iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: VIII Month of publication: August 2025

DOI: https://doi.org/10.22214/ijraset.2025.73721

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VIII Aug 2025- Available at www.ijraset.com

# Market-Plus-Pulse: LSTM-Based Equity Price Prediction Framework with Real-Time Web Interface

Gagandeep H P[1], Yashaswini Y[2]
*Department of MCA, Navkis College of Engineering, HassanIndia*

***Abstract: Financial markets exhibit high volatility driven by economic indicators, global events, and investor sentiment, making accurate price prediction crucial for portfolio management and risk assessment. This paper presents a comprehensive framework for equity price prediction using Long Short-Term Memory (LSTM) networks integrated with a Django-based web application. The system incorporates historical market data extraction via APIs, advanced preprocessing techniques including normalization and feature engineering, and a bidirectional LSTM architecture for temporal pattern recognition. Performance evaluation on S&P 500 data demonstrates significant improvements over traditional ARIMA models, achieving 23.7% reduction in RMSE and 19.2% reduction in MAE. The web interface enables real-time prediction, comparative analysis, and risk assessment tools for both institutional and retail investors. This integrated approach bridges the gap between academic research and practical financial technology deployment.***
***Keywords: LSTM Networks, Stock Price Prediction, Financial Forecasting, Deep Learning, Web Application, Risk Management***

## I. INTRODUCTION

Financial markets represent complex adaptive systems characterized by non-linear dynamics, regime changes, and multi-scale temporal dependencies [1]. Traditional econometric models such as Autoregressive Integrated Moving Average (ARIMA) and Generalized Autoregressive Conditional Heteroskedasticity (GARCH) have dominated financial forecasting for decades. However, these approaches rely on linear assumptions and struggle to capture long-term dependencies inherent in financial time series [2].

The emergence of deep learning, particularly Recurrent Neural Networks (RNNs) and their variants, has revolutionized sequential data modeling. Long Short-Term Memory (LSTM) networks, introduced by Hochreiter and Schmidhuber [3], address the vanishing gradient problem in traditional RNNs and excel at learning long-range dependencies in temporal data.

This research presents Market-Plus-Pulse, an integrated framework combining LSTM-based prediction models with a user-friendly web interface for real-time financial analysis. The system addresses three critical gaps in existing literature: (1) lack of deployable, production-ready implementations, (2) limited integration of multiple data sources and indicators, and (3) absence of comprehensive user interfaces for practical decision-making.

## II. LITERATURE REVIEW

1) *Evolutionof Financial ForecastingModels:*Traditional econometric approaches have formed the foundation of financial forecasting. Vector Autoregression (VAR) models capture multivariate relationships but fail during structural breaks [4]. Box-Jenkins ARIMA models, while popular for their statistical rigor, assume linear relationships and stationary data [5]. Machine learning approaches have gained prominence with Support Vector Machines (SVM) and Random Forest showing promise for non-linear pattern recognition [6]. However, these methods treat time series as independent observations, ignoring temporal dependencies crucial for financial data.
2) *Deep Learning in Financial Prediction:*The application of deep learning to financial forecasting has shown remarkable progress. Jeenanunta et al. [7] demonstrated that LSTM networks reduce prediction error by 12% compared to traditional methods in Southeast Asian markets. Fischer and Krauss [8] applied LSTM to S&P 500 constituents, achieving superior risk-adjusted returns. Advanced architectures have emerged combining different neural network types. Ullah and Qasim [9] merged Convolutional Neural Networks (CNNs) with LSTMs for enhanced feature extraction. Selvin et al. [10] proposed CNN-LSTM hybrids, while Panwar et al. [11] confirmed neural network superiority over classical models across multiple asset classes.Recent developments include sentiment integration and attention mechanisms. Bhandari et al.

[12] incorporated news sentiment analysis, improving accuracy by 18%. Naik and Mohan [13] applied bidirectional LSTMs, while Shankar et al. [14] employed graph-based models for capturing market interconnectedness.

3) *Research Gap Identification:* Despite significant advances, several limitations persist in current research:
- *Deployment Gap:* Most studies focus on accuracy metrics without addressing real-world deployment challenges
- *Data Integration:* Limited incorporation of alternative data sources such as news sentiment and economic indicators
- *User Accessibility:* Absence of comprehensive platforms spanning data acquisition to decision support
- *Scalability:* Few frameworks address multi-asset, multi-timeframe analysis requirements

This research addresses these gaps through a modular, scalable framework designed for practical deployment.

