Proceedings of the 4th International Conference on Management Research and Economic Development
DOI: 10.54254/2754-1169/2026.35422

Bayesian-Optimized LSTM with Sentiment-Augmented Technical Indicators for Stock Return Prediction: Evidence from CSI 300

Bairui Peng

Warwick Business School, University of Warwick, Coventry, UK
pengbr03@outlook.com

Abstract. Accurately predicting stock returns remains one of the core challenges in quantitative finance, exacerbated by nonlinear market dynamics, irrational investor behavior, and the stochasticity of financial time series. This paper proposes a novel hybrid framework —Bayesian Optimized Long Short-Term Memory Network Based on Sentiment Enhancement Technical Indicators (BOLSTM-SATI), which combines a Long Short-Term Memory (LSTM) neural network with classic technical analysis signals and multi-source investor sentiment indices from the CSI 300 index constituents. This study constructs a composite feature space including MACD, RSI, Bollinger Bands, volume balance indicator, and Average True Range (ATR). Investor sentiment is quantified through a Bayesian update mechanism that integrates news text sentiment scores obtained through FinBERT and social media sentiment indices from the Eastmoney Guba forum. To mitigate overfitting and improve generalization ability, we employ a Bayesian optimization method based on a tree-structured Parzen estimator (TPE) for LSTM hyperparameter search. Robustness to perturbations is verified through Gaussian noise injection experiments. Empirical evaluation using daily data from January 2015 to December 2022 shows that BOLSTM-SATI achieves a directional accuracy of 67.3%, a mean absolute error (MAE) of 0.0082, and a Sharpe ratio of 2.41 in backtesting simulations, outperforming baseline models including ARIMA, traditional LSTM, XGBoost, and Transformer variants. The results confirm that incorporating sentiment dynamics into technical feature learning can significantly improve prediction accuracy and the profitability of trading strategies.

Keywords: Stock Return Prediction, LSTM, MACD, Bayesian Optimization, FinBERT

1. Introduction

The components of the complex adaptive system that cause changes in asset prices generally fall into the categories of macro-economic, technological and behavioural factors. Although the Efficient Market Hypothesis (EMH) assumes that prices fully reflect all available information, research has shown that due to information asymmetry, bounded rationality and herding behaviour, abnormal profits in the stock market may still exist [1, 2]. Therefore, research has focused on forecasting the risk-adjusted returns of stocks in quantitative finance.

DOI: 10.54254/2754-1169/2026.35422

Many deep-learning models have recently been used for the prediction of financial time series. Long Short-Term Memory (LSTM) networks [3] are good at capturing non-linear temporal dependencies through their gating mechanisms and can be applied to financial data with long-term memory. However, data-driven approaches alone often ignore the domain knowledge contained in technical indicators. Although the three are generally believed to offer some predictive power, their non-linear relationships are not easily captured by the old model for statistics [4, 5].

The other reason for a change in asset prices may be reduced investor confidence by individual and institution investors in certain markets. Progress in natural language processing has led to Transformer-based models such as FinBERT, and these can now be used to extract sentiment signals from financial news and social media to provide additional explanations for changes in stock prices [6-8].

Optimise the hyperparameters of the deep learning forecasting model as well. The Tree-structured Parzen Estimator (TPE) algorithm [9] of Bayesian optimisation is a typical method to reduce the search space and perform exploration-exploitation trade-off in large-scale problems.

Based on the above analysis, this paper presents the BOLSTM-SATI framework that incorporates technical indicators and a Bayesian-updated sentiment index into a TPE-optimized LSTM model. Robustness will also be checked in a perturbation experiment on a shifted distribution. In the past eight years, the CSI 300 index has not shown any favourable response to the enhancements in prediction and backtesting efficiency of this method according to the experiment.

