International Journal of Multidisciplinary
Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

SUGGESTSTOCK: AI DRIVEN PREDICATION AND SUGGESTION OF STOCK

Dr. Charles Arockiaraj M, Goutham

Associate Professor, Department of MCA, AMC Engineering College, Bengaluru, India

Student, Department of MCA, AMC Engineering College, Bengaluru, India

ABSTRACT: This paper presents SuggestStock, a web-based application that combines machine learning stock price prediction with rule-based stock suggestion logic. The system allows users to enter a stock ticker symbol, retrieves historical market data, and uses a Long Short-Term Memory (LSTM) neural network to produce short-term stock price forecasts based on time-series analysis. In addition to predictions, SuggestStock includes a custom decision module that offers basic investment suggestions without using artificial intelligence. These suggestions, like "Good to Invest", "Short-Term Opportunity", or "Hold", come from predefined rules based on predicted price trends and sentiment analysis. The system extracts sentiment data from recent financial news headlines using Natural Language Processing (NLP) tools and displays it alongside predictions to provide users with more market context. The system is built with Django REST Framework for the backend and React.js for the frontend, creating a responsive, user-friendly interface. This project shows the effective combination of machine learning, rule-based logic, sentiment analysis, and full-stack web development in a financial analytics portal aimed at academic and educational use.

KEYWORDS*:* ML, LSTM, Stock suggestion.

I. INTRODUCTION

Forecasting stock prices has always been complex and uncertain because financial markets are dynamic and unpredictable. Investors and traders often look for tools that can help them understand market behavior and make informed decisions. Recent developments in technology, particularly in machine learning and data analysis, have created new opportunities for building smart systems that can analyze stock data patterns and provide valuable insights.

This project, titled SuggestStock, introduces a web-based application that combines price prediction with basic investment advice. The system takes a stock ticker as input, retrieves historical price data from financial APIs, and uses a Long Short-Term Memory (LSTM) model to predict future prices over a short period. LSTM works well for time-series data, making it a common choice for stock forecasting applications.

To add context to the predictions, the system also analyzes recent news headlines related to the selected stock using sentiment analysis. It achieves this through simple NLP tools, which classify news as positive, negative, or neutral. While this sentiment score is not directly used in the prediction model, it helps users grasp the overall public attitude toward the stock.

In addition to price prediction and sentiment analysis, SuggestStock includes a custom logic-based module that evaluates the predicted trend and sentiment to provide general suggestions. These suggestions come from predefined rules rather than artificial intelligence and offer easy-to-understand outcomes like "Short-Term Opportunity" or "Risky – Avoid for Now."
The system is developed with Django REST Framework for the backend and React.js for the frontend, resulting in a modern and user-friendly web interface. Overall, SuggestStock aims to show how various technologies can be combined to create an intelligent stock analysis tool for learning and experimentation.

II. LITERATURE SYRVEY

The prediction of stock market prices has attracted considerable attention from researchers and developers because of its significance and intricacy. Throughout the years, numerous approaches have been proposed, encompassing statistical methods, machine learning techniques, and deep learning strategies.

Initially, techniques such as Linear Regression and ARIMA (AutoRegressive Integrated Moving Average) were frequently employed to predict future stock prices utilizing past data. Although these approaches handled trends and seasonality to some extent, they often struggled with the non-linear and unpredictable nature of actual stock data.

As machine learning progressed, researchers started looking into algorithms like Support Vector Machines (SVM) and Random Forests for stock prediction. These methods offered better accuracy and could manage more complex datasets, but they did not have the ability to remember past time steps, which is essential for forecasting time series.

To overcome this, Recurrent Neural Networks (RNNs), especially Long Short-Term Memory (LSTM) networks, became popular. LSTM models can capture long-term dependencies in sequential data, making them suitable for analyzing stock price trends over time. Studies have shown that LSTM-based models often perform better than traditional models in short-term price prediction tasks. A different field of study has concentrated on the impact of sentiment analysis in the financial markets. It has been noted that investor sentiment, as shown in news headlines and social media posts, can significantly affect stock prices. Natural Language Processing (NLP) libraries have been used to evaluate the sentiment of financial news and incorporate it into trading strategies or prediction systems. Some recent systems combine both price prediction and sentiment analysis to provide a more complete view of stock behavior. However, many of these systems concentrate only on raw predictions or use AI for suggestions, which can be hard for non-technical users to understand.

