Stock Market Price Prediction Based on Attention-LSTM Model with Multi-Feature Bayesian Optimization

Yuhan Zhang [*], Yishan Liu

Department of International Economics, China Foreign Affairs University, Beijing, China

* Corresponding Author Email: m18810381901@163.com

Abstract. Stock price prediction plays a critical role in investment decision-making and financial regulation. However, traditional time series models and early neural networks are limited either by restrictive assumptions or by their inability to effectively handle long sequences, resulting in suboptimal prediction performance. This paper proposes a hybrid predictive model that integrates multi-feature fusion, the attention mechanism, and Bayesian optimization into a Long Short-Term Memory (LSTM) framework to enhance prediction accuracy and stability. Using daily data from the S&P 500 Index from 2020 to 2022, the study employs LSTM to capture long-term temporal dependencies, introduces an attention mechanism to highlight key sequential features, and utilizes Bayesian optimization for adaptive hyperparameter tuning. Empirical results demonstrate that compared with conventional LSTM, attention-enhanced LSTM, and Bayesian-optimized LSTM models, the proposed Multi-Feature Bayesian Optimized Attention-LSTM achieves significantly lower Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), reaching 25.51, 18.35, and 0.66%, respectively. Even during periods of extreme market volatility such as the Russia–Ukraine conflict and the U.S. Federal Reserve’s interest rate hikes in 2022, the MAPE remained below 0.70%. These findings validate the synergistic effect of multi-feature fusion, the attention mechanism, and Bayesian optimization, providing more reliable decision support for financial market participants.

Keywords: LSTM model; attention mechanism; Bayesian optimization; stock price prediction.

1. Introduction

1.1. Research Background and Significance

Stock price prediction has long been a subject of significant research interest, as it not only assists investors in developing rational strategies and mitigating risks but also supports regulatory authorities and financial institutions in policy adjustment. Early prediction models were primarily based on econometric time series approaches, such as Autoregressive (AR), Moving Average (MA), and Autoregressive Integrated Moving Average (ARIMA) models. These models analyze stock price series to identify trends, seasonality, and cyclical patterns, and then establish mathematical models for forecasting. However, such models are usually constrained by strict assumptions that real financial markets rarely satisfy. Moreover, they are sensitive to data quality and volume, and their forecasting capacity is limited under complex and volatile market conditions.

With the rapid advancement of computational technologies, neural networks have been widely adopted in stock price forecasting. Models such as Artificial Neural Networks (ANN), Backpropagation Neural Networks (BP), and Recurrent Neural Networks (RNN) have been explored. However, they suffer from inherent drawbacks, including vulnerability to local optima and inefficiency in handling long sequences. To address these limitations, Hochreiter and Schmidhuber [1] introduced the Long Short-Term Memory (LSTM) algorithm in 1997, which has since been extensively studied and refined. To further enhance predictive performance, this paper combines LSTM with the attention mechanism, and additionally applies Bayesian optimization for hyperparameter tuning to improve model accuracy and stability.

1.2. Literature Review

In stock price forecasting, LSTM is capable of learning long-term dependencies between time series data such as stock prices and trading volumes, thereby enabling more precise modeling of price fluctuations. In 2017, R. Akita [2] proposed the use of LSTM to capture time-series influences on stock prices by leveraging distributed representations of news articles and considering inter-company correlations within industries. The results indicated that distributed textual representations outperform pure numerical and bag-of-words approaches, and that LSTM models capture sequential dependencies more effectively than other methods. Chen et al. [3] (2019) employed stock data of Foxconn from the Taiwan Stock Exchange and introduced new features by averaging the previous five days' opening, high, low, trading volume, and closing prices to predict future stock prices. These predictions were further used to guide investment decisions, supplemented by technical indicators to assess buy, hold, or sell signals, with LSTM models used for validation.

The integration of the attention mechanism with LSTM enables models to focus on critical information and alleviates the limitations of LSTM in handling excessively long sequences. Recent research incorporating attention into recurrent neural networks has demonstrated superior performance in sequence prediction and classification tasks. Chen et al. [4] (2019) compared attention-enhanced LSTM with conventional LSTM in forecasting Hong Kong stock prices, concluding that attention significantly improved predictive accuracy. More recently, Heeseok Kwon et al. [5] (2023) proposed the WMA-LSTM (Wavelet Transform Multi-Head Attention LSTM), which combines wavelet-transformed LSTM with multi-head attention. Their model applied denoising via wavelet transformation and leveraged LSTM with multi-head attention for forecasting stock indices such as KOSPI, S&P 500, and Hang Seng, showing that the combined method improved both stability and predictive performance.

