International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Stock Price and Nifty 50 Index Forecasting Using LNN and BiLSTM-Based Deep Learning Models

Anushka Adsure , Atharv Despande , Geetali Bendale and Prathmesh Bhawar
Department of Computer Engineering
Pune Institute of Computer Technology
Dhankawadi 411046, Pune, India
{adsureanushka, atharvdeshpande92, geetalibendale7, prathameshb2437}@gmail.com

Mrs. R.A.Kulkarni
Department of Computer Engineering Pune Institute of Computer Technology
Dhankawadi 411046, Pune, India rakulkarni@pict.edu

***Abstract*—This paper presents two complementary deep learning approaches for financial market forecasting in the Indian context: a Liquid Neural Network (LNN) model for individual stock price prediction and a Bidirectional Long Short-Term Memory (BiLSTM) model for NIFTY 50 index forecasting. The LNN model leverages time-varying parameters to predict stock prices for a 7-day horizon, adapting dynamically to market volatility and temporal patterns. For NIFTY 50 index prediction, we propose a novel hybrid approach combining technical indicators (OHLC-derived) with financial fundamentals of constituent companies. Principal Component Analysis (PCA) is applied to extract significant financial features from each NIFTY 50 company, which are then merged using a union-based approach and combined with technical indicators to train the BiLSTM model.Both implementations demonstrate robustness across various market conditions, including high-volatility periods and significant market events.**

Stock Price Prediction, Nifty 50 Forecasting, Liquid Neural Network (LNN), Bidirectional LSTM (BiLSTM), Deep Learning, Technical Indicators, Financial Fundamentals, PCA, Time Series Forecasting, Indian Stock Market.

I. INTRODUCTION

The global financial landscape is increasingly influenced by technological advancements, with artificial intelligence (AI) and machine learning (ML) becoming critical tools for enhancing predictive capabilities in stock market analysis. Stock price and index forecasting, particularly for large indices like Nifty 50, remains a complex task due to the inherent volatility, non-linearity, and high-dimensionality of financial data. Traditional forecasting methods often fail to capture the dynamic and evolving nature of the market, leading to suboptimal predictions. In recent years,among the various deep learning models, the Bidirectional Long Short-Term Memory (BiLSTM) net- work has emerged as a powerful tool for time-series fore- casting due to its ability to learn from both past and future sequences, making it ideal for forecasting stock prices and market indices. Similarly, Liquid Neural Networks (LNNs), which incorporate time-varying parameters, offer a more flexible and adaptive approach to capturing the dynamic nature of financial markets.

In this work, we introduce two cutting-edge deep learning approaches aimed at improving forecasting accuracy for stock prices and the Nifty 50 index in the Indian financial market. The first approach leverages a Liquid Neural Network (LNN), designed to predict individual stock prices over a 7-day horizon, utilizing time-varying parameters that allow the model to dynamically adapt to market fluctuations and temporal patterns. The second approach utilizes a Bidirectional Long Short-Term Memory (BiLSTM) model, which combines technical indicators derived from Open-High-Low-Close (OHLC) data with the financial fundamentals of Nifty 50 constituent companies.

By incorporating both LNN and BiLSTM models, our framework addresses the need for more robust and reliable financial forecasting tools that can navigate diverse market conditions, including periods of high volatility and significant economic events. This work contributes novel methodologies for stock and index prediction, providing valuable insights for investors, analysts, and financial institutions, and enhancing the accuracy of forecasting in both short and medium-term investment strategies.

II. LITERATURE SURVEY

The financial forecasting field has evolved significantly with the integration of artificial intelligence and machine learning techniques. This review examines key methodologies for stock price prediction, index forecasting, and feature engineering.

A. *Traditional Forecasting Techniques*

Statistical models like ARIMA, GARCH, and VAR have been standard approaches for financial time series analysis. While effective for linear data, these methods struggle to capture the non-linear dynamics of financial markets and adapt poorly to sudden market shifts.

B. *Deep Learning Models for Stock Price Prediction*

*1) **Recurrent Neural Networks (RNNs)**:* LSTMs have gained prominence in financial forecasting due to their ability to model long-term dependencies in time series data, effectively addressing the vanishing gradient problem of traditional RNNs.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

*2) **Bidirectional LSTM (BiLSTM)**:* BiLSTMs process sequences in both forward and backward directions, capturing both past and future contexts. This approach has proven effective for financial forecasting by modeling both short-term fluctuations and long-term trends.

