Proceedings of ICMRED 2026 Symposium: The Future of Work: Strategy, Workforce Transformation, and Organizational Renewal
DOI: 10.54254/2754-1169/2026.LD32715

An Intelligent Investment Framework Based on an Improved BILSTM-Attention Model

Sunzhuo Mo

Shenzhen Audencia Financial Technology Institute, Shenzhen University, Shenzhen, China
msz20061126@outlook.com

Abstract. Regarding the lack of uncertainty quantification and the threshold rigidity in predictions of deep learning models, this paper proposes an intelligent investment framework based on an improved Bidirectional Long Short-Term Memory (BILSTM)-Attention model. The model chooses Sigmoid gating instead of Softmax attention to achieve non-competitive information aggregation. It also uses Monte Carlo Dropout by performing multiple random forward passes during the prediction phase and estimates the uncertainty of the predictive distribution, which is mapped to confidence. Based on this, the paper designs a dynamic threshold mechanism, adjusting the threshold according to market volatility and generating a 'prediction & confidence' signal to guide portfolio rebalancing. The experiments on three A-share semiconductor stocks show that the improved model generally outperforms the standard Long Short-Term Memory (LSTM) and the baseline BILSTM-Attention model. The backtesting shows that after introducing the confidence, the strategy's total return increased from 3.64% to 5.12%, and the maximum drawdown decreased; the dynamic threshold further increased the return and Sharpe ratio to 5.97% and 2.17. The framework of this paper improves prediction accuracy and investment performance, providing new ideas for quantitative investment.

Keywords: Index Terms- BILSTM-Attention, Uncertainty quantification, Dynamic threshold, Confidence.

1. Introduction

Stock price prediction plays a crucial role in quantitative investment, asset allocation and risk management. Nowadays, deep learning models have become mainstream in financial time series forecasting [1]. However, improving prediction accuracy and applying it to practical investment decisions remain challenges for both academia and industry. Existing research mainly focuses on using Long Short-Term Memory (LSTM) and its related models for financial prediction [2]. However, these models have room for improvement in prediction accuracy. Meanwhile, they lack quantification and application of prediction uncertainty [3], and there are even fewer experiments combining price prediction with dynamic investment.

To address the above gaps, this paper proposes an improved Bidirectional Long Short-Term Memory (BILSTM)-Attention model and an intelligent investment framework. The main contributions of this paper include: 1) optimizing feature fusion and the attention mechanism, while

DOI: 10.54254/2754-1169/2026.LD32715

introducing Monte Carlo Dropout to improve prediction accuracy; 2) proposing a confidence evaluation system to provide a basis for risk assessment in investment decisions; 3) designing a dynamic investment strategy based on confidence and predicted returns.

The structure of this paper is as follows: Section 2 introduces the improved BILSTM-Attention model; Section 3 introduces the method for calculating confidence and investment strategies; Section 4 presents the experimental results and analysis; Section 5 summarizes the experimental conclusions and future directions.

2. Improved BILSTM-Attention model

2.1. Analysis of the drawbacks of the baseline BILSTM-Attention model

The baseline BILSTM-Attention model [4] has two shortcomings that directly affect its value in stock price prediction.

First, when using the softmax function, the model causes competition among time steps' weights [5]: when a new important time step appears, the model reduces attention to other time steps while overemphasizing this time step, leading to a polarized distribution of weights. Thus, medium-level auxiliary information is ignored. However, in financial markets, multiple unrelated events may simultaneously affect price trends, and such polarized weight allocation cannot adequately capture the independent effects of complex factors.

Second, the model outputs a deterministic estimated value ŷ, without quantifying uncertainty. It causes a lack of confidence in the results: in stock markets with high noise, high frequency, and large amplitude fluctuations, using predictions with unknown confidence for investment carries significant risk.

2.2. Improving the BILSTM-Attention model

In view of the shortcomings of the baseline model, this paper proposes the following three improvements, which together constitute a prediction module more suitable for financial scenarios.

Improvement 1: Introducing the Sigmoid Attention Mechanism.

