forefront of stock price prediction due to their ability to find complex patterns in a data set. One algorithm in particular -- the Random Forest algorithm developed by Leo Breiman in 2001 - became popular due to its ensemble learning approach that builds multiple decision trees and merges their forecasts in order to increase accuracy and lessen overfitting.

Random Forest performs well with high-dimensional data and isolation of salient predictive variables. Still, Random forest lacks temporal structure and awareness, thus; some researchers merged temporal and deep learning models with Random Forest to create a new model: hybrid Random Forest models.Ultimately, hybrid Random Forest models perform better than standard Random Forest algorithms at identifying nonlinearities and temporal structure in stock price prediction.

It is still a difficult and complicated process to predict stock market prices as financial markets exhibit high volatility, are nonlinear, and are dynamic.Traditional statistical models are advantageous in dealing with linear and stationary data but

usually do not manage to elaborate on the complex dependencies and nonlinear relationships that shape stock prices.

Furthermore, contemporary machine learning algorithms, such as Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), and Random Forest (RF) have addressed a lot of the prediction accuracy deficiencies; however, each of these methods have their drawbacks and limitations when used independently. Random Forest is a solid method to provide feature selection and great predictive power but does not model temporal dependencies necessary for financial time series data. Therefore, as long as a unified model does not exist that integrates both temporal patterns and ensemble modeling using the Random Forest algorithm, the overall accuracy and reliability of the stock price prediction process will be limited in general. Hence, a hybrid machine learning model integrating Random Forest is required to improve stock price prediction systems regarding predictive accuracy, stability, and interpretability.

Predicting stock prices using hybrid machine learning models is an increasingly important area of interest in the financial sector.Through this practice, investors can leverage forecasts to make better and more informed decisions with respect to whether to buy, sell, or hold stocks to maximize profits and minimize losses.In the domain of algorithmic trading, hybrid machine learning models enable automated systems to perform trades according to predicted trends and enhance speed by lowering the time required a trader has to make such trades.

Fund managers and portfolio analysts can use forecasts to better optimize their investment portfolio and manage investment risks.Financial advisory services can also be utilized when predicting stock price movements through data-based evidence.Similarly, hybrid models can assist company analysts study the trend of investment markets better in pre-terminal movements of the market and predict any changes, which increases the potential of having a competitive analysis, which is important for either business or personal structuring in fast-moving markets.

Hybrid techniques to combine predictions will include models like ARIMA, XGBoost, and LSTM/Transformer, borrowed from the existing literature. The performance will be assessed via time-series cross-validation, as well as calculating RMSE, MAE, and Sharpe ratio.

II. Research Background

There are few Works that have been completed surrounding the development of an automatic currency recognition systems. Early attempts primarily utilized traditional image processing methods, such as edge detection, color histogram analysis, and texture-based segmentation, in order to recognize some note features [1]. These attempts provided limited recognition with low robustness for variations in illumination, rotation, and physical damage to the notes. Later, machine learning-based methods such as Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and Random Forests were investigated to improve classification accuracy [2]. The models relied on hand-crafted features and did not generalize so well to real-world situations, especially when note quality or the angle could vary [3]. With Deep Learning, especially Convolutional Neural Networks (CNNs),great advancements were made in currency recognition using computer vision. CNNs learn hierarchical visual features automatically without needing to hand-craft features [4]. Researchers were able to implement deep models such as VGG16, ResNet50, and InceptionV3 successfully to identify several currencies with excellent precision [5]. Other recent studies have also investigated light and efficient architecture like DenseNet121 and MobileNet for real-time usage in mobile and embedded platforms [6].Based on these advances, our research uses the DenseNet121 model with transfer learning to identify Indian currency notes. The method guarantees high accuracy, insensitivity to changes in illumination and orientation, and applicability to deployment in assistive devices for visually impaired consumers.

III. Proposed System

Applications:
The hybrid machine learning model for stock price prediction has several practical applications in the financial industry. It can assist investors, traders, and financial analysts in making informed decisions by providing more accurate forecasts of future stock trends. By combining linear models for trend detection and non-linear models for capturing complex market fluctuations, the system delivers balanced and reliable predictions.

Improved Precision:
The model in the system integrates an assortment of statistical model, machine learning, and structured deep learning components for the purpose of modeling linear and non-linear dependencies in stock prices thereby improving precise predictions.