2. Related work

2.1. Application of machine learning in stock return prediction

Long before the development of deep learning, many methods have been used in financial forecasting based on machine learning [10]. Recurrent Neural Networks (RNNs) and their LSTM variations are now popular models for handling time-series data in finance due to their capacity to model sequential dependencies [11]. Fischer and Krauss [12] have shown that LSTM networks perform better than random forests, deep neural networks and logistic regression in predicting the next-day directional returns of S&P 500 constituent stocks. Ding et al. [13] have later proposed event-driven neural networks in their research to combine the structured representation of news events with price data.

Gradient boosting algorithms are another set of ensemble methods that have also performed well in financial forecasting (e.g., XGBoost and LightGBM) [14]. These models are good at learning non-linear feature interactions and are robust to outliers, so they are suitable for financial data with a heavy-tailed return distribution. In recent years, Transformer-based architectures [15] have been used for financial forecasting to address the long-range dependency problem of traditional RNNs via a self-attention mechanism.

2.2. Technical analysis integration

Technical indicators can be employed to show the distribution of a market and identify various trends and momenta of the data. Gerald Appel developed the MACD index to show the difference between short-term and long-term exponential moving averages (EMAs) of a price series and thus identify changes in price momentum and trend direction. Based on the above empirical research, the future trend of price changes for both developed and developing economies can be predicted using the MACD indicator [16]. Wilder proposed the RSI index [17] to measure how rapidly and how

DOI: 10.54254/2754-1169/2026.35422

significantly the price has changed, and thereby determine whether it is in an overbought or oversold state. Bollinger Bands are adaptive channels around a moving average to show changes in volatility [18].

Recently, many studies have used multiple indicators and machine learning for joint analysis. Sezer et al. [19] put forward a machine learning-based Bollinger Bands model that uses volatility normalization channels to predict market volatility and thus provides dynamic support and resistance levels [19]. conducted an all-encompassing review of deep learning applications in technical indicators and pointed out that hybrid models combining technical features with deep architectures performed better than either method individually. 2.3 According to the research in behavioural finance, as shown by papers [20, 21], investor sentiment also affects asset prices in an organised manner, particularly in the short term. Baker and Wurgler [22] constructed an all-weather general sentiment index by taking some market proxy indicators and found that it could predict cross-sectional return anomalies. As social media and online financial communities have spread, new sources of text data for sentiment analysis have also appeared. Chen and others [23] have shown that the information in posts on financial social media is related to changes in stock prices and other economic indices. Advances in Natural Language Processing have improved the accuracy of sentiment extraction. FinBERT [8] is a domain-specific BERT model that has been pre-trained on a large volume of financial data to achieve good results in sentiment analysis for finance. It has been shown in terms of the building of trading strategies that it outperforms dictionary-based sentiment analysis methods significantly [24].

2.3. Bayesian optimization for hyperparameter tuning

Optimisation of hyperparameters is needed to improve the performance of LSTMs, but it is computationally expensive in a large search space. Bayesian optimisation builds a probabilistic surrogate model of the objective function (such as a Gaussian process (GP) or a tree-based Parzen estimator (TPE)) and selects candidate parameters for evaluation by maximising the acquisition function [9]. Bayesian optimisation performs as well as or better than grid search and random search with significantly fewer function evaluations [25]. Application of financial forecasting models shows that Bayesian-optimised LSTM architectures significantly outperform equivalent architectures with default parameterisation [26].

3. Data and feature engineering

3.1. Data sources and samples

Constituent stock empirical analysis is carried out in this paper for the CSI 300 Index from January 2015 to December 2022, with a total of about 1,952 trading days. CSI 300 Index (000300.SH) is the prediction target, and daily OHLCV data will be obtained from the Wind Financial Terminal.

To maintain the quality of the data, stocks with less than 400 trading days have been excluded; thus, a survival-adjusted panel of 241 stocks was obtained. The data set is divided into training (2015-2019), validation (2020) and testing (2021-2022) sets chronologically.

Around 1.2 million financial news articles were collected via the Tonghuashun Financial News API, and more than 4.7 million posts were obtained from Guba Forum, a large community for Chinese retail investors. Remove noise from all the texts, such as HTML tags, special characters and spam content.