Attention-LSTM applications in financial markets are no longer rare, as researchers increasingly integrate multiple algorithms for higher accuracy. For example, Talabathula Jayanth and A. Manimaran [6] introduced DA-Bi-LSTM-BO (Dual Attention Bi-directional LSTM with Bayesian Optimization), demonstrating superior performance over existing models such as ED-LSTM, ED-Bi-LSTM, and AM-LSTM. Their findings underscored the model's potential to enhance trading decisions through more accurate predictions. Inspired by these advancements, this paper leverages Bayesian optimization to tune the hyperparameters of the Attention-LSTM model, iteratively forecasting stock prices with the aim of delivering improved accuracy and reliability for financial market participants.

2. Methodology

2.1. Principles of the LSTM Model

In machine learning, handling sequential data has long been a major research focus. Traditional neural networks face limitations in modeling sequences with temporal dependencies, which led to the development of Recurrent Neural Networks (RNN). However, RNN are prone to vanishing or exploding gradients, particularly when processing long sequences, resulting in poor performance. The Long Short-Term Memory (LSTM), introduced as a special variant of RNN, effectively addresses these issues.

The core innovation of the LSTM model lies in its memory cell and gating mechanisms—namely, the input gate, forget gate, and output gate. The memory cell enables information transfer across time steps, retaining essential long-term dependencies. The input gate, governed by the sigmoid activation function, determines which new information from the current input and previous hidden state can be added to the cell state. The forget gate, also based on sigmoid activation, outputs values between 0 and 1 to regulate which historical information should be discarded or retained. The output gate similarly applies sigmoid activation to control which portions of the cell state contribute to the hidden state and final output. By selectively remembering and forgetting information at each time step, LSTM effectively mitigates the vanishing/exploding gradient problem and captures long-term dependencies in sequential data. The structure of an LSTM neuron is illustrated in Fig. 1.

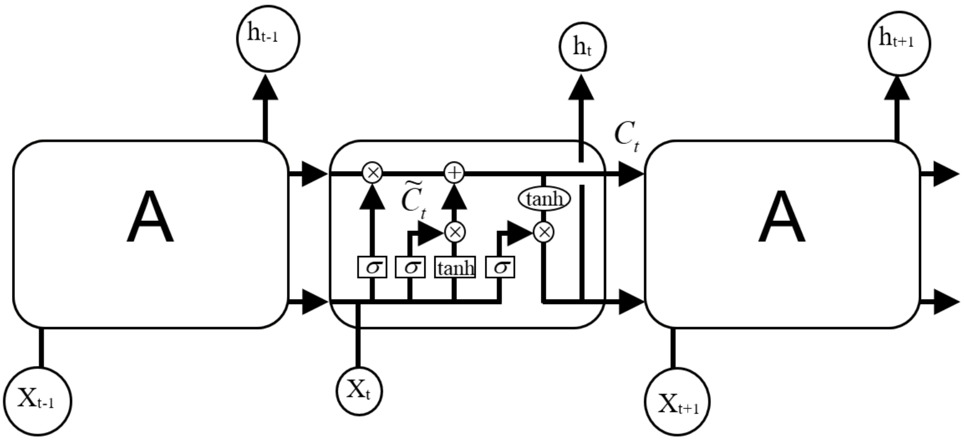


Fig. 1 LSTM Neuron Structure

The computations inside the LSTM neuron proceed as follows:

Input Gate:

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

$$\tilde{C}_t = \tanh(W_c * [h_{t-1}, x_t] + b_i) \tag{2}$$

Here, σ represents the sigmoid activation function; x_t is the input at time t; W_i and b_i denote the weight matrix and threshold vector, respectively.

The value of the input gate can be obtained by (1), and the value in the cell state to be added can be obtained by (2), and whether the value is added to the state at time t depends on the value calculated by (1).

Forget Gate:

$$f_t = \sigma\,(W_f * [h_{t-1}, x_t] + b_f) \tag{3}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad \text{(Cell state)} \tag{4}$$

Whether the memory unit in the neuron preserves the historical state depends on the value calculated by (3).