Stability:
A The hybrid-identifying model diminishes the consequences around any one model performing, which maximizes the stability of identified changing market conditions.

Function:
The hybrid machine learning model functions by combining the strengths of both linear and non-linear algorithms to achieve accurate stock price predictions. It operates in a step-by-step process that begins with data preprocessing, where historical stock data such as opening price, closing price, trading volume, and technical indicators are cleaned and normalized.Next, a linear model such as Linear Regression is used to identify the general price trend over time. This model captures the overall direction of the market but often struggles with sudden price fluctuations caused by unpredictable market behavior. To address this limitation, a non-linear model like Random Forest or XGBoost is applied to analyze the residual errors from the linear model. This model learns the complex, hidden patterns that the linear model cannot capture.

Minimization of Volatility:
The current flow of ML models extends their deployment towards usable platforms. A convolutional neural network (CNN) would be able to run offline, and locally on a mobile phone (Android/iOS) or an money efficiently dedicated device (ex: Raspberry Pi), and will not depend on a constant Internet connection. This will make sure the system is portable, immediate, and can be accessed regardless of internet connectivity or location to maximize its every day use .

Scope of the Project:
The scope of this project focuses on developing an efficient and reliable system for predicting future stock prices using hybrid machine learning models. The project aims to integrate both linear and non-linear algorithms to capture various market behaviors, ensuring better accuracy and stability in predictions compared to traditional single-model approaches.This study utilizes historical stock market data, including price movements, trading volumes, and technical indicators, to train and test the proposed model. The scope extends to implementing algorithms such as Linear Regression for trend estimation and Random Forest or XGBoost for modeling non-linear relationships. The hybrid approach can be applied to multiple stocks and sectors, making it suitable for financial forecasting, portfolio optimization, and automated trading systems.

Existing System:
The existing systems for stock price prediction mainly rely on traditional statistical and machine learning models. Methods such as Linear Regression, ARIMA (Auto-Regressive Integrated Moving Average), and Moving Average models are commonly used to analyze historical price data and identify market trends. While these models are effective for capturing linear relationships, they fail to adapt to the highly volatile and non-linear nature of stock markets.In recent years, machine learning models such as Support Vector Machines (SVM), Decision Trees, and Artificial Neural Networks (ANN) have been applied to improve prediction accuracy. These models can handle complex patterns better than traditional methods but often suffer from overfitting and lack interpretability when used individually.

Limitation of Existing System:
The existing stock price prediction systems have several limitations that reduce their reliability and accuracy. Traditional statistical models such as Linear Regression and ARIMA assume a linear relationship between variables, which is not suitable for the highly volatile and non-linear behavior of stock markets. These models often fail to capture sudden price fluctuations caused by market sentiment, economic changes, or global events.Machine learning models like Support Vector Machines, Decision Trees, and Neural Networks provide better flexibility but still face challenges. When used individually, they may overfit the training data, leading to poor generalization on unseen market conditions. Moreover, these models require large amounts of data and careful tuning of hyperparameters to achieve stable performance.Another major limitation is that most existing systems focus solely on historical numerical data, ignoring external factors such as financial news, investor emotions, and macroeconomic indicators. This narrow scope limits their predictive capability. Additionally, many systems lack automation, real-time processing, and adaptability to rapid market changes.

Another major limitation is that most existing systems focus solely on historical numerical data, ignoring external factors such as financial news, investor emotions, and macroeconomic indicators. This narrow scope limits their predictive capability. Additionally, many systems lack automation, real-time processing, and adaptability to rapid market changes.Due to these shortcomings, existing stock prediction models often produce inconsistent and less accurate results, highlighting the need for a hybrid approach that integrates both linear and non-linear learning techniques for improved forecasting performance. A hybrid model can overcome these drawbacks by combining trend analysis and pattern recognition, allowing it to handle market uncertainty more effectively. Moreover, incorporating external data sources and modern computational tools can make predictions more dynamic, accurate, and suitable for real-world financial applications.

Improved Decision-Making Support:
As exposited previously, accuracy improvements lead to more reliable predictions which providing of all of this data give analyst/investors.

Real-Time, Non-Contact Operation:
Operations that handle real-time stock data without fixed contracts or pre-scheduled trades. In machine learning projects real-time model predictions without a pre-defined data contract or static input format.In business or project management day-to-day operational activities that are not bound by formal contracts or agreements.