DOI: 10.54254/2754-1169/2026.35422

3.2. Technical indicator construction

We calculated the following technical indicators based on daily OHLCV data(see table 1): All technical indicator values are standardized using z-score normalization computed over rolling 252-day windows to prevent look-ahead bias. Missing values arising from indicator warm-up periods are forward-filled using the first valid observation.

Table 1. Technical indicators incorporated in the feature space

Indicator	Formula / Parameters	Economic Interpretation
MACD Line	EMA(12) − EMA(26)	Trend momentum divergence
Signal Line	EMA(9) of MACD Line	MACD smoothing, crossover signal
MACD Histogram	MACD Line − Signal Line	Momentum acceleration
RSI(14)	100 − 100/(1 + RS)	Overbought/oversold momentum
Bollinger Upper Band	SMA(20) + 2σ(20)	Dynamic resistance level
Bollinger Lower Band	SMA(20) − 2σ(20)	Dynamic support level
BB%B	(Close − Lower)/(Upper − Lower)	Position within Bollinger channel
ATR(14)	Max(H−L, |H−C_{-1}|, |L−C_{-1}|)	Realized volatility proxy
OBV	Σ sign(ΔClose)×Volume	Cumulative volume-price pressure
Williams %R(14)	−100×(H_{14}−C)/(H_{14}−L_{14})	Momentum oscillator

3.3. Investor sentiment index construction

To incorporate investor sentiment into the forecasting framework, we construct a multi-source sentiment index BtB_tBt based on financial news and social media data. The construction follows a three-stage process.

First, each document is processed using FinBERT, which classifies text into positive, neutral, and negative categories. A continuous sentiment score is derived from the predicted probabilities:

$$s_i = p_i^+ - p_i^-, \text{where} s_i \in [-1,1] \tag{1}$$

Daily sentiment indices for news and social media are then computed as volume-weighted averages:

$$S_t^n ews = (\Sigma(s_i \times w_i))/(\Sigma w_i) \tag{2}$$

$$S_t^s ocial = (\Sigma(s_j \times w_j))/(\Sigma w_j) \tag{3}$$

where w_i and w_j denote document-level weights (e.g., engagement or frequency-based weights).

Second, we apply a recursive Bayesian updating mechanism to integrate temporal information. The previous day's sentiment serves as the prior, while new observations form the likelihood update. The unified sentiment index is defined as:

$$B_t = w_n ews \times S_t^n ews + w_s ocial \times S_t^s ocial \tag{4}$$

where w_news = 0.6 and w_social = 0.4 are determined via validation performance.

Third, we construct additional sentiment-related features to capture higher-order dynamics. Sentiment momentum is defined as:

$$\Delta B_t = B_t - B_{t-5} \quad (5)$$

Sentiment dispersion is computed as:

$$\sigma_t^s = sqrt((1/N_t) \times \Sigma(s_i - S_t)^2) \quad (6)$$

Finally, we define a sentiment–trading interaction term to capture joint effects between emotional intensity and market activity

$$I_t = B_t \times OBV_t \quad (7)$$

4. Model architecture and Bayesian optimization

4.1. LSTM architecture

The BOLSTM-SATI (see figure 1)model employs a stacked LSTM architecture. Let x_t ∈ ℝ^d denote the feature vector at time t, comprising d = 16 technical and sentiment features. The input sequence is X = {x_{t-T+1}, ..., x_t} with lookback window T. The LSTM update equations at each time step are:

$$f\_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad \text{(Forget gate)} \quad (8)$$

$$i\_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad \text{(Input gate)} \quad (9)$$

$$\widetilde{C}\_t = \tanh(W \cdot C \cdot [h_{t-1}, x_t] + b \cdot C) \quad \text{(Candidate cell state)} \quad (10)$$

$$C\_t = f\_t \odot C_{t-1} + i\_t \odot \widetilde{C}_t \quad \text{(Cell state update)} \quad (11)$$

$$h\_t = o\_t \odot \tanh(C\_t) \quad \text{(Hidden state output)} \quad (12)$$

The output h_T from the final time step is passed through two fully connected layers with Re LU activation, followed by a linear output layer predicting the next-day log return r_{t+1} = log(P_{t+1}/P_t). Training employs the Adam optimizer with gradient clipping (max norm = 1.0) to prevent exploding gradients. Dropout regularization is applied to all LSTM layers and the penultimate dense layer.