Output Gate:

$$O_t = \sigma(W_o * [h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t * \tanh(C_t) \tag{6}$$

The output gate decides which parts of the cell state contribute to the hidden state h_t.

Through these mechanisms, LSTM selectively stores and discards information, thereby effectively capturing sequential dependencies.

2.2. Incorporation of the Attention Mechanism

The attention mechanism, inspired by human cognitive processes of selective focus, enhances a model’s ability to concentrate on key information. In deep learning, it modifies the conventional encoder-decoder architecture. The raw input is first encoded into a vector representation. Then, attention calculates similarity scores among input elements, generating attention scores through a learnable weight matrix. These scores are normalized using the softmax function to produce attention weights. Weighted sums of the input vectors yield context vectors focusing on critical features, which are subsequently passed to the decoder or later layers for prediction tasks.

In stock price forecasting, numerous factors influence price fluctuations, making it difficult to identify the most critical ones. The attention mechanism effectively addresses this by focusing on the most influential features. The structure of the Attention-LSTM model is illustrated in Fig. 2.

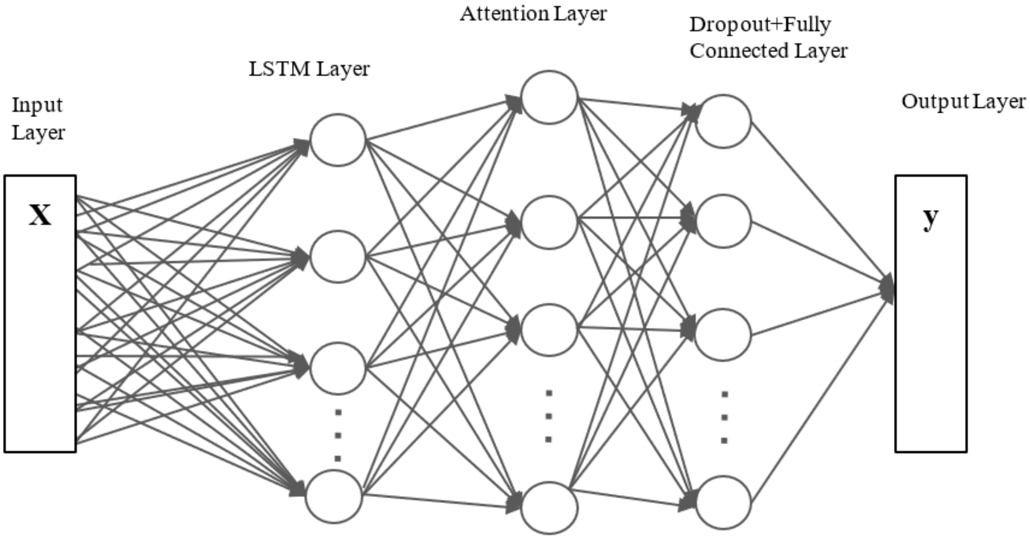


Fig.2 Attention-LSTM Model Structure

The result of LSTM output is encoded by the encoder to obtain a vector representation that can be processed, and then the corresponding weight of the coding vector is calculated by Attention. The obtained weight is weighted to the coding vector and input to the decoder and the result is obtained by the decoder. The following is the calculation process of the Attention layer:

$$e_{t,i} = \upsilon^T \tanh(W_q h_t + W_k h_i) \tag{7}$$

Compute the attention score between the current hidden state h_t and historical hidden states h_i ,where W_q is the query weight matrix, W_k is the key weight matrix, υ is the attention vector.

$$\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_{j=1}^{T} \exp(e_{t,j})} \tag{8}$$

$$c_t = \sum_{i=1}^{T} \alpha_{t,i} h_i \tag{9}$$

The obtained attention score is normalized using the Softmax function to obtain the attention weight. According to this weight, the historical hidden state is weighted and summed to obtain the context vector. After obtaining the output of the Attention layer, the obtained context vector and the hidden state are input into the fully connected layer to calculate the prediction result.

2.3. Bayesian Optimization

Bayesian optimization is an iterative optimization approach that uses probabilistic models to approximate objective functions and search for optimal solutions. Its core idea is to employ Bayesian inference to update the posterior distribution of the objective function at each iteration and select the next candidate point accordingly.