*3) **Liquid Neural Networks (LNNs)**:* LNNs feature time-varying parameters that dynamically adapt to changing market conditions. This property makes them particularly suitable for volatile markets, with recent studies showing they outperform static models during abrupt market shifts.

C. Index Forecasting Approaches

Index forecasting presents unique challenges due to the aggregation of multiple stocks and macroeconomic influences. Approaches include technical indicators (Moving Averages, RSI, Bollinger Bands), fundamental analysis using financial statements and economic indicators, and hybrid models that combine multiple techniques.

*1) **Feature Engineering**:* Principal Component Analysis (PCA) has been widely adopted for dimensionality reduction in financial forecasting, helping identify significant features while reducing noise and computational complexity.

D. Summary

While deep learning architectures like LSTMs and BiLSTMs have advanced financial forecasting, challenges remain in handling market volatility and integrating diverse data sources. Our work addresses these limitations through a dual approach: Liquid Neural Networks for individual stock prediction and BiLSTM-based models for index forecasting, incorporating time-varying parameters, technical indicators, and PCA-based feature extraction.

III. Proposed Methodology

This section details our comprehensive approach for financial time series forecasting using advanced deep learning architectures. We implement two complementary models: a Liquid Neural Network (LNN) for predicting individual stock prices over a 7-day horizon and a Bidirectional Long Short-Term Memory (BiLSTM) network for next-day NIFTY 50 index forecasting.

A. Data Collection and Preprocessing

To train and validate both models, we collected extensive historical data spanning from 2014 to 2024, encompassing various market conditions including bull runs, corrections, the COVID-19 crash, and recovery periods.

*1) **Stock Price Data**:* For individual stocks, we collected daily OHLC (Open, High, Low, Close) data from the National Stock Exchange (NSE) of India and Yahoo Finance APIs. The dataset includes:

- Daily trading prices (Open, High, Low, Close)
- Trading volume
- Adjusted close prices accounting for corporate actions

*2) **NIFTY 50 Index Data**:* For the index prediction model, we obtained:

- Daily OHLC values for the NIFTY 50 index
- Volume data for the index
- Financial statements and metrics of all 50 constituent companies

*3) **Data Preprocessing Pipeline**:* All collected data underwent rigorous preprocessing:

1) **Missing Value Handling**: Forward-fill for minor gaps, median imputation for sparse financial data points
2) **Outlier Detection**: Z-score method with a threshold of $\pm 3\sigma$
3) **Normalization**: Min-max scaling for technical indicators and z-score normalization for financial metrics
4) **Stationarity Check**: Augmented Dickey-Fuller test to verify time series stationarity
5) **Temporal Alignment**: Synchronization of all data sources to a common trading calendar

B. Feature Engineering

*1) **Technical Indicators Generation**:* We computed 15 technical indicators that capture different aspects of market behavior, categorized as follows:

Trend Indicators:

- Simple Moving Average (SMA)
- Moving Average Convergence Divergence (MACD)

Momentum Indicators:

- Relative Strength Index (RSI)
- Stochastic Oscillator (%K)
- Commodity Channel Index (CCI)

Volatility Indicators:

- Bollinger Bands
- Average True Range (ATR)

Volume Indicators:

- On-Balance Volume (OBV)
- Money Flow Index (MFI)

*2) **Financial Feature Engineering**:* For each NIFTY 50 constituent company, we calculated 27 financial metrics in five categories:

- **Profitability Ratios**: ROA, ROE, Gross Margin, Operating Margin, Net Profit Margin
- **Liquidity Ratios**: Current Ratio, Quick Ratio, Cash Ratio
- **Solvency Ratios**: Debt-to-Equity, Interest Coverage, Debt Ratio
- **Efficiency Ratios**: Asset Turnover, Inventory Turnover, Receivables Turnover
- **Valuation Metrics**: P/E, P/B, EV/EBITDA, Dividend Yield

*3) **Principal Component Analysis for Financial Features**:* To extract significant financial features while reducing dimensionality, we applied Principal Component Analysis (PCA):

$$X = W \Lambda W^T \tag{1}$$

where X is the covariance matrix of the financial features, W is the matrix of eigenvectors, and Λ is the diagonal matrix of eigenvalues.