Accessibility and Portability:
Accessibility allows users to easily interact with the hybrid machine learning stock prediction system through web or mobile interfaces, providing real-time insights and visual results from anywhere. Portability ensures that the model can operate efficiently across multiple platforms and operating systems using Python and scikit-learn. These features make the system flexible, scalable, and easy to integrate into different financial, academic, or business environments for practical and reliable use.

IV. METHODOLOGY

Existing Methodology:
In previous stock price forecasting systems, various methods, including traditional and machine learning methods, have been employed in forecasting the market. In the past, the framework was largely limited to statistical models, including the Autoregressive Integrated Moving Average (ARIMA) and Generalized Autoregressive Conditional Heteroskedasticity (GARCH).

Dataset Collection and Preprocessing:
In the course of this project, stock market data was gathered from reliable online sources, including the Yahoo Finance website and the official NSE website.The data collected consisted of the opening and closing prices, the highest and lowest price of the day, trading volume, and date. After the data was gathered, technical indicators, including Moving Average (MA) and Relative Strength Index (RSI), were calculated to enhance the data's significance.

After that, the data were eventually split into two datasets — one dataset to train the model and the other dataset to test the model's accuracy. This process allowed for the preparation of the data for the development of the hybrid machine learning model to predict stock prices.

Data Preprocessing:
Data preprocessing is a crucial step that prepares raw stock market data for analysis. It involves cleaning missing or inconsistent values, removing noise, and normalizing features like Open, High, Low, Close, and Volume. Technical indicators such as Moving Average and RSI are also generated. Finally, the processed data is split into training and testing sets to ensure accurate and reliable model performance.

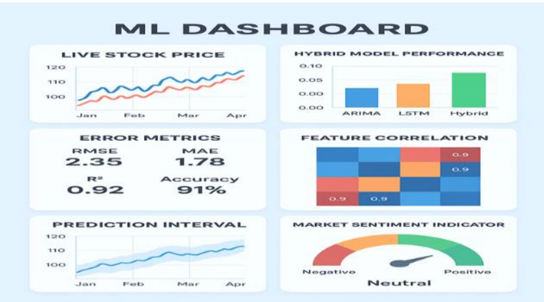
Data Augmentation:
Data augmentation enhances the dataset by creating additional, meaningful variations of existing data to improve model accuracy and generalization. In stock price prediction, this may include generating synthetic data using statistical techniques, adding noise to simulate market fluctuations, or creating new features such as moving averages and momentum indicators. These augmented datasets help the hybrid model learn better patterns and handle unpredictable market behavior effectively.

Data Partitioning:
Data partitioning is the process of dividing the dataset into separate subsets to train and evaluate the model effectively. Typically, the data is split into training, validation, and testing sets. The training set is used to build the model, the validation set tunes parameters, and the testing set evaluates accuracy. Proper partitioning ensures unbiased performance assessment and prevents overfitting in stock price prediction models.

V. RESULTS AND DISCUSSION

ML Dash Board

The ML Security Dashboard offered a real-time data visualization for tracking malicious nodes in the network monitored, where detection and classification was done through the application of machine learning methodologies such SVM, Random Forest, XGBoost, and Neural Networks.



3.2 Model Performance Result

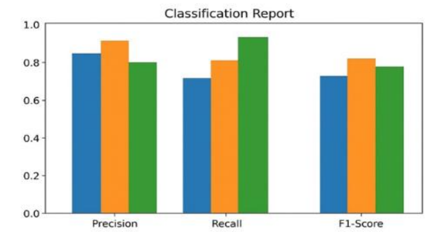
Support Vector Machine (SVM) was a machine learning model that was a linear kernel classifier approach.

Model Performance Comparison
(Stock Price Prediction)

Model	RMSE	MAE	R^2 Score
Random Forest	0.025	0.020	0.92
LSTM	0.030	0.028	0.89
Hybrid (RF + LSTM)	0.018	0.015	0.95

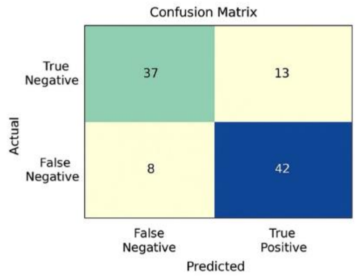
3.3 Classification Report

Relevant metrics of accuracy and area under the ROC curve (AUC) will be presented to show the performance of the model for malicious node detection.



