Original Research Article

Research on stock price prediction based on LSTM model integrated with SHAP interpretation: A case study of CSI 300 index

Chenglong Cao*

School of Statistics and Applied Mathematics, Anhui University of Finance and Economics, Bengbu, Anhui, 233030, China

Abstract: Aiming at the nonlinear noise feature and poor interpretability of financial time series prediction, this paper constructs an interpretable LSTM prediction framework combined with SHAP, taking CSI 300 daily data from 2015 to 2025 as samples. The dataset is divided at 70%:10%:20%, and comparative tests verify that the 5-day time window has optimal prediction accuracy. The model can fit medium-long term index trends but shows obvious lag under drastic market fluctuations. SHAP analysis indicates that opening, high, low and closing prices are core predictive features, while trading and volatility indicators play auxiliary roles. This research balances prediction precision and model transparency, providing quantitative decision support for index investment and risk control.

Keywords: stock price prediction; LSTM model; SHAP interpretation; CSI 300 index

1. Introduction

With the rapid development of financial markets, accurate forecasting of stock price trends helps institutional investors optimize investment strategies and provides reliable guidance for rational decision-making among individual investors. Consequently, research on stock price prediction has attracted extensive academic attention[1]. Traditional econometric models such as ARMA[2] and GARCH[3] characterize financial time series based on predefined functional forms. However, real-world noisy and complex time series cannot be fully described by parametric analytical equations, which makes machine learning a mainstream research method nowadays. Shallow machine learning algorithms including SVM[4] and BP neural networks[5] suffer from obvious limitations when modeling high-dimensional complex data, accompanied by the curse of dimensionality and inefficient feature representation[6]. For this reason, researchers turn to deep learning models with stronger generalization performance.Singh and Srivastava (2017)[7] adopted deep neural networks for technical analysis on Google stock and achieved accurate daily price forecasting. Yadav and Vishwakarma (2024)[8] constructed a deep learning model to predict stock prices in the final trading hour based on six-hour historical market data. Lin et al. (2021)[9] applied deep learning to forecast prices of the S&P 500 Index and CSI 300 Index. Recurrent Neural Networks (RNNs)[10] are widely applied to sequential data prediction, yet they suffer from long-term information forgetting. The Long Short-Term Memory (LSTM)[11] network was proposed to overcome this defect, making it more suitable for stock price prediction. Nevertheless, LSTM is a black-box deep sequential neural network, which cannot explicitly quantify which factors drive price fluctuations and their respective contribution values. SHAP, a popular additive interpretability framework[12], can be embedded into the LSTM forecasting framework. This combination effectively addresses the black-box problem and quantifies the marginal contribution of each feature to individual prediction outcomes.

2. Theoretical basis

2.1. LSTM model

As a variant of RNN, the LSTM model introduces controlled self-loops to effectively alleviate the vanishing gradient problem caused by increased network depth and long time lags in RNN. Thanks to its sophisticated structural design, LSTM is especially well-suited for tasks with long delays and time intervals. The core computation of a single LSTM cell works as follows: it takes the input features at the current moment and the memory state from the previous moment. Relying on the three-gate structure, it performs stepwise calculations to filter historical information, update new information and generate the current output. Finally, it outputs the hidden state of the current time step, updates the cell state, and passes the information to the next cell. The detailed procedures are presented below.

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

$$i_t = \sigma\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right) \tag{2}$$

$$\tilde{C}_t = \tanh\left(W_c \cdot \left[h_{t-1}, x_t\right] + b_c\right) \tag{3}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{4}$$

$$o_t = \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right) \tag{5}$$

$$h_t = o_t \odot \tanh\left(C_t\right) \tag{6}$$

Where h_{t-1} and x_t denote the inputs; W_f, W_i, W_c, W_o are weight matrices; b_f, b_i, b_c, b_o are bias terms; f_i, i_t and o_t trepresent the outputs of the gating mechanisms; C_t is the output of the memory cell at time step t; $\tilde{C}_t$ is the input to the memory cell at the current time step; and tanh and σ are activation functions.

2.2. SHAP additive explanations

Step 1: Sampling Interpolation Points.