Using Gaussian Processes (GP) as surrogate models, the procedure works as follows.

Model Initialization: A Gaussian Process Regression (GPR) is chosen as the prior. A Gaussian process is a stochastic process where any finite subset follows a joint Gaussian distribution:

$$F(x) \sim GP(m(x), k(x, x')) \tag{10}$$

For a given input space x, the Gaussian process is completely defined by the mean function $m(x)$ and covariance function (kernel function) $k(x, x')$. Where x is the known sample variable, x' is the new sample variable. In general, the squared exponential function is selected as the covariance function:

$$k(x, x') = \sigma_f^2 \exp\left(-\frac{(x - x')^2}{2l^2}\right) \tag{11}$$

Where σ^2 is the signal variance and l the length scale.

Initialization: Randomly sample hyperparameter values in the search space, evaluate them, and form the initial dataset of observations.

Posterior Construction: Update the posterior distribution of the Gaussian process using Bayes' theorem based on observed data, progressively narrowing down regions likely to contain the optimum.

Acquisition Function: Use an acquisition function to balance exploration and exploitation. In this study, Expected Improvement (EI) is adopted:

$$EI(x) \begin{cases} (\mu(x) - f(x^+) - \xi)\Phi(Z) + \sigma(x)\phi(Z) & if \sigma(x) > 0 \\ 0 & if \sigma(x) = 0 \end{cases} \tag{12}$$

$$Z = \frac{\mu(x) - f(x^+) - \xi}{\sigma(x)} \tag{13}$$

Where $f(x^+)$ is the current best observation, Φ and ϕ denote the cumulative distribution function and probability density function of the standard normal distribution, respectively.

Evaluation: Evaluate the candidate hyperparameter configuration on the true objective function.

Iteration: Repeat the process until stopping criteria are met (e.g., maximum iterations, negligible improvement, or full exploration of the hyperparameter space).

For optimizing the Attention-LSTM model, the hyperparameter search space must first be defined. The training, validation, and evaluation process of the model is encapsulated into an objective function, with validation loss or accuracy as the output metric. Bayesian optimization then explores the hyperparameter space using strategies such as handling categorical parameters, early stopping, and parallel optimization. Finally, the best configuration is identified and used to train the final model, which is then tested for performance.

This approach enables efficient identification of optimal hyperparameters, significantly improving the predictive performance of the Attention-LSTM model.

3. Empirical Study

3.1. Experimental Design

This study selects daily data of the S&P 500 Index from 2020 to 2022 as the research sample. The S&P 500, compiled by S&P Dow Jones Indices, is a benchmark index of the U.S. stock market. It comprises 500 large-cap companies listed on the New York Stock Exchange (NYSE) and NASDAQ, representing approximately 80% of U.S. market capitalization. Covering 11 sectors, including information technology, finance, healthcare, and consumer industries, the index is widely regarded as a core indicator of the U.S. economy and capital markets.

Notably, the S&P 500 includes global leaders such as Apple, Microsoft, and Amazon. Its price fluctuations not only reflect domestic economic cycles but also influence global financial markets through channels such as international trade and capital flows. The index exhibits an average correlation of 0.68 with other major indices, such as the Euro Stoxx 50 and the Nikkei 225, underscoring its significance in international asset allocation.

Furthermore, the index demonstrates distinct time series characteristics—trend, volatility, and nonlinearity—that align with the strengths of the proposed Attention-LSTM model. Specifically, LSTM captures long-term dependencies, the attention mechanism highlights critical temporal features, and Bayesian optimization adaptively tunes model parameters for robust performance under complex fluctuations.

The S&P 500's historical dataset is also continuous and standardized, with daily trading records traceable back to the 1950s, minimizing systemic missing data. Its constituent firms are highly transparent in financial disclosures, avoiding the noise typically present in small-cap stocks. For the target period of 2020-2022, the index experienced extreme conditions such as the COVID-19 pandemic, quantitative easing, and monetary tightening, offering a natural environment for robustness testing.

3.2. Data Processing

3.2.1 Dataset Description and Partition

The dataset comprises daily observations of the S&P 500 Index from January 2020 to December 2022, sourced from Investing.com. After excluding holidays and non-trading days, a total of 793 valid records remains. The input features include core trading indicators: opening price (Open), highest price (High), lowest price (Low), closing price (Close), and daily percentage change (Change).