4.2. Bayesian hyperparameter optimization

We use Bayesian optimisation based on the Tree-structured Parzen Estimator (TPE) to tune model hyperparameters in a data-efficient way. The key architectural and training parameters in the search space are a lookback window size, the number of LSTM layers, hidden units, dropout rate, learning rate, batch size and L2 regularization strength.

The two density estimators of the TPE model are l(x) for good-performing configurations and g(x) for the other samples, respectively. New hyperparameter candidates are generated by maximising the ratio l(x)/g(x) and serve as the acquisition criterion.

DOI: 10.54254/2754-1169/2026.35422

Perform 100 rounds of optimisation and time-series cross-validation on the training set. Mean Squared Error (MSE) on the validation folds serves as the objective function for optimisation. HyperOpt is used for the actual implementation.

4.3. Perturbation robustness analysis

Gaussian perturbation analysis is used to investigate the impact of noise and non-stationarity in the market on the stability of the model. Specifically, independent noise $\epsilon \sim N(0,\sigma 2)$ is added to each feature dimension of the test inputs, where $\sigma \in \{0.01,0.05,0.10,0.20\}$. Simulate possible data quality problems, such as market microstructure noise, delayed feeds and distributional shifts in the training and deployment environments. Robustness of the model is shown by the reduction in prediction accuracy with increasing noise.

Conduct an ablation study to determine how much the modified elements are changed. Remove the sentiment-related features and use a simple average function instead of Bayesian fusion to isolate the incremental effect of the proposed sentiment integration.

5. Results

As shown in table 1, BOLSTM-SATI demonstrates reasonable robustness to small perturbations (σ = 0.01–0.05), with directional accuracy declining by only 0.3 to 1.5 percentage points. At higher noise levels (σ = 0.10–0.20), performance degrades more substantially, suggesting sensitivity to extreme distributional shift. Critically, directional accuracy remains above 59% even at σ = 0.20, substantially above the 50% random baseline, indicating residual predictive power under severe input corruption.

Table 2. Perturbation robustness analysis under Gaussian input noise injection on the test set

Noise Level σ	MAE	Dir. Accuracy (%)	Δ MAE vs. Clean	Δ Acc. vs. Clean
0 (clean)	0.0082	67.3	—	—
0.01	0.0084	67.0	+2.4%	−0.3pp
0.05	0.0091	65.8	+10.9%	−1.5pp
0.10	0.0103	63.4	+25.6%	−3.9pp
0.20	0.0129	59.7	+57.3%	−7.6pp

BOLSTM-SATI demonstrates reasonable robustness to small perturbations (σ = 0.01–0.05), with directional accuracy declining by only 0.3 to 1.5 percentage points. At higher noise levels (σ = 0.10–0.20), performance degrades more substantially, suggesting sensitivity to extreme distributional shift. Critically, directional accuracy remains above 59% even at σ = 0.20, substantially above the 50% random baseline, indicating residual predictive power under severe input corruption.

6. Conclusion

BOLSTM-SATI is a new hybrid deep learning model for stock return prediction proposed in this paper, combining LSTM-based sequence modelling with multiple types of technical indicators, Bayesian-updated investor sentiment, and TPE-based hyperparameter optimisation. Based on CSI 300 data from 2015 to 2022, the model has achieved a directional accuracy of 67.3%, an MAE of 0.0082, and a backtesting Sharpe ratio of 2.41; thus, it outperforms the ARIMA, Vanilla LSTM,

DOI: 10.54254/2754-1169/2026.35422

XGBoost, and Transformer baselines significantly. Perturbation Robustness experiments show that the model is stable in the presence of noise.