The implementation process of SHAP additive explanations proceeds as follows: first, a series of intermediate interpolation points are uniformly sampled along the straight path between the baseline sample and the test sample to prepare for subsequent gradient calculation and integration operations. Next, partial derivatives of the model output with respect to each input feature are computed for all interpolation points, so as to quantify the degree of influence of minor changes in a single feature on the model's prediction results. Then, the integral is numerically approximated by multiplying the average gradient along the path by the feature variation, and the gradients at all interpolation points are summed to obtain the total contribution of each feature to the prediction of the test sample under a single baseline, which corresponds to the result of Integrated Gradients. Finally, the Integrated Gradients results for all background samples are averaged to eliminate the bias introduced by a single baseline, yielding the final feature attributions that approximately satisfy the SHAP axioms.

$$x_\alpha = x' + \alpha\left(x - x'\right), \quad \alpha \in \left\{0, \frac{1}{K}, \frac{2}{K}, \ldots, 1\right\} \tag{7}$$

$$\nabla f(x_\alpha) = \left(\frac{\partial f(x_\alpha)}{\partial x_1}, \frac{\partial f(x_\alpha)}{\partial x_2}, \ldots, \frac{\partial f(x_\alpha)}{\partial x_d}\right) \tag{8}$$

$$IG_i(x, x') \approx \left(x_i - x_i'\right) \times \frac{1}{K} \sum_{k=1}^{K} \frac{\partial f\left(x' + \frac{k}{K}\left(x - x'\right)\right)}{\partial x_i} \tag{9}$$

$$\phi_i(x) = \frac{1}{M} \sum_{m=1}^{M} IG_i\left(x, x_m'\right) \tag{10}$$

Where x is the test sample,x' is the baseline sample, α is the scaling coefficient ranging from 0 to 1, and x_α is the interpolation point along the path.Where $\nabla f(x_\alpha)$ is the gradient of the model output at the interpolation point.Where $IG(x,x')$is the total contribution of feature i to the test sample prediction under a single baseline, and $(x_i\text{-}x_i')$ is the variation of feature i from the baseline to the input.where ϕ_i (x) is the final SHAP value of feature i, M denotes the total number of background samples, and $IG_i(x,x_m')$ stands for the Integrated Gradients attribution of feature i under the m-th background sample.

3. Empirical analysis

3.1. Data source and parameter configuration

This study selects daily trading data of the CSI 300 Index from January 1, 2015 to December 31, 2025, including closing price, opening price, highest price, lowest price, trading volume, trading turnover, price change percentage, amplitude, average price and turnover rate. All data are sourced from the Wind Database. The dataset is split into a training set, a validation set and a test set at a ratio of 70%, 10% and 20%. This study adopts the LSTM neural network model to extract deep feature representations that drive the fluctuations of the stock index, so as to predict the closing price of the CSI 300 Index. Detailed parameter configurations are shown in **Table 1.**

Table 1. Parameter settings.

Table column subhead	Subhead	Subhead
Parameter	AdamW	Adaptive optimizer with weight decay
Initial learning rate	0.001	Step size for parameter updates
Learning rate scheduler	ReduceLROnPlateau	Decay the learning rate based on validation loss
Maximum epochs	200	Upper limit of training iterations
Early stopping	15	Terminate training if no optimal loss is found on the validation set for 15 consecutive epochs
Loss function	MSE	Mean squared error loss for regression prediction
Dropout	0.2	Randomly deactivate neurons to prevent overfitting
batch_size	32	Batch size

3.2. Interpretable model prediction

To explore how window length affects forecasting outcomes, this paper conducts closing price predictions with window lengths of 1, 3, 5 and 7 under identical parameter settings. We compare model performance using MSE, RMSE and MAPE metrics.

Table 2. Comparison of different window lengths.

Window Length	MSE	RMSE	MAPE
3	45.52	64.21	1.17%
5	42.11	62.35	1.09%
7	47.37	66.78	1.31%

As shown in **Table 2**, the time window length significantly impacts LSTM prediction accuracy, and the 5-day window is chosen for the model because it strikes an optimal balance between effective market information and noise while outperforming the 3-day and 7-day windows with insufficient historical data and redundant information respectively.

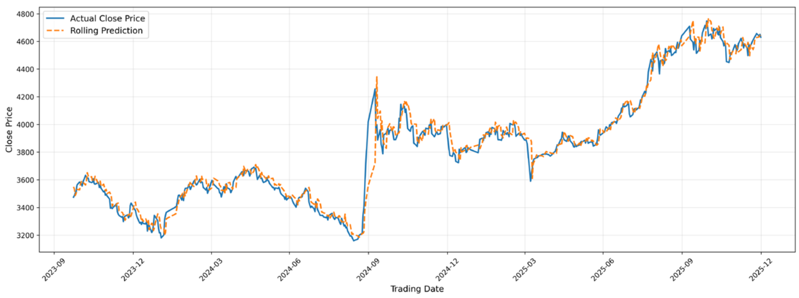

Figure 1. Rolling forecast results of the LSTM model.

As illustrated by the rolling forecast in **Figure 1**, the model effectively captures the general trend of the CSI 300 Index close, with predictions closely matching actual prices and reliably tracking market rises, falls and consolidations, especially under stable market conditions. Yet obvious lag and bias emerge amid sharp spikes, crashes and high-level turbulence, such as the market swings in September 2024, March 2025 and late 2025. To sum up, the model boasts excellent in-sample fitting and can reflect the index's long-term price trend, but needs further optimization to capture short-term turning points and extreme volatility.