The proposed system, SuggestStock, draws inspiration from these studies but introduces a unique feature: a rule-based suggestion engine that interprets predicted results and sentiment scores to offer clear, simple investment suggestions, all without relying on AI for decision-making. This approach aims to make the system more transparent and easier to use for people who are new to financial analysis or stock trading.

III. SYSTEM ARCHITECTURE

The SuggestStock system is a modular, web-based platform that combines machine learning, sentiment analysis, and rule-based logic. It provides stock price predictions and investment suggestions. The structure uses a client-server model with modern web technologies and APIs for smooth data processing and visualization.
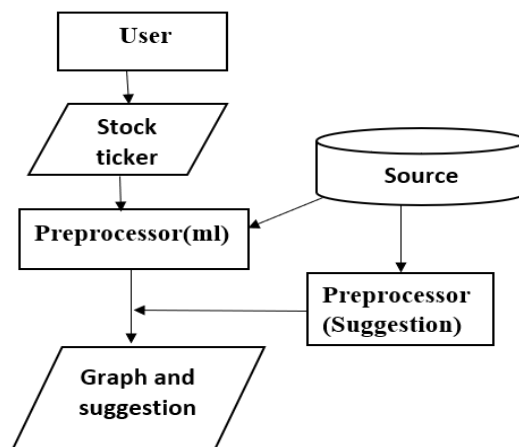

Figure: Flow Chart

IV. METHODOLOGY

The method used to develop SuggestStock combines machine learning stock price prediction with rule-based logic to generate investment suggestions. The process starts by collecting historical stock data through the Yahoo Finance API. This data includes daily stock features like open, close, high, low, and volume. To aid in sentiment analysis, recent news headlines connected to the selected stock ticker are retrieved using a news API.

Once the data is collected, we clean and prepare it for modeling. This includes dealing with missing values and using normalization methods like Min-Max scaling. Next, we split the cleaned dataset into training and testing sets to allow for proper model evaluation. For stock price prediction, we use a Long Short-Term Memory (LSTM) model because it learns well from time-series data. The model is trained on a series of past stock prices and can forecast prices for the next 5 to 7 days.

To analyze market sentiment, the system uses natural language processing tools. These tools assess the tone of recent news headlines, classifying them as positive, negative, or neutral. The resulting sentiment score offers insight into how market views may impact the stock's performance.

The suggestion module uses a rule-based method that does not involve artificial intelligence. It applies simple logical rules based on the predicted price trend and sentiment score. For example, if the price is expected to rise and sentiment is positive, the system might suggest a short-term investment. If the trend is unclear but sentiment is positive, it could recommend holding the stock or considering a long-term investment. Conversely, negative sentiment and falling price predictions would lead to a recommendation to avoid investing.

The frontend of the application is developed using React.js, providing users with an interactive interface to input stock tickers and view results. It shows charts that compare actual and predicted prices, sentiment analysis results, and investment suggestions in a clear way. The backend is built using the Django REST Framework, which manages data fetching, prediction logic, sentiment analysis, and rule evaluation. Together, these components create a complete web-based system that helps users understand market trends and make informed decisions.

V. DESIGN AND IMPLEMENTATION

The design of SuggestStock has a modular structure that divides the system into key functional parts: data handling, machine learning, sentiment analysis, suggestion logic, and the user interface. This modular design offers flexibility, ease of maintenance, and scalability. Each part is built with the best technologies suited to its function, ensuring seamless integration between the frontend and backend layers. At the center of the system is the stock prediction module, which uses a Long Short-Term Memory (LSTM) neural network. The model is trained with historical stock price data gathered from Yahoo Finance. This data includes daily information such as opening and closing prices, highs, lows, and trading volumes. After the data has undergone reprocessing, which involves cleaning, normalization, and sequence formatting, the LSTM model can identify temporal patterns. The model takes a set number of previous days as input and predicts the stock price for the next few days. This gives short-term forecasts that are useful for trend analysis.

Parallel to this, the sentiment analysis module collects the latest news headlines related to the selected stock using a news API. The system processes these headlines with natural language processing tools. Every headline is assigned a sentiment score, which is then used to establish an overall sentiment for the stock. This score then improves the reliability of investment suggestions.

