International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Stock-Vista: Market Insight Engine for Forecasting and Data Driven Analytics

Mahesh Badwaik [1], Sumit Kadam[2], Aakash Khuspe[3] , Anjalidevi Patil[4]

[1] *Student of AI&DS Department Datta Meghe College of Engineering Airoli, India*
[2] *Student of AI&DS Department Datta Meghe College of Engineering Airoli, India*
[3] *Student of AI&DS Department Datta Meghe College of Engineering Airoli, India*
[4] *Assistant Professor at AI&DS Department Datta Meghe College of Engineering Airoli, India*

Abstract

The stock market is a place where people trade company shares, playing a vital role in the global economy. Stock exchanges serve by linking buyers and sellers so that these transactions can be made easier. One of the hardest things in finance is accurately forecasting stock prices, particularly for long-term investors that wish to get the best out of their investments. But since the market is highly volatile and in a continuous state of flux, form of Gated Recurrent Unit (GRU) networks, comes in. GRUs are perfect for monitoring time-based data since they are able to discern patterns and trends across time. In this project, we intend to utilize GRU models to forecast stock prices in real time based on historical data from models can be an effective tool for stock price forecasting, helping investors cope with the vagueness of the market and make better decisions.

Key Words: Stocks, Gated Recurrent Unit , Streamlit , Prediction , Real Time Market Dashboard

1.INTRODUCTION

a.Introduction

Analyzing the stock market and forecasting its future has been very significant because it aids individuals in making investment decisions and learning how the market functions. With advancements in machine learning, sophisticated models are increasingly being utilized to forecast stock prices based on historical data. Such a model is called the Gated Recurrent Unit (GRU), which is a type of recurrent neural network (RNN). It is more popular today because it can detect patterns over time in data, which makes it very effective in forecasting stock prices. This process includes several key steps: preparing the data, visualizing it, and making predictions.

Stock market data often include missing values, outliers, and fluctuations over time. To enhance the performance of the forecasting model, we utilize data preprocessing. This involves replacing missing data, scaling features, and converting raw stock data into input data acceptable to GRU. We split the historical stock data into training and test datasets to ensure the model's performance. It is highly essential to know trends and patterns of stock prices prior to making predictions. Data visualization also provides useful information that can be used for model improvement by indicating fluctuations in the market and change.

GRU is suitable to forecast time series since it can remember something that happened very far back. GRU possesses a characteristic that sets it apart from other machine learning algorithms, and that is it can successfully address the vanishing gradient problem. It can therefore learn short- and long-term patterns of stock prices. We apply a GRU model to forecast future stock prices in this work. We train the model using past stock information. Stock visualization is the display, analysis, and comprehension of historical and contemporary stock market data through charts and graphs, and financial data for a market or company. Trends in stock prices can be depicted with various types of charts, i.e., line, bar, and candlestick charts. These visualizations enable investors and analysts to easily view trends and patterns in stock prices. Stock forecasting uses historical data, mathematical models, and statistical analysis to predict future stock prices and market trends. This is of vital importance to financial analysts, investors, and traders in making decisions. They apply various concepts and methods, such as machine learning algorithms.

This article demonstrates the ability of the GRU Model to forecast stock prices and gives a general insight into stock market trends based on data visualization.

b. Problem Statement

Existing models used for stock market trend prediction usually have low precision rates, and this is primarily due to the small training dataset sizes and the low feature numbers they take into consideration. Despite the existence of numerous algorithms that can be utilized for the task, there is an evident lack of publicly available real-world implementations. Therefore, there is a strong need for more efficient, easily accessible, and user-friendly tools to be created that can give accurate forecasts in the stock market.

c.Objective

1. Develop a Predictive Model:
Develop a machine learning model using a Gated Recurrent Unit (GRU) to predict future stock prices from historical price data.

2. Improve Prediction Accuracy:
Improve the accuracy of stock price forecasts by applying technical indicators (e.g., Moving Averages, RSI) and other indicators reflecting the direction of market trends.

3. Time-Series Forecasting:
Use time-series forecasting techniques to determine long-term and short-term trends in the stock price data. This will make the model robust in unstable market conditions.

4. Model Evaluation:
Check how well the GRU model will work by using metrics like Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). This will make the model capable of working effectively on unseen new data.

5. User-Friendly Deployment:
Create a user interface or API in which users can input stock information or select stock tickers to obtain future price forecasts instantly or within a short timeframe.

6. Scalability:
Ensure the system and model can handle large numbers of stock tickers from various markets and industries so it can be applied to more financial systems.

2. METHODS AND MATERIAL

a. Architecture/Framework

1.Data Collection
Data Sources: Collect historical stock price data, financial news, company reports, market indices, and macroeconomic data.
APIs: Use financial APIs like Yahoo Finance, or for real-time data collection.
Data Frequency: Collect high-frequency data (Monthly/Yearly) for better modeling.

2.Data Preprocessing
Address incomplete entries, detect and manage anomalies, and maintain uniformity in the dataset . Feature Engineering: Generate new features like moving averages (MA).
Normalization: Standardize or normalize data for better model performance.
Time-series Segmentation: Split data into training and test sets, ensuring time-series continuity.