## III. METHODOLOGY

### A. System Architecture

The Market-Plus-Pulse framework consists of four integrated components:
1) *Data Acquisition Module:* Real-time and historical data retrieval via Yahoo Finance API
2) *Preprocessing Engine:* Data cleaning, normalization, and feature engineering
3) *LSTM Prediction Model:* Bidirectional LSTM architecture for temporal pattern recognition
4) *Web Interface*: Django-based application for user interaction and visualization

### B. Data Collection and Preprocessing

Historical daily OHLCV (Open, High, Low, Close, Volume) data from 2001-2024 was collected for S&P 500 constituents. Data preprocessing included:

1) *Data Cleaning:*
- Outlier detection using Interquartile Range (IQR) method
- Missing value imputation using forward-fill for price data
- Volume normalization using log transformation

2) *Feature Engineering:*
- Simple Moving Averages (SMA): 5, 10, 20, 50 periods
- Exponential Moving Averages (EMA): 12, 26 period, Relative Strength Index (RSI): 14-period
- Moving Average Convergence Divergence (MACD)
- Bollinger Bands with 2 standard deviations
- Price momentum indicators

3) *Data Normalization: Min-Max scaling applied to all features:*

$$x\_normalized = (x - x\_min) / (x\_max - x\_min)$$

### C. LSTM Model Architecture

The implemented bidirectional LSTM architecture consists of:

1) *Layer Configuration:*
- Input Layer: 60 timesteps with 12 features
- Bidirectional LSTM Layer 1: 128 units, dropout=0.2
- Bidirectional LSTM Layer 2: 64 units, dropout=0.2
- Dense Layer 1: 32 units, ReLU activation
- Dense Layer 2: 16 units, ReLU activation
- Output Layer: 1 unit, linear activation

2) *Training Parameters:*
- Optimizer: Adam with learning rate 0.001
- Loss Function: Mean Squared Error (MSE)
- Batch Size: 32

- Epochs: 100 with early stopping (patience=15)
- Validation Split: 20%

## IV. EXPERIMENTAL RESULTS AND ANALYSIS

### A. Dataset Characteristics

The study utilized data from 20 S&P 500 companies representing different sectors:

1) Technology: AAPL, MSFT, GOOGL, NVDA, TSLA
2) Finance: JPM, BAC, GS, WFC, MS
3) Healthcare: JNJ, PFE, UNH, ABT, BMY
4) Consumer: AMZN, WMT, HD, DIS, NKE

Total dataset: 1,840,000 observations across all companies.

### B. Performance Comparison

Table I: Model Performance Comparison

| Model | RMSE | MAE | MAPE (%) | DA (%) |
|---|---|---|---|---|
| ARIMA | 0.0847 | 0.0523 | 8.34 | 52.3 |
| SVR | 0.0782 | 0.0527 | 7.89 | 54.7 |
| Random Forest | 0.0735 | 0.0521 | 7.12 | 56.8 |
| Standard LSTM | 0.0698 | 0.0489 | 6.45 | 58.9 |
| Bidirectional LSTm | 0.0096 | 0.0053 | 5.98 | 61. |

### C. Individual Stock Performance

Table II: Top 5 Stock Predictions (Bidirectional LSTM)

| Stock | RSME | MAE | MAPE(%) | DA(%) | Sharp Ratio |
|---|---|---|---|---|---|
| AAPL | 0.0634 | 0.0487 | 5.27 | 63.4 | 1.247 |
| MSFT | 0.0621 | 0.0479 | 5.98 | 62.1 | 1.186 |
| GOOGL | 0.0608 | 0.0472 | 5.89 | 60.8 | 1.156 |
| JPM | 0.0601 | 0.0534 | 6.45 | 59.7 | 1.098 |
| JNJ | 0.0712 | 0.0548 | 6.78 | 58.9 | 1.067 |

### D. Statistical Significance Testing:

Paired t-tests confirm statistical significance ($p < 0.01$) of LSTM improvements over traditional models across all evaluation metrics. The Diebold-Mariano test validates superior forecasting accuracy at 95% confidence level.

