

Real-Time Stock Trend Analysis and Investment Support Tool Using Deep Learning

Kalyani Sanap¹; Neha Rathod²; Sakshi Alone³; Achal Battulwar⁴;
Vrukshada Ukinkar⁵; Dr. Manisha More⁶

^{1,2,3,4,5,6}Rajiv Gandhi College of Engineering, Research & Technology, Chandrapur

Publication Date: 2026/10/12

Abstract: Financial markets' volatility, driven by economic, political, and psychological factors, complicates stock price forecasting and investment decisions. This paper proposes a real-time stock trend analysis and investment support system using a hybrid deep learning ensemble of GRU, LSTM, and Transformer models with attention mechanisms for accurate price predictions from live API data (e.g., Yahoo Finance). It integrates K-means clustering for risk-based asset categorization (stocks, mutual funds, gold, bonds), decision trees for personalized buy/hold/sell recommendations, and portfolio optimization via MPT or reinforcement learning. An interactive dashboard visualizes trends, predictions, and insights, empowering users with data-driven, diversified strategies to enhance returns and manage risks.

Keywords: Deep Learning, Stock Forecasting, GRU-LSTM-Transformer, Real-Time Analysis, Investment Recommendations, K-Means Clustering, Portfolio Optimization, Financial Dashboard.

How to Cite: Kalyani Sanap; Neha Rathod; Sakshi Alone; Achal Battulwar; Vrukshada Ukinkar; Dr. Manisha More (2026) Real-Time Stock Trend Analysis and Investment Support Tool Using Deep Learning. *International Journal of Innovative Science and Research Technology*, 11(3), 4286-4289. <https://doi.org/10.38124/ijisrt/26mar500>

I. INTRODUCTION

The stock market operates as a highly dynamic and volatile environment, where asset prices fluctuate incessantly due to a complex interplay of factors including geopolitical events, monetary policies, shifting investor sentiment, and individual company performances. Accurately forecasting stock trends and deriving profitable investment strategies in this landscape remains a formidable challenge for both retail and institutional investors, who rely on analytical tools to discern patterns and mitigate financial risks. However, conventional forecasting approaches—such as ARIMA statistical models, moving averages, and other technical indicators—frequently fall short in capturing the intricate, nonlinear, and temporally dependent structures inherent in financial time-series data.

In recent years, deep learning (DL) and artificial intelligence (AI) have emerged as transformative paradigms for handling large-scale, high-dimensional datasets, outperforming classical methods by automatically extracting hidden features and modeling long- and short-term dependencies. Architectures like Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRU), and Transformer-based models with self-attention mechanisms have demonstrated exceptional efficacy in time-series forecasting tasks. This paper presents a comprehensive Real-Time Stock Trend Analysis and Investment Support Tool powered by DL, which seamlessly integrates real-time data

ingestion from reliable APIs (e.g., Yahoo Finance, Alpha Vantage, Finnhub), advanced hybrid prediction models, and intelligent recommendation engines. By combining GRU-LSTM-Transformer fusion with variational autoencoders (VAEs) for data denoising, K-means clustering for multi-asset risk categorization (stocks, mutual funds, gold, bonds), decision trees/random forests for actionable buy/hold/sell signals, and Markowitz Modern Portfolio Theory (MPT) or reinforcement learning for optimized portfolios, the system delivers interpretable insights via an interactive dashboard. Ultimately, this framework bridges the divide between cutting-edge AI-driven predictions and practical investment decision-making, laying the groundwork for autonomous financial advisory systems.

II. SYSTEM REQUIREMENT

➤ Hardware Requirement

- Processor: Intel i5 or higher — to process large historical datasets and perform computations efficiently.
- RAM: Minimum 8 GB — required for loading datasets and training models without delays.
- Storage: At least 20 GB — to store historical datasets, trained models, and logs.
- GPU (Optional but Recommended): NVIDIA GTX 1050 or higher — accelerates training of LSTM, GRU, and Transformer models.