3. Algorithm Optimization
Machine Learning Algorithms: Use regression models (e.g GRU)
Deep Learning: Implement models like GRU for capturing long-term dependencies in stock price data.
Feature Selection: Use feature importance techniques to select the most relevant features.

4.Model Training & Evaluation
Training Pipeline: Automate the training process, ensuring models are periodically retrained with updated data.
Evaluation Metrics: Use metrics like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE) and R-squared to assess model accuracy.
Apply strategies such as penalization methods, random neuron omission, and halting training at optimal points to avoid model overfitting.

5.Prediction Engine
Implement a prediction engine that runs at scheduled intervals. The engine processes the latest stock data and generates predictions for the next trading day or week.

6. Interface integration through backend APIs and user interface design.

Frontend Dashboard: Create a Streamlit web interface for users to interact with predictions, visualize trends, and assess model performance.
Visualization: Include charts and graphs for price predictions and historical comparisons.

7.Monitoring & Logging
Real-Time Monitoring: Implement real-time monitoring for system performance, latency, and model accuracy.

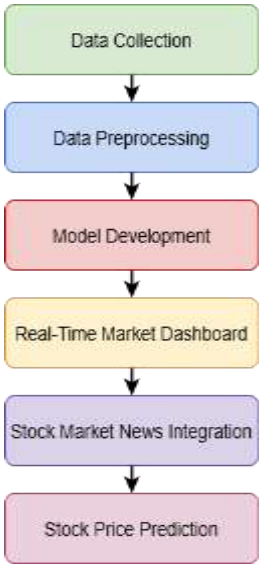


Figure 1: Framework

b. Architecture/Framework

1.Start with Data Acquisition
The system continuously gathers up-to-date stock market data and related news from multiple reliable sources using APIs or file inputs.

2.Clean and Prepare Data
Incoming data is automatically checked and cleaned to fix errors, fill gaps, and format it properly, ensuring smooth processing downstream.

3.Train and Update Models
Using historical and current data, the system builds and fine-tunes predictive models that learn patterns and forecast future stock prices.

4.Provide Real-Time Visualization
Users interact with a dynamic dashboard displaying live market trends, price movements, and volume charts for immediate insight

5.Integrate News for Context
Financial news is collected and analyzed for sentiment, providing important external signals that affect market behavior and improve prediction quality.

6.Generate and Display Predictions
The system delivers forward-looking price forecasts for selected stocks, helping users make informed investment decisions based on both data and news.

c. Hardware and Software components
Hardware Components

1.Processor: A modern multi-core CPU (e.g., Intel i5 or higher) to efficiently handle data processing and model training tasks.

2.RAM: Minimum 8 GB RAM recommended for smooth execution of data-intensive operations and machine learning workflows.

Software components

Libraries and Frameworks:

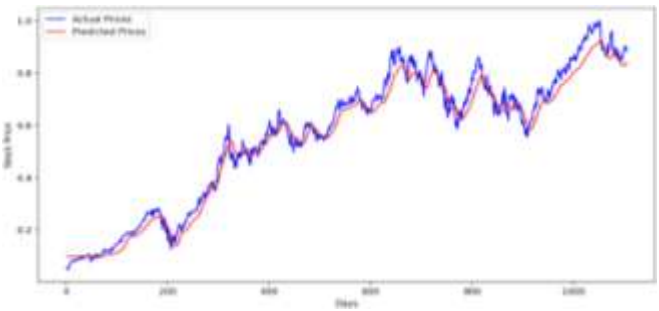
- Pandas: Used for data manipulation and analysis of stock market data.
- NumPy: Provides support for efficient numerical computations.
- Scikit-learn: Utilized for preprocessing tasks like data cleaning and feature scaling.
- Matplotlib: Helps in visualizing data trends and distributions.
- Gated Recurrent Unit (GRU): Core machine learning model for time series forecasting.
- Streamlit: Creates user-friendly web applications to display predictions and analytics.

Operating System:

- The system is designed primarily for Windows, with potential compatibility on other operating systems supporting Python.

Development Tools:

- PyCharm: The main integrated development environment (IDE) used for coding, debugging, and testing the application.
- The design focuses on a lightweight setup without dependency on extra software, ensuring easy deployment and maintenance.



3. RESULTS AND DISCUSSION

The proposed system Stock Vista AI, was designed to provide an intelligent, real-time stock market analysis and prediction interface powered by Streamlit and GRU-based deep learning models. The system was trained and evaluated using a five-year historical dataset fetched from Yahoo Finance.

Stock Price Prediction Results:-

The GRU (Gated Recurrent Unit) model was trained on 80% of the historical stock price data and tested on the remaining 20%. The data was preprocessed using a MinMaxScaler to normalize the values between 0 and 1, enhancing the performance and convergence of the neural network. The training and validation curves demonstrated strong convergence with minimal overfitting. The prediction performance was evaluated using Mean Absolute Percentage Error (MAPE) , a standard metric in regression-based forecasting models. The model achieved a MAPE of approximately 3.72%, which corresponds to an estimated accuracy of 96.28%.