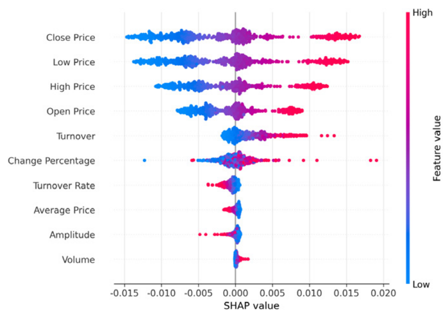

Figure 2. SHAP explanations for LSTM forecasts.

The SHAP approach is adopted to unpack the LSTM forecasting mechanism for CSI 300 closing prices and quantify each variable's importance and impact direction. Red indicating high feature levels and blue indicating low levels. The importance ranking is closing price, lowest price, highest price, opening price, trading turnover, price change, turnover rate, average price, amplitude and trading volume, suggesting that price indicators dominate the prediction, while trading and volatility indicators play secondary roles. As shown in **Figure 2**, the leading closing price has a clear positive correlation with forecasts; the second-ranked lowest price drags predictions down, while the third-ranked highest price pushes them up, indicating that the model prioritizes downside support. The opening price provides limited additional information, and these four price variables form the core forecasting logic. Trading turnover brings asymmetric upward pressure under sufficient liquidity; price change mainly matters during strong market swings; high turnover rate reflects trading divergence and lowers expected prices; wider intraday amplitude increases uncertainty and depresses forecasts, while trading volume loses independent explanatory power after controlling for turnover. Overall, SHAP quantifies feature contributions and alleviates the black-box nature of the LSTM network. The results confirm a prediction pattern led by price levels, supplemented by capital activity and adjusted by volatility risks. This feature structure matches the model's strong trend-fitting capacity: its reliance on price series enables close tracking of the CSI 300 Index, while the low weights of variables related to abrupt reversals and extreme volatility explain the forecast lag during sharp surges, crashes and trend turning points.

4. Conclusions

Taking the CSI 300 Index as the sample, this paper constructs an interpretable LSTM prediction model embedded with the SHAP framework to tackle the nonlinear noisy financial time series and the black-box problem of deep learning models. Using daily data from 2015 to 2025, this study evaluates model performance and feature influence. Three main conclusions are drawn. First, the LSTM model works well for short-term index prediction, and the 5-day time window achieves the best performance with MSE of 42.11, RMSE of 62.35 and MAPE of 1.09%. The model can capture regular market trends but lags notably during sharp price swings and abrupt reversals. Second, SHAP results uncover the model's decision-making mechanism: price indicators including closing, lowest, highest and opening prices serve as core predictors, with trading turnover providing supplementary positive information, while turnover rate and volatility suppress forecasts amid high uncertainty. This feature structure explains the prediction lag under extreme market shocks. Third, this study offers a quantitative interpretability method for financial deep learning and supplies investment references for index trading and risk management.

About the author

*Corresponding author：Chenglong Cao (Email: 2210918411@qq.com).

References

[1] Agrawal J G,Chourasia V,Mittra A.State-of-the-artinstockpredictiontechniques[J]. International Journal of Advanced Researchin Electrical,Electronics and Instrumentation Engineering,2013,2(4):1360-1366.

[2] Box G,Jenkins G M.Analysis:Forecasting and control[J].Sanfrancisco,1976,10.

[3] Bollerslev T. Generalized autoregressive conditional heteroskedasticity[J].Journal of econometrics, 1986,31(3):307-327.

[4] Cortes C,Vapnik V.Support-vector networks[J].Machine learning,1995,20(3):273-297.

[5] Rumelhart D E,Hinton G E,Williams R J.Learning internal representations by error propagation [R].1985.

[6] Bengio Y,Le Cun Y. Scaling learning algorithms towards AI[J].Large-scale kerne lmachines, 2007, 34(5):1-41.

[7] Singh R,Srivastava S. Stock prediction using deep learning[J].Multimedia Tools and Applicatio ns,2017,76(18):18569-18584.

[8] Yadav A K,Vishwakarma V P. An integrated blockchain based real time stock price prediction model by CNN,BiLSTM and AM[J].Procedia Computer Science,2024,235:2630-2640.

[9] Lin Y,Yan Y,Xu J,et al. Forecasting stock index price using the CEEMDAN-LSTM model[J].The North American Journal of Economics and Finance,2021,57:101421.

[10] Elman J L. Finding structure in time[J].Cognitive science,1990,14(2):179-211.

[11] Hochreiter S,Schmidhuber J. Longshort-termmemory[J].Neural computation,1997,9(8):1735-1780.

[12] Lundberg S M,Lee S I.A unified approach to interpreting model predictions[J].Advances in neural information processing systems,2017,30.