This paper abandons the Softmax normalization that causes weight competition and introduces an independent gating vector when processing the hidden state of each time step:

$$g_t=\sigma(W_g h_t+b_g) \tag{1}$$

Here, σ is the Sigmoid function. The difference from traditional processing methods [6] is: first, for the hidden state h_t at each time step, it is transformed through an independent fully connected layer, with no direct competition between time steps; second, g_t is a vector, not a scalar, with the same dimension as h_t, allowing control over the information flow of each feature channel in h_t; third, the Sigmoid function is used to compress the output to the (0, 1) range, with its output acting as a gate to modulate the intensity of each feature dimension.

Then calculate the element-wise gated product to obtain the weighted result of a single feature: $c=g_t \odot h_t$. Finally, sum z_t over all time steps to calculate the context vector c: c[i] = z_1 [i] + z_2 [i] + ... + z_t [i].

$$c=\sum_{t=1}^{T} g_t \odot h_t \tag{2}$$

DOI: 10.54254/2754-1169/2026.LD32715

Here, ⊙ represents element-wise multiplication. After the above improvements, the model achieves non-competitive information aggregation and fine-grained feature selection.

Improvement 2: Introducing the MCDropout mechanism.

This paper transforms Dropout from a traditional regularization tool into a framework for Bayesian approximate inference. Unlike conventional methods, this Dropout layer remains active during both the training and inference phases of the model [7]. This operation makes the model equivalent to a slightly different sub-network during each forward pass. After multiple samplings (N forward passes) are operated, the set of prediction results can be regarded as Monte Carlo samples of the model's posterior predictive distribution. It providing a basis for quantifying uncertainty.

Improvement 3: Using end-to-end multi-dimensional feature fusion.

In this paper, various features are inputted into the model. Through the aforementioned vectorized gating mechanism, the model automatically learns the different importance level among different features and adjusts the weights of feature channels while processing the stock price time series. This avoids the subjective bias caused by manually designed weights and enhances the model's ability to capture features in different environments.

3. Construction of the intelligent investment framework

As shown in Figure 1, the intelligent investment framework consists of three core modules: a stock price prediction module based on the improved BILSTM-Attention, an uncertainty quantification module based on MCDropout, and a dynamic adaptive investment module.

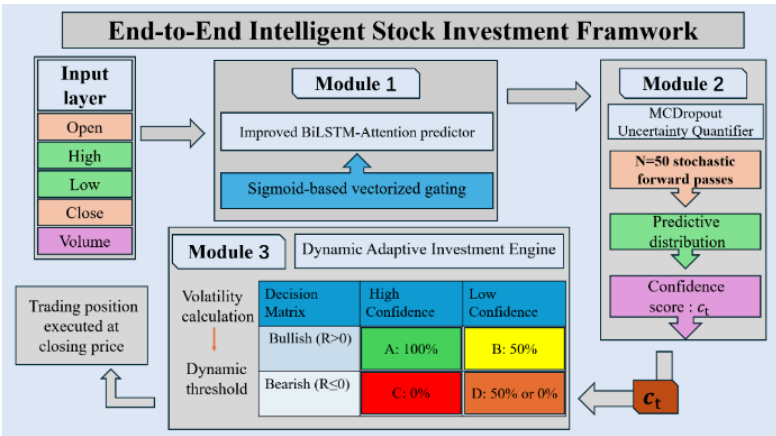

Figure 1. Composition of the intelligent investment framework (picture credit: original)

3.1. Uncertainty quantification module

Gal and Ghahramani proposed Monte Carlo Dropout and revealed that using Dropout during deep learning model training and keeping Dropout active during testing is equivalent to performing approximate variational inference in Bayesian neural networks. This method can achieve Bayesian neural network learning at minimal computational cost [8].

As shown in Figure 2, the uncertainty quantification module quantifies the confidence of the model's own predictions by introducing Dropout layers during training and keeping them active during inference, thereby performing efficient Bayesian approximate inference and providing a foundation for intelligent investment.