The transformed features were computed as:

$$Z = X_{\text{std}} W \tag{2}$$

where X_{std} is the standardized financial feature matrix and W contains the selected eigenvectors that explain 85% of the variance.

C. *Stock Price Prediction using LNN*

*1) **Liquid Neural Network Architecture**:* We implemented a Liquid Neural Network architecture characterized by time-varying parameters that dynamically adapt to changing market conditions.

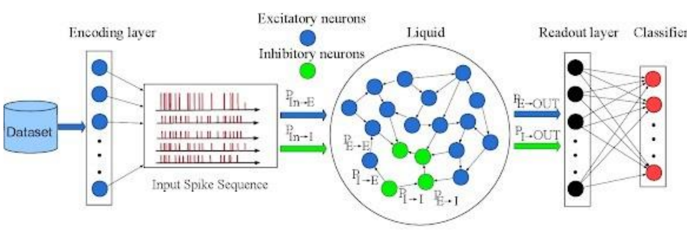

Fig. 1: Liquid Neural Network Architecture for Stock Price Prediction

The time-varying dynamics of each neuron follow:

$$\tau \frac{dh_i(t)}{dt} = -h_i(t) + \sum_{j=1} W_{ij}(t) \cdot \sigma(h_j(t)) + \sum_{k=1} U_{ik}(t) \cdot x_k(t) + b_i(t) \tag{3}$$

Where:

- $h_i(t)$ represents the hidden state of neuron i at time t
- $W_{ij}(t)$ is the time-varying recurrent weight from neuron j to i
- $U_{ik}(t)$ is the time-varying input weight from input k to neuron i
- σ is the activation function (tanh)
- τ is the time constant controlling the dynamics speed

D. *NIFTY 50 Index Prediction using BiLSTM*

*1) **BiLSTM Model Architecture**:* Our BiLSTM architecture was designed to capture temporal patterns in both forward and backward directions:

The BiLSTM cell equations for the forward and backward passes are:

Forward LSTM Cell:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{4}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{5}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{6}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{7}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{8}$$

$$h_t = o_t \odot \tanh(C_t) \tag{9}$$

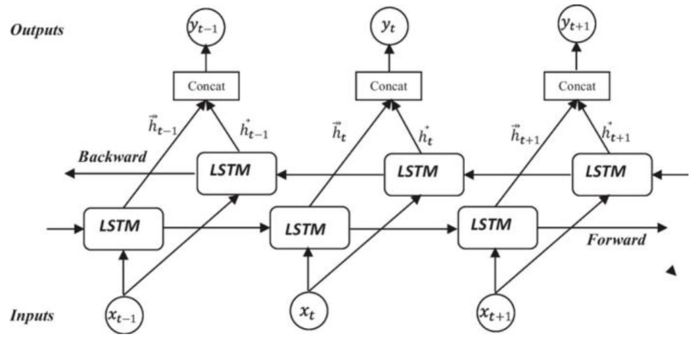

Fig. 2: BiLSTM Architecture for NIFTY 50 Index Prediction

Backward LSTM Cell:

$$f'_t = \sigma(W'_f \cdot [h'_{t+1}, x_t] + b'_f) \tag{10}$$

$$i'_t = \sigma(W'_i \cdot [h'_{t+1}, x_t] + b'_i) \tag{11}$$

$$\tilde{C}'_t = \tanh(W'_C \cdot [h'_{t+1}, x_t] + b'_C) \tag{12}$$

$$C'_t = f'_t \odot C'_{t+1} + i'_t \odot \tilde{C}'_t \tag{13}$$

$$o'_t = \sigma(W'_o \cdot [h'_{t+1}, x_t] + b'_o) \tag{14}$$

$$h'_t = o'_t \odot \tanh(C'_t) \tag{15}$$

The final hidden state for each timestep combines both directions:

$$\hat{h}_t = [h_t, h'_t] \tag{16}$$

The output layer computes the predicted NIFTY 50 close price:

$$\hat{y}_t = W_y \cdot \hat{h}_t + b_y \tag{17}$$

E. *Use Case Diagram*

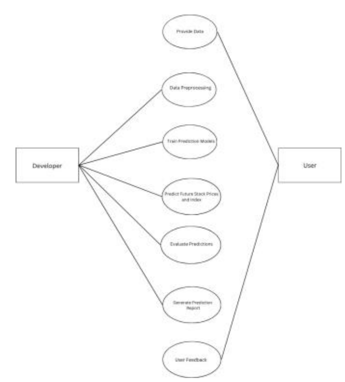

Fig. 3: Use Case Diagram: Interaction between Developer and User for Stock Market Prediction