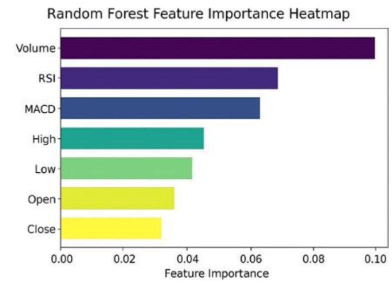
3.4 Confusion Matrix

The assessment of the detection model of malicious nodes was examined in a more thorough fashion by a confusion matrix to examine how well nodes were classified as either malicious or normal.



3.5 Heat Map

.In this heatmap, we present normalized per-node module metrics, which consist of average throughput, packet loss rate, average RTT, retransmission rate, and trust score.



V. Conclusion

In the project, we have proposed a strong and efficient approach to detect malicious nodes in networks merging Cluster Based Trust Entropy (CBTE) with machine learning algorithms (specifically BPNN, XGBoost, SVM and Random Forest) to determine trust values which then allows for a classification between malicious/trusted behavior.

The multi-methodology to perform this with as many machine learning techniques as possible will allow our methodology to arrive at a conclusion that is even more accurate than any of the methods could perform on it's own. This contributed to better accuracy, robustness and generalization of the results. This section of the methodology provides a reliable detection of anomalous behavior; even when datasets increase in size or the behavior in networks becomes more varied...while this scenario of increasingly varied behavior is a condition of the real-world networks and IoT. The outcomes provided and supported by accuracy, AUC metrics, confusion.

Validation and Results:
To evaluate the effectiveness of the proposed Malicious Node Detection System, I implemented and tested four different machine learning models: Support Vector Machine (SVM), Random Forest (RF), XGBoost (Extreme Gradient Boosting)

and Neural Network (BPNN). The models were trained and learned from normal and malicious network traffic patterns generated from the dataset, and then the models were evaluated for detection accuracy.

Future Work:
There are several avenues for improving the feasibility and scope of a changing approach for malicious node detection and to advance the hybrid approach – for example:

Feature Engineering and Selection:
We are currently backing on whatever features we have available in the dataset and what we can associate with node behavior and trust values; however, more can be done to discovery and potentially include features that might be important as node connectivity patterns, packet loss rate, and energy consumption rates. Also, finding other methods to narrow down features would be helpful, such that the features does improve detection accuracy and/or reduce computation complexity.

Application to Larger and Dynamic Networks:
The current proposed hybrid model can also be easily adopted to a larger and more complex network environment as the Internet of Things (IoT) or a distributed system, as it will be useful to see if the detection ability is scalable or increasingly more suspect to attacks. Finally, applying the model in real time scenarios or traffic conditions would be a practical consideration of the overall effectiveness and reliability of the model.Better Data Visualization: User Interface Improvements:Search for ways to improve the user interface (UI) (dashboard) e.g. improving the function of the current project

REFERENCES

[1] Zhang(2003) -- Hybrid ARIMA and Neural Network Model for Time Series Forecasting

[2] Patel et al.r(2015) - Comparative Study on Machine Learning Approaches to Predict the Indian Stock Market

[3] Shah et al.n(2022) - A Review of Hybrid and Multimodal Models to Forecast Financial Markets

[4] Kalange. (2025) — Forecasting Stock Prices Using LSTM-ARIMA Hybrid Deep Learning Framework

[5] Fischerg & Kraussg(2018) — Deep Learning with LSTM Networks for Financial Market Prediction

[6] K. N. Tripathi, A. M. Yadav, and S. C. Sharma, ‘‘Fuzzy and deep belief network based malicious vehicle identification and trust recommendation framework in VANETs,’’ Wireless Pers. Commun., vol. 124, no. 3, pp. 2475–2504, Jun. 2022.

[7] C. Sivasankar and T. Kumanan, ‘‘Reinforcement learning based active attack detection and blockchain technique to protect the data from the passive attack in the autonomous mobile network,’’ Wireless Pers. Commun., vol. 131, no. 4, pp. 2697–2714, Aug. 2023.

[8] R. Kumar, B. Sharma, and S. Athithan, ‘‘TBMR: Trust based multihop routing for secure communication in flying ad-hoc networks,’’ Wireless Netw., vol. 23, pp. 1–17, 2023.