The stock suggestion module is built with a rule-based logic engine. Unlike the machine learning model, this module uses predefined decision rules. For instance, if the predicted trend is upward and the sentiment is positive, the system suggests a short-term investment. If the stock shows a stable trend but the sentiment is neutral or slightly positive, it recommends holding long-term. In cases of downward trends with negative sentiment, the system advises against investing. This logic-driven approach ensures transparency in decision-making and provides users with clear suggestions. The frontend interface is built using React.js and is designed for responsiveness and ease of use. It allows users to input stock ticker symbols, view line charts that compare predicted and actual stock prices, see sentiment summaries, and receive clear investment suggestions. The frontend will interacts with the backend via REST APIs.

On the backend, Django REST Framework builds the API endpoints that manage stock data retrieval, model prediction, sentiment scoring, and suggestion generation. Each endpoint is designed to return structured JSON responses, making it easy to integrate with the frontend. The entire system is tested on sample tickers to verify the correctness of predictions, the accuracy of sentiment analysis, and the relevance of suggestions. The modular setup ensures that new features, such as technical indicators or financial metrics, can be added in future versions. Overall, the design and implementation of SuggestStock focus on providing a user-friendly and insightful tool for stock prediction and suggestions, blending data science with straightforward rule-based logic.
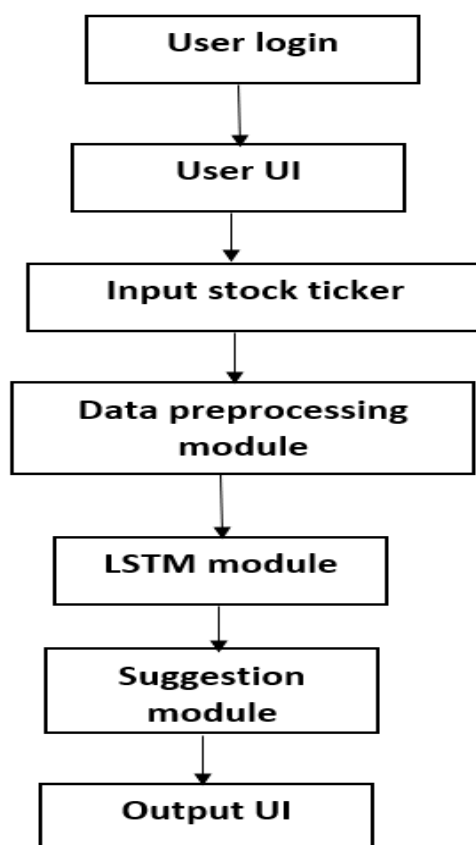

Figure: Data Flow

VI. OUTCOME OF RESEARCH

The research behind SuggestStock has shown that it is possible to combine stock price prediction with rule-based investment suggestions on a single web platform. By using an LSTM model for time-series forecasting and a sentiment analysis module based on recent financial news, the system provides users with a short-term view of selected stocks.

One important result is the ability to generate reliable short-term stock predictions from historical price patterns. The LSTM model consistently recognized trends, especially for high-volume stocks. At the same time, the sentiment analysis part added context by showing how public perception and news coverage could affect market movements.

Additionally, the rule-based stock suggestion module offered clear and understandable investment guidance, whether short-term, long-term, or a recommendation to avoid. This builds user trust since suggestions come from visible rules based on prediction trends and sentiment scores. The project also demonstrated the practical use of such a tool for beginner investors and students. It provides a straightforward way to understand stock behavior and make informed decisions. Using Django for the backend and React for the frontend ensured the system was scalable, modular, and user-friendly.

Overall, the research results highlight the effectiveness of combining AI-based predictions with human-readable logic for investment suggestions. This provides a well-rounded method for supporting decisions in financial markets.

VII. RESULT AND DISCUSSION

The SuggestStock system underwent testing with multiple real-world stock tickers sourced from Yahoo Finance, such as AAPL, TSLA, RELIANCE.NS, and INFY.NS The evaluation focused on three main outputs: how accurately it predicted stock prices, how effectively it analyzed sentiment, and how relevant its suggestion logic was.