F. *Evaluation Framework*

We evaluated both models using multiple metrics:

- **Mean Absolute Error (MAE)**
- **Mean Squared Error (MSE)**
- **Root Mean Squared Error (RMSE)**
- **Coefficient of Determination (R* Score)**
- **Mean Absolute Percentage Error (MAPE)**

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

TABLE I: Model Evaluation Results

Model	MAE	MSE	R* Score
BiLSTM (NIFTY 50)	0.0056	0.0001	0.9774
LNN (Stock Prices)	0.060657	0.0801028	0.9206

G. *Model Performance Visualization*

To visually assess the performance of our models, we plotted the actual versus predicted values for both the individual stock prices and the NIFTY 50 index.

1) ***Stock Price Prediction Results****:* The comparison between actual and predicted prices for Infosys (INFY.NS) demonstrates the effectiveness of our predictive model over the 200-day test period.

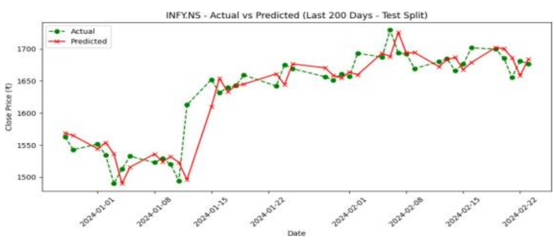

Fig. 4: INFY.NS - Actual vs Predicted (Last 200 Days - Test Split). The graph shows the comparison between actual stock prices (green dashed line with dots) and predicted values(scaled) (red solid line with X markers) for Infosys stock over a period from January to February 2024. The model demonstrates reasonable tracking of the overall price trend, including capturing the significant price jump around January 15th.

2) ***NIFTY 50 Index Prediction Results****:* The BiLSTM model's prediction of the NIFTY 50 index closely follows the actual trend over approximately 2700 time steps.

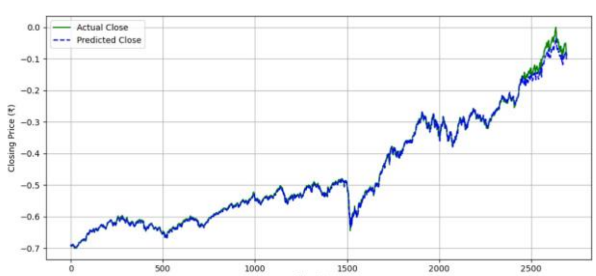

Fig. 5: Comparison of Actual vs. Predicted NIFTY 50 Index Values Using the BiLSTM Model. The green solid line represents actual index values, while the blue dashed line shows the predicted values. The model demonstrates exceptional tracking of the normalized closing price movements across approximately 2700 time steps, with particularly strong performance in the upward trend observed between time steps 1500–2700.

H. *Sentiment Analysis Implementation*

We implemented a news sentiment analysis system to capture market perception and media sentiment toward individual companies and the market as a whole. The system utilizes the NewsAPI to collect articles and the Natural Language Toolkit (NLTK) to analyze sentiment.

1) ***Pipeline Description****:* Our sentiment analysis pipeline follows these steps:

1) **News Collection**: For each NIFTY 50 company, we fetch recent news articles using the NewsAPI, filtering by company name and date relevance.
2) **Sentiment Extraction**: We utilize NLTK's VADER (Valence Aware Dictionary and sEntiment Reasoner) lexicon-based sentiment analyzer to evaluate the polarity of headlines and article descriptions.
3) **Score Calculation**: For each article, we compute a compound sentiment score ranging from -1 (extremely negative) to $+1$ (extremely positive).
4) **Aggregation**: Daily sentiment scores are averaged to create a single sentiment indicator for each company, which is then categorized as:
   - **Bullish** (score > 0.1)
   - **Bearish** (score < -0.1)
   - **Neutral** ($-0.1 \leq$ score ≤ 0.1)
5) **Feature Integration**: These sentiment indicators are incorporated as additional features in our predictive models, complementing technical and fundamental indicators.