### E. Computational Performance:

Training time: 2.3 hours on NVIDIA RTX 4080 (average across all stocks) Inference time: 0.023 seconds per prediction Memory usage: 4.2 GB during training

## V. WEB APPLICATION IMPLEMENTATION

### A. Django Framework Architecture

The web application utilizes Django 4.2 with the following components:

1) *Backend Components*:

- Models: Database schemas for user data, predictions, and portfolios
- Views: API endpoints for data processing and model inference
- Middleware: Authentication, rate limiting, and error handling
- Celery Integration: Asynchronous task processing for model training

2) *Frontend Components:*
- HTML/CSS: Responsive design using Bootstrap 5
- JavaScript: Interactive charts using Chart.js and D3.js
- AJAX: Real-time data updates without page refresh

### B. Key Features

1) *Real-time Prediction Dashboard:*
- Live price feeds with 15-minute delay
- Next-day price predictions with confidence intervals
- Interactive candlestick charts with technical indicators

2) *Portfolio Analysis:*
- Multi-stock portfolio optimization
- Risk metrics calculation (VaR, CVaR, Sharpe Ratio)

Correlation matrix visualization

3) *Comparative Analysis:*
- Side-by-side stock comparison
- Sector performance analysis
- Historical vs. predicted price visualization

4) *Risk Management Tools:*
- Stop-loss and take-profit recommendations
- Position sizing based on risk tolerance
- Alert system for significant price movements

### C. User Interface Design

The interface follows modern UI/UX principles with:
1) Responsive grid layout for multiple device compatibility
2) Dark/light theme toggle for user preference
3) Intuitive navigation with breadcrumb functionality
4) Real-time data streaming with WebSocket integration

## VI. PRACTICAL APPLICATIONS

### A. Investment Strategy Support

The system supports multiple investment strategies:

1) *Trend Following:*
- Moving average crossover signals
- Momentum-based entry/exit points
- Risk-adjusted position sizing

2) *Mean Reversion:*
- Bollinger Band squeeze identification
- RSI oversold/overbought signals
- Support/resistance level prediction

3) *Portfolio Optimization:*
- Modern Portfolio Theory implementation
- Risk parity allocation strategies
- Dynamic rebalancing recommendations

### B. Risk Management Applications

*Value at Risk (VaR) Calculation:*

$VaR\_95\% = \mu - 1.645 \times \sigma$

Where $\mu$ represents expected return and $\sigma$ represents standard deviation.

*Expected Shortfall (ES):*

$ES\_95\% = E[R|R \leq VaR\_95\%]$

### *C. Target User Groups*

1) *Retail Investors:* Simplified interface with educational tooltips
2) *Financial Advisors:* Client portfolio management tools
3) *Institutional Traders:* Advanced analytics and API access
4) *Academic Researchers:* Model comparison and backtesting capabilities

## VII.LIMITATIONS AND FUTURE WORK

A. *Current Limitations:*Our machine learning framework, while promising, isn't without its challenges. The biggest hurdle we face is how the model behaves when markets enter uncharted territory. When unexpected events shake up the financial world—think 2008 financial crisis or the COVID-19 pandemic—our model struggles because it's never seen anything quite like these situations before. We're also heavily dependent on getting clean, reliable data every day. If our data feeds go down or we start getting garbage data, our predictions quickly become unreliable. Another reality we can't ignore is that training these models requires serious computing power, which means smaller trading firms or individual investors might find it tough to implement. There's also the regulatory side of things—we can only work with historical patterns and can't incorporate the kind of fundamental analysis that human traders rely on, which means we might miss important market-moving news or economic shifts.

B. *Future Enhancements:*We have exciting plans to make this system much more robust and accessible. On the technical front, we're looking at implementing Transformer models—the same technology that powers ChatGPT—to help our system better understand long-term market trends. We're also exploring Graph Neural Networks to map out how different markets influence each other, and GANs to create realistic "what-if" scenarios for stress testing our strategies. Beyond just price data, we want to teach our system to read the news and gauge market sentiment from social media buzz, incorporate economic indicators like unemployment rates and GDP growth, and even use satellite imagery to predict commodity prices based on crop yields or oil storage levels. To make this technology more accessible, we're working on versions that can run on local computers and mobile apps, so traders don't need to rely on cloud connections. We're also excited about blockchain possibilities—imagine having a transparent record of every prediction our model makes, or automatically executing trades through smart contracts. Perhaps most importantly, we're committed to building in safeguards against bias and making sure our AI decisions can be explained in plain English, especially as regulators start paying closer attention to algorithmic trading.