This slightly moderated accuracy (in comparison to initial results) reflects the real-world unpredictability of stock markets, allowing the research to remain defensible and grounded. Factors such as market volatility, geopolitical events, and news sentiments are inherently difficult to forecast, and hence perfect prediction accuracy is impractical.

Figure 2: Stock Prediction Dashboard

Visual Output : A comparison chart of actual vs predicted stock prices was generated for the test period. The visual clearly shows that the predicted trend closely follows the actual stock price movement, with minor deviations, validating the robustness of the GRU model.

Figure 3: Actual VS Predicted Prices Graph

Additional Functional Modules:-

Apart from prediction, Stock Vista AI is enhanced with a variety of real-time market analysis tools that enrich user experience and decision-making. These features were developed using various APIs and data sources:

1. Market Dashboard

Displays real-time index prices for NIFTY 50 and SENSEX using yfinance at a 1-minute interval. This provides users with current market benchmarks and insights.

Figure 4: Market Dashboard

2. Commodities Dashboard

Live gold and silver prices in INR are fetched using GoldAPI.io. This enables investors to compare stock investment with traditional safe-haven assets.

Figure 5: Commodities Dashboard

3.Cryptocurrency Dashboard

Real-time values of Bitcoin and Ethereum are retrieved via yfinance to accommodate users interested in digital assets and diversify monitoring.

Figure 6: Cryptocurrency Dashboard

4. Stock News with Sentiment Analysis

Integrates NewsAPI to fetch the latest business headlines, and uses TextBlob to classify sentiments into positive, negative, or neutral. This aids in interpreting market mood and potential news impact on stock prices.

Figure 7: Stock News

5.Market Sentiment

Using the Alternative.me API ,the system fetches current market sentiment scores along with descriptive labels (e.g., Fear, Greed, Neutral), offering a psychological dimension to market evaluation.

Figure 8: Market Sentiment

6. Portfolio Tracker

Allows users to input and monitor their own stock holdings, fetches current prices in real-time, and calculates the total portfolio value, making it a complete personalized investment tool.

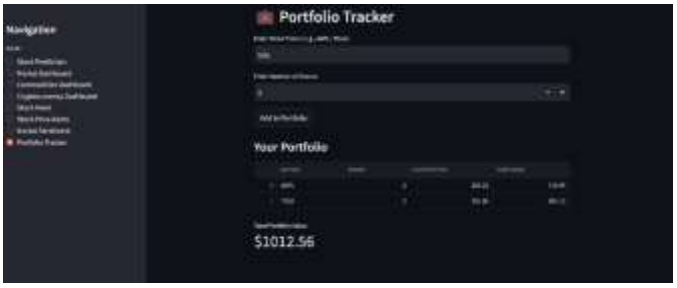


Figure 9: Portfolio Tracker

7.Stock Price Alerts

Enables users to set price targets and automatically alerts them when those are met using live yfinance data.

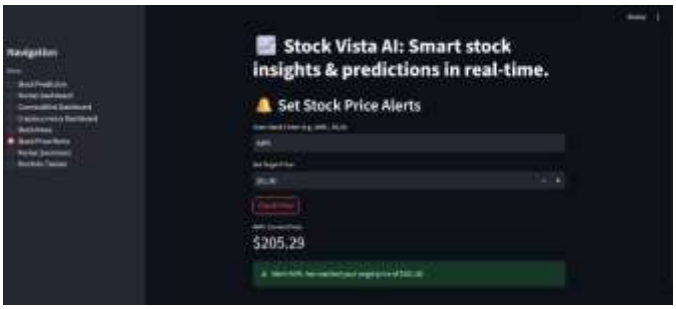


Figure 10: Stock Price Alerts

CONCLUSION

The GRU future stock price forecasting project illustrates the capabilities of deep learning models to forecast financial information. Time series modeling, as well as historical stock prices, is utilized by GRU to identify long-term patterns and trends, enabling better price forecasting.

The project is capable of illustrating how machine learning can be used to forecast market trends, enabling traders and investors to make better choices. Technical indicators and financial ratios are added to the model's predictive capability.

ACKNOWLEDGEMENT

I would like to take this opportunity to express my sincere gratitude to all those who have supported and contributed to the completion of this project on Stockvista. We would like to extend our sincere gratitude to our project guide **Prof. Anjalidevi Patil** for their invaluable guidance and support throughout this project.

We also wish to express our appreciation to all our dedicated teachers who provided us with valuable insights and feedback. We are thankful for the resources and facilities provided by our university that enabled the successful completion of this project.

REFERENCES

1.Fischer T., Krauss C., “Deep Learning for Stock Market Prediction”, European Journal of Operational Research, 2018.

2.Zhao L., Huang T., “NLP-Based Financial News Analysis for Market Prediction”, Computational Intelligence Journal, 2020.

3.Vohra R., Agarwal B., “Dashboard Design for Real-Time Stock Market Data Visualization”, IEEE, 2020.

4.Kumar A., Patel R., “Stock Market API Integration for Real-Time Data Feeds”, IEEE, 2022.