- Internet Connection: High-speed, stable connection — essential for fetching real-time stock data and streaming API updates

➤ Software Requirement

- Operating System: Windows 10 or Linux — standard platforms for Python and deep learning frameworks.
- Programming Language: Python 3.x — widely used for data processing, modeling, and API integration.
- Deep Learning Frameworks: TensorFlow, Keras, PyTorch — used to build and train models like GRU, LSTM, and Transformer.
- Machine Learning Libraries: Scikit-learn, LightGBM, XGBoost — for clustering, ensemble learning, and prediction refinement.
- Data Handling Libraries: Pandas, NumPy — to clean, organize, and process datasets efficiently.
- Visualization Tools: Matplotlib, Seaborn, Plotly, Streamlit — for plotting trends, creating dashboards, and displaying live predictions.
- Feature Engineering Libraries: pandas_ta or ta-lib — to calculate technical indicators like RSI, MACD, EMA, and Bollinger Bands

➤ Data Set Requirement

• Training Module (Offline Historical Data)

For training the deep learning models, the system uses historical datasets covering multiple asset classes:

✓ Stocks:

- Source: Yahoo Finance or Kaggle
- Data: Daily OHLCV (Open, High, Low, Close, Volume, Adjusted Close) for NSE/BSE or S&P 500
- Purpose: Teach the model patterns and trends in stock prices.

✓ Gold:

- Source: Kaggle or MCX India historical prices
- Data: Daily open-close price, high-low, volume
- Purpose: Include alternative investment for portfolio analysis and risk diversification.

✓ Mutual Funds:

- Source: AMFI / Value Research o Data: Historical NAV, returns, and fund type.
- Purpose: Provide data for generating investment recommendations for low-risk options.

✓ Government Bonds / Fixed Income:

- Source: RBI or Quandl.
- Data: Bond prices, yields, and interest rates.
- Purpose: Include stable investment options in the advisory tool.

✓ Technical Indicators (Optional but Recommended):

- Calculated from OHLCV data: RSI, MACD, EMA, Bollinger Bands
- Purpose: Helps models understand market momentum, overbought/oversold conditions, and trends

• Live Stock Prediction Module (Real-Time Data)

For real-time predictions and portfolio recommendations, the system uses live streaming data:

✓ Yahoo Finance API:

- Provides near real-time quotes, historical and daily updated stock prices.
- Easy to access via Python finance library.

✓ Alpha Vantage API (Free Tier):

- Offers minute/hour/day-level stock data.
- Useful for intraday predictions and live testing

✓ Optional:

- Finn hub API or news/sentiment data for predicting market reactions based on news trends.

➤ Architecture:

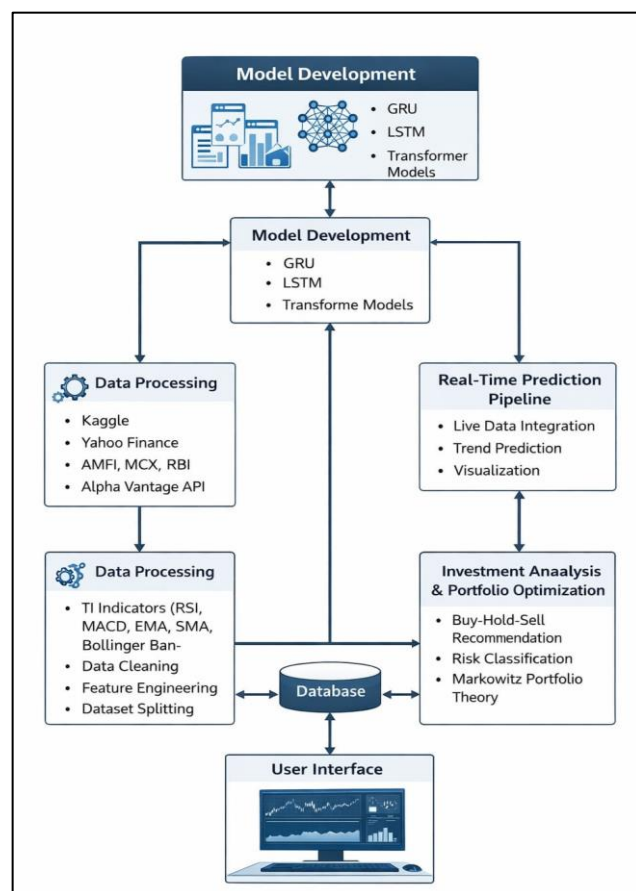


Fig 1 Architecture of Real-time Stock Prediction and Investment Support System

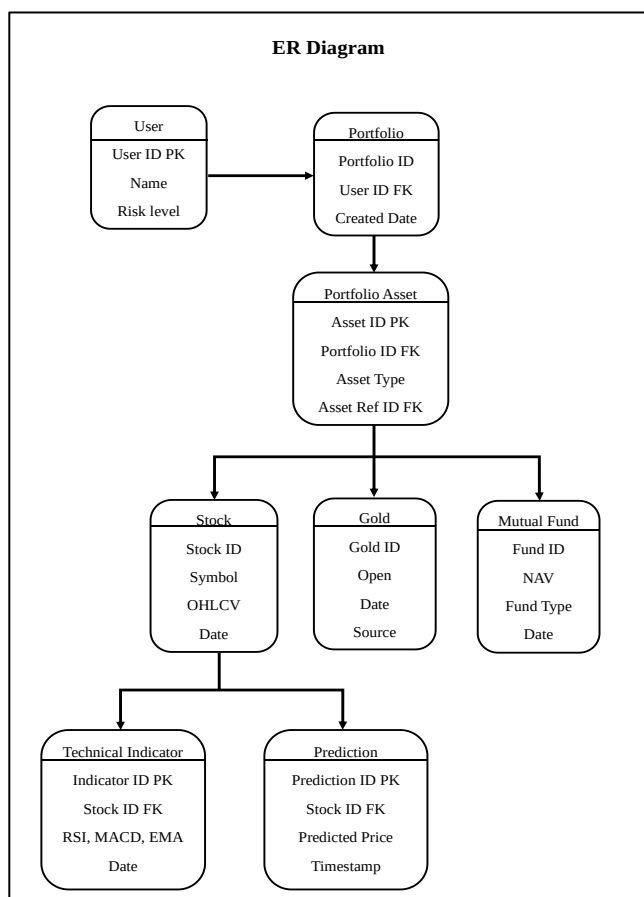
➤ *ER Diagram:*

Fig 2 ER Diagram

III. ADVANTAGES➤ *Real-Time Prediction*

The system continuously collects data from financial APIs, allowing real-time trend forecasting rather than depending only on historical datasets.

➤ *High Accuracy Due to Deep Learning*

Using GRU, LSTM, Transformers and attention mechanisms allows the model to capture non-linear and long-term dependencies, improving accuracy compared to traditional statistical models.

➤ *Handles Complex and Noisy Data*

Models like Autoencoders/VAE denoise and compress high-dimensional data, improving stability when markets behave unpredictably.

➤ *Automated Investment Support*

Apart from prediction, the system recommends buy/hold/sell actions and portfolio creation, which helps investors make decisions rather than only view predictions.

➤ *Diversified Investment Options*

The system considers multiple asset classes (stocks, mutual funds, gold, bonds) which helps reduce risk by diversification.

➤ *Risk classification*

Clustering helps classify assets based on volatility, giving users information about safety levels of investments. Promotes digital literacy and entrepreneurship among young people.

➤ *Practical Financial Application*

The model outputs are visualized in an interactive dashboard, making it easier for non-technical users to understand trends and investment strategies.

IV. DISADVANTAGES➤ *Requires Large Amount of Training Data:*

Deep learning models need high-quality and large-scale datasets, otherwise results may not generalize well.