## VIII. CONCLUSION

This research presents Market-Plus-Pulse, a comprehensive LSTM-based framework for equity price prediction that successfully bridges academic research and practical application. The bidirectional LSTM architecture demonstrates significant improvements over traditional forecasting methods, achieving 23.7% reduction in RMSE compared to ARIMA models and 61.2% directional accuracy.The integrated Django web application provides institutional-grade functionality through an intuitive interface, supporting real-time prediction, portfolio optimization, and risk management. The system's modular architecture ensures scalability and adaptability to evolving market conditions and user requirements.Key contributions include: (1) comprehensive comparison of deep learning architectures for financial prediction, (2) production-ready implementation with robust web interface, (3) integration of multiple technical indicators and risk metrics, and (4) statistical validation of performance improvements across diverse market conditions.The framework's practical impact extends beyond academic metrics, providing actionable insights for investment decision-making, risk assessment, and portfolio management. Future enhancements incorporating transformer architectures, alternative data sources, and edge computing capabilities position this research at the forefront of financial technology innovation.The success of this implementation demonstrates the potential for AI-driven financial analysis tools to democratize sophisticated investment strategies while maintaining the rigor required for institutional applications.

## IX. ACKNOWLEDGMENT

The authors acknowledge the support provided by Navkis College of Engineering and express gratitude to the open-source community for the tools and libraries that made this research possible.

Special thanks to the Yahoo Finance API team for providing accessible financial data and to the TensorFlow development community for the deep learning framework.

## REFERENCES

[1] Makridakis, S., et al. 2021. Advanced Econometric Models for Multivariate Forecasting. International Journal of Forecasting.
[2] Gers, F. A., et al. 2020. Gated Recurrent Units in Modern RNNs. Neural Networks Review.
[3] Lütkepohl, H. 2021. VAR Models in Contemporary Econometrics. Journal of Econometrics.
[4] Friedman, J. H. 2021. Gradient Boosting for Nonlinear Regressions. Annals of Statistics.
[5] Jeenanunta, C., et al. 2020. Stock Price Prediction With Long Short-Term Memory Recurrent Neural Network. IEEE Conference.
[6] Ullah, K., & Qasim, M. 2020. Google Stock Prices Prediction Using Deep Learning. IEEE.
[7] Selvin, S., et al. 2017. Stock Price Prediction Using LSTM, RNN and CNN-Sliding Window Model. IEEE.
[8] Panwar, B., et al. 2021. Stock Price Prediction Using Regression and SVM. IEEE.
[9] Bhandari, H. N., et al. 2022. Predicting Stock Market Index Using LSTM. Machine Learning with Applications.
[10] Naik, N., & Mohan, B. R. 2022. Study of Stock Return Predictions Using Recurrent Neural Networks with LSTM. Engineering Applications of Neural Networks.
[11] Shankar, P., et al. 2022. Design and Development of an Ensemble Model for Stock Market Prediction Using LSTM, ARIMA, and Sentiment Analysis. Deep Learning Techniques.
[12] Diqi, M. 2022. StockTM: Accurate Stock Price Prediction Model Using LSTM. International Journal of Informatics and Computation.
[13] Das, S., et al. 2022. Stock Price Prediction Using Bi-LSTM and GRU-Based Hybrid Deep Learning Approach. Proceedings of Doctoral Symposium.
[14] Karim, M. E., et al. 2022. A Deep Learning-Based Approach for Stock Price Prediction Using Bidirectional Gated Recurrent Unit and Bidirectional Long Short Term Memory Model. Neural Computing and Applications.
[15] Naik, N. 2022. Study of Stock Return Predictions Using Recurrent Neural Networks with LSTM. International Conference.
[16] Banyal, S., et al. 2023. Indian Stock-Market Prediction Using Stacked LSTM and Multi-Layered Perceptron. International Journal of Innovative Technology.