To preserve temporal dependencies and avoid 'future information leakage,' the dataset is split chronologically: Training set (70%): January 1, 2020-December 31, 2021 (529 records), used for model learning. Validation set (15%): January 3, 2022-June 30, 2022 (132 records), used for hyperparameter tuning and early stopping. Test set (15%): July 1, 2022-December 30, 2022 (132 records), used to evaluate generalization ability.

3.2.2 Distribution Characteristics

As shown in Table 1, the dataset displays left-skewed, platykurtic distributions, indicating a degree of homogeneity across features. This aligns with the behavior of financial indices, where extreme declines are more likely, and volatility tends to be dispersed. Such characteristics make the dataset suitable for financial forecasting tasks.

Table 1. Distribution Characteristics of the S&P 500 Dataset

	Close	Open	High	Low
Standard Deviation	548.02	550.45	544.05	554.11
Mean	3859.48	3858.00	3892.00	3821.88
Skewness	-0.48	-0.49	-0.45	-0.51
Kurtosis	-0.49	-0.46	-0.60	-0.37
Maximum	4786.00	4785.25	4808.25	4770.50
Minimum	2220.50	2220.25	2386.00	2174.00

3.2.3 Outlier Detection and Normalization

Outliers were identified using the Z-score method:

$$Z_i = \frac{X_i - \mu}{\sigma} \tag{14}$$

Where X_i is the feature value, μ is the mean, and σ is the standard deviation. The inspection showed no significant anomalies in the dataset. To eliminate the impact of dimensional differences among features, Min–Max normalization was applied to scale all variables into the [0,1] range:

$$X_i^{'} = \frac{X_i - \min(X)}{\max(X) - \min(X)} \tag{15}$$

Finally, the normalized two-dimensional feature data were reshaped into a three-dimensional format (samples, time steps, features) to meet the requirements of recurrent neural networks. This step both enhances training stability and ensures compatibility with model input.

3.3. Evaluation Metrics

To assess predictive accuracy, the following metrics are employed.

Root Mean Squared Error (RMSE), sensitive to large errors, measuring average squared deviations between predicted and actual values.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{16}$$

Mean Absolute Error (MAE), measures the average magnitude of absolute deviations.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{17}$$

Mean Absolute Percentage Error (MAPE), captures average relative error, allowing comparison across scales.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| * 100\% \tag{18}$$

Where y_i is the actual closing price, $\hat{y}_i$ the predicted price, and n the sample size. Smaller values indicate better predictive performance.

3.4. Model Design

In this study, the benchmark model employed is the Attention-LSTM model. The LSTM layer consists of 64 neurons. With the “return” parameter set to True, it outputs the full sequence. The output of the LSTM layer serves as the input for the Attention layer.

Specifically, by computing the attention weights of the hidden states at different time steps and normalizing them via Softmax, a context vector is derived through weighted summation. The Dropout layer plays a crucial role in preventing overfitting. Thus, the Dropout rate is configured to randomly render the output of a portion of neurons ineffective.

The fully - connected layer, containing 32 neurons, acts as a pivotal link. It further integrates, filters, and non - linearly transforms the long - term dependencies of the time series and the key information weights captured by the preceding LSTM layer and Attention layer. By doing so, it combines the dispersed feature information into more representative comprehensive features, thereby offering more effective input for the ultimate prediction.

The Relu activation function is utilized to endow the model with the ability to learn the intricate interaction relationships among features. This allows the model to transcend the limitations of linear models and more precisely capture the underlying patterns of stock price fluctuations.

Finally, the output layer contains a single neuron, and the output result is set as the opening price of the next trading day.

Bayesian optimization is used to optimize the hyperparameters of this model, and the posterior distribution of the objective function is constructed by Gaussian process to explore the hyperparameter space efficiently. Table 2 shows the key hyperparameters that need to be optimized:

Table 2. Hyperparameters and Ranges

Hyperparameter	Rationale	Range
Learning rate	Controls parameter update speed	[1e^{-5}, 1e^{-2}] (uniform)
Dropout rate	Balances fitting and generalization	[0.15, 0.45]
Neurons	Determines feature extraction ability	[32, 128]

3.5. Model Training and Parameter Optimization

All models were trained with the Adam optimizer, Mean Squared Error (MSE) loss, 50 epochs, and a batch size of 64. Early stopping was employed to prevent overfitting, halting training if validation loss failed to improve for five consecutive epochs. After 30 iterations of Bayesian optimization, the best hyperparameter combination was obtained, as shown in Table 3.