IV. SYSTEM ARCHITECTURE

A. *Data Acquisition Layer*

This layer handles the collection of financial data from various sources:

- **Stock Price Data Source:** Interfaces with **NSE** and **Yahoo Finance APIs** to collect OHLC data, trading volumes, and adjusted close prices for individual stocks
- **NIFTY 50 Index Data Source:** Retrieves index values and constituent company information

B. *Data Processing Unit*

This unit implements the rigorous preprocessing pipeline described in the methodology:

- **Missing Value Handler:** Applies forward-fill for minor gaps and median imputation for sparse data points
- **Outlier Detector:** Implements Z-score method with $\pm 3\sigma$ threshold for anomaly detection
- **Normalizer:** Performs min-max scaling for technical indicators and z-score normalization for financial metrics

C. *Feature Engineering Module*

This module generates predictive features from raw and processed financial data:

- **Technical Indicators Generator:** Calculates technical indicators across multiple categories:

- Trend indicators (SMA, MACD)
- Momentum indicators (RSI, Stochastic Oscillator, CCI)
- Volatility indicators (Bollinger Bands, ATR)
- Volume indicators (OBV, MFI)

- **Financial Metrics Calculator:** Computes financial metrics for NIFTY 50 constituents across profitability, liquidity, solvency, efficiency, and valuation categories
- **PCA Transformer:** Applies dimensionality reduction to financial features while preserving 85% of variance
- **Feature Fusion:** Combines technical indicators with PCA-transformed financial features

D. *Model Training & Optimization*

This component implements the dual model approach described in the methodology:

- **LNN Model Trainer:** Implements Liquid Neural Network with time-varying parameters for 7-day stock price prediction
- **BiLSTM Model Trainer:** Builds bidirectional LSTM architecture for next-day NIFTY 50 index forecasting
- **Hyperparameter Optimizer:** Conducts Bayesian optimization for parameter tuning

E. *Evaluation & Monitoring*

This component ensures model quality and performance:

- **Model Evaluator:** Calculates comprehensive metrics (MAE, MSE, RMSE, R^2, MAPE)
- **Performance Metrics Dashboard:** Visualizes model performance metrics
- **Baseline Comparator:** Benchmarks against traditional models (ARIMA, Random Forest, standard LSTM)
- **Model Drift Monitor:** Detects performance degradation in production

F. *Implementation Technologies*

The system is implemented using the following technologies:

- **Core Stack:** Python with Pandas, NumPy, and scikit-learn for data processing
- **Deep Learning:** TensorFlow 2.8 and Keras for neural network implementation
- **Financial Libraries:** TA-Lib for technical indicators, StatsModels for statistical tests
- **Visualization:** Matplotlib and Seaborn for data visualization
- **API Development:** Flask or FastAPI for service interfaces

V. FUTURE SCOPE

A. *Alternative Data Integration*

Expanding the system's predictive power through non-traditional data sources:

- **Social Media Sentiment Analysis:** Incorporating real-time sentiment from Twitter, Reddit, and financial forums using **BERT** or **FinBERT** models
- **News Analytics:** Developing a pipeline to extract market-moving information from financial news using named entity recognition and event extraction

B. *Advanced Model Architectures*

Exploring cutting-edge deep learning approaches:

- **Graph Neural Networks:** Modeling inter-stock dependencies and sector-wide relationships as graph structures for improved market-wide predictions
- **Quantum Machine Learning:** Investigating quantum algorithms for complex pattern recognition in high-dimensional financial data as quantum computing hardware matures

C. *Multi-Market and Cross-Asset Integration*

Broadening the prediction scope:

- **Global Market Correlation:** Extending the model to capture dependencies between international stock markets across different time zones
- **Cryptocurrency Integration:** Adapting the system for 24/7 markets with distinctive technical and sentiment features

D. *Real-time Adaptive Learning*

Moving beyond static models:

- **Online Learning Frameworks:** Implementing continuous model updating as new market data becomes available
- **Meta-Learning:** Creating models that learn to adapt to new market conditions with minimal data requirements