The two following are the actual effects. Firstly, the mood of investors in social networks contains leading indicators for short-term stock price changes in the Chinese retail-oriented market that have not yet been reflected in technical indicators. Second, principled Bayesian hyperparameter optimisation is an essential part of the production-level deep learning system for finance, and it outperforms default-parameterised models in terms of consistency.

(i) Extend the framework to cross-sectional stock selection via a panel LSTM architecture; (ii) Augment high-frequency intraday data and limit order book features; (iii) Investigate a reinforcement learning formulation that directly optimizes portfolio-level objectives; (iv) Employ a sentiment fusion method for other emerging markets with a large number of retail investors.

References

[1] Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *Journal of Finance, 25*(2), 383–417.

[2] Shiller, R. J. (2015). *Irrational exuberance* (3rd ed.). Princeton University Press.

[3] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

[4] Brock, W., Lakonishok, J., & LeBaron, B. (1992). Simple technical trading rules and the stochastic properties of stock returns. *Journal of Finance, 47*(5), 1731–1764.

[5] Neely, C. J., Rapach, D. E., Tu, J., & Zhou, G. (2014). Forecasting the equity risk premium: The role of technical indicators. *Management Science, 60*(7), 1772–1791.

[6] Baker, M., & Wurgler, J. (2007). Investor sentiment in the stock market. *Journal of Economic Perspectives, 21*(2), 129–152.

[7] Liao, L., Zhang, X., & Zhu, N. (2022). Stock market participation and the internet. *Review of Finance, 26*(1), 193–232.

[8] Huang, A. H., Wang, H., & Yang, Y. (2023). FinBERT: A large language model for extracting information from financial text. *Contemporary Accounting Research, 40*(2), 806–841.

[9] Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. *Advances in Neural Information Processing Systems, 24*.

[10] White, H. (1988). Economic prediction using neural networks: The case of IBM daily stock returns. *Proceedings of IEEE ICNN, 2*, 451–458.

[11] Siami-Namini, S., Tavakoli, N., & Siami Namin, A. (2019). The performance of LSTM and BiLSTM in forecasting time series. *Proceedings of IEEE BigData 2019*, 3285–3292.

[12] 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.

[13] Ding, X., Zhang, Y., Liu, T., & Duan, J. (2015). Deep learning for event-driven stock prediction. *Proceedings of IJCAI 2015*, 2327–2333.

[14] Ke, G., et al. (2017). LightGBM: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems, 30*.

[15] Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting, 37*(4), 1748–1764.

[16] Faber, M. T. (2007). A quantitative approach to tactical asset allocation. *Journal of Wealth Management, 9*(4), 69–79.

[17] Wilder, J. W. (1978). *New concepts in technical trading systems*. Trend Research.

[18] Bollinger, J. (2002). *Bollinger on Bollinger bands*. McGraw-Hill.

[19] Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review. *Applied Soft Computing, 90*, 106–181.

[20] De Long, J. B., Shleifer, A., Summers, L. H., & Waldmann, R. J. (1990). Noise trader risk in financial markets. *Journal of Political Economy, 98*(4), 703–738.

[21] Stambaugh, R. F., Yu, J., & Yuan, Y. (2012). The short of it: Investor sentiment and anomalies. *Journal of Financial Economics, 104*(2), 288–302.

DOI: 10.54254/2754-1169/2026.35422

[22] Baker, M., & Wurgler, J. (2006). Investor sentiment and the cross-section of stock returns. *Journal of Finance, 61*(4), 1645–1680.
[23] Chen, H., De, P., Hu, Y. J., & Hwang, B. H. (2014). Wisdom of crowds: The value of stock opinions transmitted through social media. *Review of Financial Studies, 27*(5), 1367–1403.
[24] Yang, H., Liu, X. Y., Zhong, S., & Walid, A. (2020). Deep reinforcement learning for automated stock trading: An ensemble strategy. *Proceedings of ACM ICAIF 2020*.
[25] Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems, 25*.
[26] Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PLOS ONE, 12*(7), e018.