Table 3. Optimal Hyperparameter Combination

Learning Rate	Dropout Rate	Neurons
0.001	0.20	64

The optimized model exhibited steady declines in both training and validation loss, with minimal gap between the two, indicating effective fitting without overfitting.

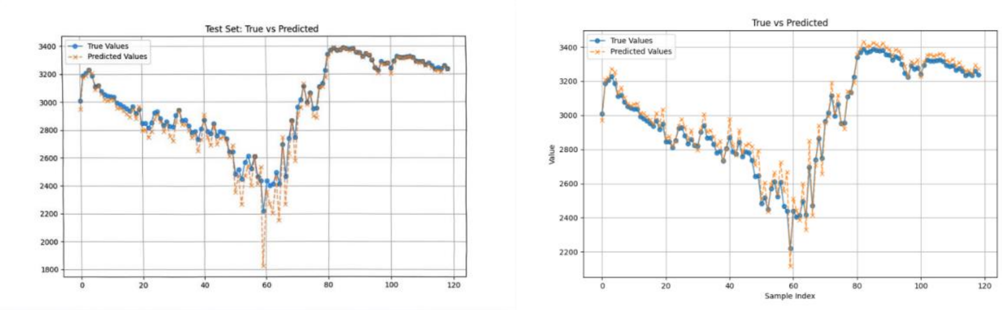
3.6. Comparative Experiment and Analysis

To validate the synergistic effect of multi-feature fusion, attention mechanism, and Bayesian optimization, comparative experiments were conducted across different models. Results show that applying attention or Bayesian optimization alone yielded limited improvements, whereas their combination produced significant gains.

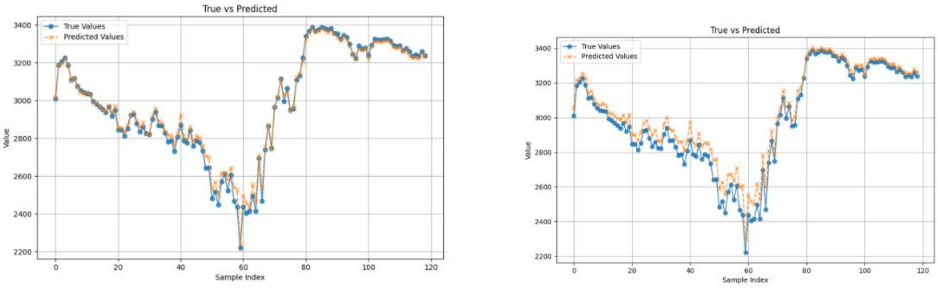
As shown in Table 4 and Fig. 3, the Attention mechanism improved the extraction of critical temporal features, while Bayesian optimization precisely tuned model parameters. Combined with multi-feature fusion, the BO-Attention-LSTM achieved superior performance in all metrics. Even during turbulent periods such as the Russia–Ukraine conflict (March 2022) and Federal Reserve rate hikes (June 2022), the model’s MAPE remained below 0.70%, outperforming the baseline LSTM’s 1.53%.

Table 4. Performance Comparison Across Models

Model	RMSE	MAE	MAPE (%)
LSTM	69.63	40.88	1.53
Attention-LSTM	52.24	40.33	1.41
BO-LSTM	59.44	49.08	1.77
BO-Attention-LSTM	25.51	18.35	0.66



LSTM Attention-LSTM



BO-LSTM BO-Attention-LSTM

Fig. 3 Fit image comparison

4. Conclusion and Future Work

This study addresses the problem of stock price prediction by overcoming the limitations of traditional models in complex financial markets. We constructed a hybrid model that integrates multi-feature fusion, an Attention-LSTM architecture, and Bayesian optimization.

First, multiple core trading indicators of stock prices were fused to provide the model with comprehensive input features. Second, LSTM was employed to capture long-term dependencies in time series, while the attention mechanism enhanced the model’s ability to focus on critical temporal features. Finally, Bayesian optimization was applied to efficiently search for optimal hyperparameter configurations, further improving model performance.