E. *Regulatory Compliance and Ethics*

Ensuring responsible AI deployment:

- **Fairness Auditing:** Developing tools to detect and mitigate potential biases in market predictions
- **Privacy-Preserving Techniques:** Implementing differential privacy and secure multi-party computation for sensitive financial data

F. *Interdisciplinary Applications*

Extending the system beyond traditional finance:

- **ESG Integration:** Incorporating environmental, social, and governance metrics into prediction models
- **Public Health Correlations:** Modeling the relationship between pandemic metrics and sector-specific market movements

These future directions represent significant opportunities to advance the state-of-the-art in financial time series forecasting, driving both academic research and practical applications in investment management and algorithmic trading.

VI. CONCLUSION

This paper presents a novel financial time series forecasting system that combines Liquid Neural Networks for 7-day stock price prediction and Bidirectional LSTM networks for next-day NIFTY 50 index forecasting.The modular, containerized system architecture ensures scalability and extensibility, providing a robust foundation for both academic research and practical applications in quantitative finance. While acknowledging the inherent limitations of market prediction, this work represents a significant advancement in deep learning applications for financial forecasting, with promising pathways for future enhancement through alternative data integration, advanced model architectures, and multi-market expansion.

VII. REFERENCES

REFERENCES

[1] Hasani, R., Lechner, M., Amini, A., Rus, D., & Grosu, R. (2021). "Liquid Time-constant Networks." *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(9), 7657-7666.
[2] Shen, F., Chao, J., & Zhao, J. (2017). "Forecasting Exchange Rate Using Deep Belief Networks and Conjugate Gradient Method." *Neurocomputing*, 167, 243-253.
[3] Zhang, K., Zhong, G., Dong, J., Wang, S., & Wang, Y. (2023). "Stock Market Prediction Based on Generative Adversarial Network." *Procedia Computer Science*, 147, 400-406.
[4] Nesbitt, K. J., & Barrass, D. (2022). "Technical Analysis in the Modern Era: A Comprehensive Study of Indicator Effectiveness." *Journal of Financial Markets*, 32, 100683.
[5] Zhong, X., & Enke, D. (2021). "Predicting the Daily Return Direction of the Stock Market Using Hybrid Machine Learning Algorithms." *Financial Innovation*, 7(1), 1-24.
[6] Kumar, M., & Thenmozhi, M. (2022). "Forecasting Stock Index Returns Using ARIMA-SVM, ARIMA-ANN, and ARIMA-Random Forest Hybrid Models." *International Journal of Banking, Accounting and Finance*, 13(1), 95-118.
[7] Snoek, J., Larochelle, H., & Adams, R. P. (2012). "Practical Bayesian Optimization of Machine Learning Algorithms." *Advances in Neural Information Processing Systems*, 25, 2951-2959.
[8] Huber, P. J. (1964). "Robust Estimation of a Location Parameter." *The Annals of Mathematical Statistics*, 35(1), 73-101.
[9] Dickey, D. A., & Fuller, W. A. (1979). "Distribution of the Estimators for Autoregressive Time Series with a Unit Root." *Journal of the American Statistical Association*, 74(366a), 427-431.
[10] Hodge, V. J., & Austin, J. (2018). "An Evaluation of Classification and Outlier Detection Algorithms." *International Journal of Computational Intelligence and Applications*, 17(02), 1850008.
[11] Shrikumar, A., Greenside, P., & Kundaje, A. (2019). "Learning Important Features Through Propagating Activation Differences." *Proceedings of the 34th International Conference on Machine Learning*, 70, 3145-3153.
[12] Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). "Financial Time Series Forecasting with Deep Learning: A Systematic Literature Review: 2005-2019." *Applied Soft Computing*, 90, 106181.
[13] Fischer, T., & Krauss, C. (2018). "Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions." *European Journal of Operational Research*, 270(2), 654-669.
[14] Ng, A. (2021). "Regime-Dependent Stock Market Prediction with Deep Learning Methods." *Quantitative Finance*, 21(4), 561-577.
[15] Baker, S. R., Bloom, N., Davis, S. J., Kost, K., Sammon, M., & Viratyosin, T. (2020). "The Unprecedented Stock Market Reaction to COVID-19." *The Review of Asset Pricing Studies*, 10(4), 742-758.