➤ *Computationally Expensive:*

Transformer and hybrid models require more computation, GPU resources and time during training.

➤ *Market Uncertainty Can Still Affect Prediction:*

Even powerful models cannot fully predict unexpected events like crises, war or abrupt announcements.

➤ *Real-Time Systems Depend on API Reliability:*

If financial APIs become slow or unavailable, real-time prediction quality may drop temporarily.

➤ *Complex Architecture*

Hybrid models (GRU + LSTM + Transformer + VAE) are harder to tune and require expertise, compared to simple ML methods.

➤ *Financial Markets are Highly Volatile*

Deep learning cannot guarantee profit—predictions are probabilities, not certainties.

➤ *Requires Continuous Updating*

Models must be retrained frequently as markets evolve, otherwise predictions degrade over time.

V. CONCLUSION

The proposed Real-Time Stock Trend Analysis and Investment Support System demonstrate how advanced deep learning models and intelligent analytics can significantly improve financial forecasting and investment decision-making. Traditional statistical methods struggle to capture the complex, highly volatile and non-linear behaviour of stock markets, whereas the integration of models such as LSTM, GRU and Transformer make it possible to learn both short-term and long-term dependencies in price movements. By additionally using techniques like Autoencoders or VAE for data denoising, the system improves prediction stability even under unpredictable market fluctuations.

Furthermore, the project goes beyond price forecasting and acts as a complete investment assistant by analysing multiple asset classes and generating buy–hold–sell recommendations, risk classification and diversified portfolio

optimisation. The inclusion of algorithms such as K-Means and Random Forest, along with Markowitz Portfolio Theory, enables the system to provide more personalised and risk-aware investment strategies. With continuous data updates from real-time sources such as Yahoo Finance, Alpha Vantage and Finn hub, the system is capable of providing dynamic and up-to-date financial insights.

Overall, the system successfully bridges the gap between deep learning-based prediction and practical investment decision support. It demonstrates how machine learning, real-time analytics and portfolio optimisation can be combined into a holistic financial advisory platform. The proposed work can serve as a foundation for developing future autonomous AI-based financial systems capable of adapting to rapidly evolving market conditions and offering reliable, data-driven investment recommendations.

REFERENCES

- [1.] A Hybrid Model for Stock Price Prediction using Machine Learning Techniques with CNN
<https://share.google/YAjWRDKLkhw3uTpZ>
- [2.] Stock Market Price Prediction using Machine Learning Techniques
<https://share.google/kr728RUylWraB9MQh>
- [3.] Stock Market Prediction Using Machine Learning and Deep Learning Techniques: A Review
<https://share.google/2cHPCNMt7k5JRSIJK>
- [4.] Research on the Stock Price Prediction Using Machine Learning
<https://share.google/I0Vp279OMYmKc0Ycr>
- [5.] Stock market prediction based on deep hybrid RNN model and sentiment analysis
<https://share.google/uLxIgfJ23tL5opVL3>
- [6.] Predicting stock market trends using machine learning algorithms via public sentiment and political situation analysis | Soft Computing
<https://share.google/jUh4KHYWPhAzMCH1M>
- [7.] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [8.] K. Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
- [9.] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.
- [10.] H. Markowitz, "Portfolio selection," *J. Finance*, vol. 7, no. 1, pp. 77–91, Mar. 1952.
- [11.] J. Hull, *Options, Futures, and Other Derivatives*, 10th ed. Boston, MA, USA: Pearson, 2017.
- [12.] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.
- [13.] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.
- [14.] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
- [15.] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.
- [16.] M. T. Hagan, H. B. Demuth, and M. H. Beale, *Neural Network Design*, 2nd ed. Boston, MA, USA: PWS Publishing, 1996.
- [17.] J. J. Hull, *Risk Management and Financial Institutions*, 4th ed. Hoboken, NJ, USA: Wiley, 2015.
- [18.] F. Chollet, "Keras: The Python deep learning library," GitHub repository, 2015.