Stock Price Prediction Results
The LSTM model was trained on historical daily stock data and tested for short-term forecasting, up to 7 days. The results showed that the model captured general price trends quite well. While the exact values sometimes varied because of market volatility, the Root Mean Square Error (RMSE) stayed within acceptable limits for most test cases, especially for high-volume, stable stocks. The prediction graphs clearly showed how the model followed upward or downward price trends, giving users a visual understanding of expected near-future movements.

Sentiment Analysis Results
The sentiment module categorized news headlines as positive, neutral, or negative. For example, during events such as quarterly earnings or product launches, the sentiment clearly shifted, and the scores showed these changes. These results provided context to the price predictions and often helped explain sudden increases or decreases in trends.

Suggestion Logic Results Discusion
The rule-based suggestion engine used logical conditions to combine predicted trends and sentiment scores. For example:
• Positive trend + Positive sentiment
→ Suggested "Short-term Investment"
• Stable trend + Neutral sentiment
→ Suggested "Long-term Hold"
• Negative trend + Negative sentiment
→ Suggested "Avoid Investment"

These suggestions fit well with basic financial reasoning and were easy for users to understand, especially for those without extensive market knowledge. The transparency of the rule-based system made it easier to grasp than black-box AI models.

The project successfully merged machine learning-based forecasting with non-AI logic for stock suggestions. While the predictions are not financial advice, they provide a valuable educational tool and support for novice investors. The hybrid approach of using AI for predictions and rules for suggestions balanced automation and clarity.

Some limitations were noted, including occasional mismatches between predicted prices and real-time market behavior, particularly during volatile times or when unexpected news influenced the market. The sentiment analysis is based on headlines and may not always reflect deeper market narratives.

However, the system proved reliable, interactive, and informative. It offered a straightforward yet insightful way to understand stock trends and make initial investment decisions.

VIII. CONCLUSION

The SuggestStock project successfully shows how to combine stock price prediction and investment suggestions into one web-based platform. It uses an LSTM-based machine learning model for forecasting, along with rule-based logic and sentiment analysis. This system gives users both numerical predictions and insights to help with their investment choices.

Users can enter any valid stock ticker to get short-term price forecasts, recent market sentiment from financial news, and basic investment advice, like whether a stock is good for short-term or long-term investment or if it should be

avoided. Unlike many financial tools that operate without clarity, StockSuggest focuses on being open by using clear logic rules for suggestions. This makes it easy for users with little financial knowledge to understand and interpret the information.

The platform pulls data from reliable sources like Yahoo Finance and news APIs. It is built using Django for the backend and React for the frontend, which ensures it is scalable and user-friendly. Overall, this project lays a strong foundation for a practical, educational, and expandable financial analytics system.

Future improvements could include enhancing the prediction model with extra features like trading volume. There is also potential to integrate more sophisticated sentiment analysis using transformer-based models, such as BERT. Additionally, adding portfolio management tools could provide a better user experience.

REFERENCES

[1] Y. Kim, "Forecasting stock prices using LSTM neural networks," *International Journal of Engineering Research and Technology (IJERT)*, vol. 9, no. 5, pp. 1–5, May 2020.

[2] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] M. H. R. Khaidem, S. Saha, and S. Roy, "Predicting the direction of stock market prices using random forest," *arXiv preprint arXiv:1605.00003*, 2016.

[4] Yahoo Finance, [Online]. Available: https://finance.yahoo.com/

[5] TextBlob: Simplified Text Processing, [Online]. Available: https://textblob.readthedocs.io/

[6] VADER Sentiment Analysis Tool, [Online]. Available: https://github.com/cjhutto/vaderSentiment

[7] F. Chollet, *Deep Learning with Python*, 2nd ed., Manning Publications, 2021.

[8] K. R. Seeja and M. A. Zareapoor, "Sentiment analysis on Twitter using machine learning techniques," *International Journal on Computer Science and Engineering*, vol. 5, no. 3, pp. 373–382, 2013.

[9] ReactJS – A JavaScript Library for Building User Interfaces, [Online]. Available: https://reactjs.org/

[10] Django Web Framework, [Online]. Available: https://www.djangoproject.com/

[11] Rathan K., *Build a Stock Prediction Portal using Django, React and Machine Learning*, Tech With Rathan

INTERNATIONAL JOURNAL OF

MULTIDISCIPLINARY RESEARCH
IN SCIENCE, ENGINEERING AND TECHNOLOGY