Using daily data from the S&P 500 Index spanning 2020-2022, we conducted preprocessing, training, and comparative experiments. Empirical results show that the BO-Attention-LSTM model consistently outperformed the baseline LSTM, Attention-LSTM, and BO-LSTM models. Specifically, RMSE, MAE, and MAPE were reduced by 63.36%, 55.11%, and 56.86%, respectively, compared with the baseline LSTM. Importantly, even under extreme market volatility—such as the COVID-19 pandemic, Russia–Ukraine conflict, and U.S. Federal Reserve rate hikes—the model maintained low error rates, underscoring the effectiveness of multi-technology integration in improving both prediction accuracy and robustness. These findings demonstrate the practical value of the proposed model in real-world financial scenarios.

Nevertheless, there remains room for further improvement. Currently, the model primarily relies on trading indicators. Future research could incorporate macroeconomic data, sentiment from financial news, and industry-related information to enrich feature dimensions and capture a broader range of market drivers. Moreover, more sophisticated attention mechanisms—such as multi-head attention or cross-modal attention—could be explored in combination with LSTM to enhance performance. Transfer learning may also be introduced to improve adaptability in small-sample contexts or emerging markets.

Additionally, the model’s generalizability can be further validated by applying it to different stock indices or longer time horizons. Through continuous optimization of model structure and expansion of application scenarios, this line of research holds promise for providing more accurate and flexible technical support for financial market forecasting.

References

[1] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 15 Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[2] R. Akita, A. Yoshihara, T. Matsubara and K. Uehara, "Deep learning for stock prediction using numerical and textual information," 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), Okayama, Japan, 2016, pp. 1-6, doi: 10.1109/ICIS.2016.7550882.

[3] C. Y. Lai, R. -C. Chen and R. E. Caraka, "Prediction Stock Price Based on Different Index Factors Using LSTM," 2019 International Conference on Machine Learning and Cybernetics (ICMLC), Kobe, Japan, 2019, pp.1-6, doi: 10.1109/ICMLC48188.2019.8949162.

[4] Chen S, Ge L. Exploring the attention mechanism in LSTM-based Hong Kong stock price movement prediction[J].Quantitative Finance,2019,19(9):1507-1515.

[5] Heeseok Kwon, & Minhyuk Lee (2023). Stock Index Forecasting Using Combined Model of Wavelet Transform LSTM and Multi-Head Attention. KOREAN MANAGEMENT SCIENCE REVIEW, 40(2), 97-112. 10.7737/KMSR.2023.40.2.097

[6] Jayanth T, Manimaran A. Unlocking Stock Price Prognostication Dual Attention-Infused Bi-Directional LSTM Empowered by Bayesian Optimization DA-Bi-LSTM-BO[J]. SN Computer Science, 2024, 5(8):1046-1046.

[7] Lin Xin, Zhu Xiaodong. LSTM stock price prediction model based on the attention mechanism [J]. Journal of Chongqing Technology and Business University (Natural Science Edition), 2022, 39(02): 75-82.DOI:10.16055/j.issn.1672-058X.2022.0002.011.

[8] Zi-sheng O, Xi-te Y, Yongzeng L. Systemic financial risk early warning of financial market in China using Attention-LSTM model[J].North American Journal of Economics and Finance,2021,56

[9] Youfeng N ,Mingxi G ,Wenhao Y , et al.A Bayesian optimization-based LSTM model for DGA domain name identification approach[J].Journal of Physics: Conference Series,2022,2303(1).

[10] Sang, S., & Li, L. (2024). A Novel Variant of LSTM Stock Prediction Method Incorporating Attention Mechanism. Mathematics, 12(7), 945. https://doi.org/10.3390/math12070945

[11] Benchaji, I., Douzi, S., El Ouahidi, B. et al. Enhanced credit card fraud detection based on attention mechanism and LSTM deep model. J Big Data 8, 151 (2021).

[12] Z. Zhang et al., "HGLA: Biomolecular Interaction Prediction Based on Mixed High-Order Graph Convolution With Filter Network via LSTM and Channel Attention," in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 21, no. 6, pp. 2011-2024, Nov.-Dec. 2024, doi: 10.1109/TCBB.2024.3434399.

[13] Liwei T, Li F, et al. Forecast of LSTM-XGBoost in Stock Price Based on Bayesian Optimization[J]. Intelligent Automation & Soft Computing, 2021,29(3):855-868.