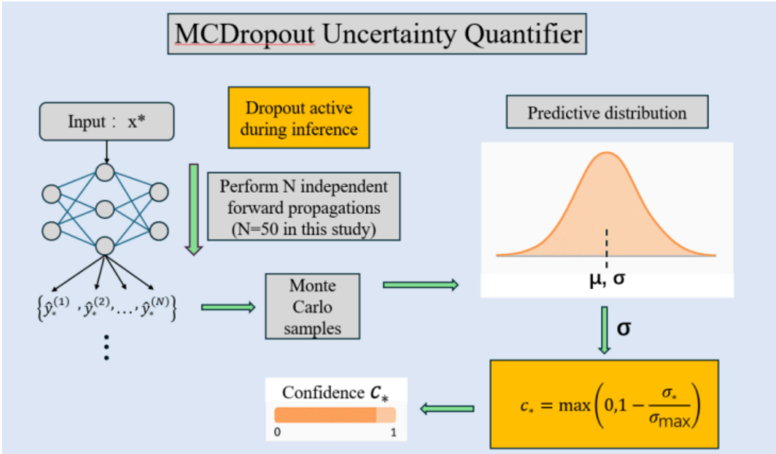

Figure 2. Uncertainty quantification process (picture credit: original)

3.1.1. Construction of predictive distribution

For a given sample x_* , perform N independent forward passes (in this paper, N=50). Since Dropout randomly deactivates some neurons in each pass, each propagation produces a series of slightly different predicted outputs: $\left\{\widehat{y}_*^{(1)}\widehat{y}_*^{(2)}\ldots\widehat{y}_*^{(N)}\right\}$. This set of outputs is equivalent to drawing a set of Monte Carlo samples from the posterior distribution predicted for x_* , providing a basis for estimating predictive uncertainty.

3.1.2. Uncertainty quantification and confidence reflection

Use the standard deviation σ_* of this set of samples as the measure of uncertainty. To obtain an intuitive and directly usable confidence, this paper maps σ_* to the [0,1] interval and defines it as the prediction confidence c_* :

$$c_*=\max\left(0,1-\frac{\sigma_*}{\sigma_{max}}\right) \tag{3}$$

Here σ_{max} is determined as the 95% of the standard deviations obtained from all samples in the validation set over N MCDropout forward passes. When the predictive distribution is highly concentrated, c_* approaches 1, indicating that the model is very confident in its prediction, and the trading signal is more reliable, suitable for adopting a more aggressive investment strategy; when the distribution is very dispersed, c_* approaches 0, indicating high uncertainty in the model's prediction, and a more cautious strategy should be taken to avoid potential risks.

3.2. Dynamic adaptive investment module

The dynamic adaptive investment module is the decision-making core of the entire intelligent investment framework. Its function is to convert predicted returns and confidence levels into specific portfolio adjustment instructions. Traditional quantitative strategies almost do not consider the model's own confidence and use fixed thresholds to filter information [9]. The volatility of stock prices is significantly time-varying; fixed thresholds cannot effectively control risk during periods of high volatility and fail to adequately capture opportunities during periods of low volatility [10]. This

DOI: 10.54254/2754-1169/2026.LD32715

paper introduces two innovations in this module: a strategy for dynamically adjusting confidence thresholds and a confidence-driven position adjustment strategy.

3.2.1. Calculation of the baseline confidence threshold (τ_{base})

The calculation of the baseline threshold is based on historical confidence sequences. τ_{base} is calculated using a sliding window method, with the window set to w days (in this paper, w equals 20, corresponding to the number of trading days in a month). The calculation steps are as follows: Based on the historical confidence set $C=\{ c_1,c_2,\ldots\ldots,c_{t-1} \}$, the moving average of historical confidence μ_t is calculated as the core reference for the baseline value. Here, c_i represents the confidence value obtained by MCDropout on the i-th trading day:

$$\tau_{base}=\frac{1}{w}\sum_{i=t-w}^{t-1} c_i \tag{4}$$

3.2.2. Calculation of the dynamic confidence threshold (τ_t):

The confidence threshold τ_t is dynamically calculated daily and varies with market volatility. In this paper, the rolling standard deviation of returns over the past M trading days (in this paper, M=20) is defined as market volatility. The formula for calculating the dynamic threshold is as follows [11]:

$$\tau_t=\tau_{base}+\gamma\cdot\left(\frac{v_t}{\bar{v}}\ 1\right) \tag{5}$$

Here, $\gamma>0$ is the sensitivity coefficient (in this paper, $\gamma = 0.15$), and $\bar{v}$ represents the average volatility over a past period (e.g., 20 trading days), used to assess the deviation of the current day's volatility v_t . When v_t is higher than the historical level $\bar{v}$, the threshold should increase, and the model only executes high-confidence rebalancing strategies to reduce risk; when market volatility is lower, the threshold should decrease, allowing the model to be more proactive in executing strategies and capturing opportunities.

3.2.3. Confidence-driven position adjustment strategy

On each trading day, the investment module receives two core parameters: the return direction derived from the predicted stock price r_t^{pred} (calculated from the predicted price and the current price) and the confidence c_* computed by the model. The decision rules are as follows:

When the model predicts an increase and the confidence is high, it outputs a Class A signal: aggressively buy, adjusting the position to 100%. If the model predicts an increase but the confidence is low, cautiously buy, adjusting the position to 50%. When the model predicts a decrease and the confidence is high, it outputs a Class C signal: firmly sell, adjusting the position to 0%. While the model predicts a decrease but the confidence is low, Class D signal is generated: if the actual stock price rose the previous day, adjust the position to 50% at appearance; if the stock price fell the previous day, maintain 0% at appearance. If there are two consecutive days of loss while maintaining a 50% position (closing price lower than the previous day's closing price), a stop-loss signal is triggered, reducing the position to 0%.

For Class D signals, the decision logic inside aims to balance risk and potential opportunity. When the stock price rose the previous day, there may be some market inertia or rebound momentum. Therefore, taking a 50% position allows the strategy to retain some flexibility without

DOI: 10.54254/2754-1169/2026.LD32715

being completely bearish. When the stock price has already fallen the previous day, to avoid further losses, the strategy chooses to fully liquidate the position to mitigate risk.

The above rules directly quantify the model's inherent uncertainty into principles that drive risk allocation, systematically incorporating the model's confidence into the risk management framework.

3.3. Workflow of the intelligent investment framework

This section integrates the above three modules into an executable system. The overall process is divided into two stages: offline training and online decision-making, both using the same model structure and parameters. Offline Training Stage: The model is trained end-to-end through supervised learning. This paper's training objective is to minimize the mean squared error to bring the predictions closer to the actual values. During training, the Adam optimizer adjusts the model parameters, combined with early stopping and learning rate decay strategies. At the same time, the Dropout layer after the BILSTM layer remains active, preventing model overfitting during training and providing a structural foundation for quantifying uncertainty in subsequent MCDropout operations during inference. Online Decision-Making Stage: After model training is completed, it enters the trading decision phase. On each trading day, the following six steps are executed in sequence to form a complete decision-making loop: data preparation, prediction generation and uncertainty evaluation, confidence calculation, dynamic threshold calculation, position adjustment execution, and performance recording.

4. Experiments and results analysis

4.1. Data preparation

The data for this paper comes from the Akshare database, selecting three Chinese A-share semiconductor industry stocks: GigaDevice (603986), Beijing Junzheng (300223), and Shikong Technology (605178), with a time span from January 20, 2021, to January 20, 2026 (a total of 5 years), to test the model's performance in an active and volatile market. The feature selection in this paper includes five items: open price, highest price, lowest price, closing price, and trading volume. The data preprocessing workflow is as follows: missing values were handled using forward fill, outliers (using the 3-sigma method) were replaced with the median, and all features were normalized to the [0,1] interval using MinMax-Scaler. In this paper, a 20-day sliding window was used to construct input sequences to predict the closing price of the next trading day. The evaluation metrics for the model's predictive ability are MSE, RMSE, MAE, MAPE, and R^2; the profitability and risk control capability of the intelligent investment framework are evaluated using seven indicators: directional accuracy, total return, base return, excess return, win rate, Sharpe ratio, and maximum drawdown. The models selected for this paper include the standard LSTM model, the baseline BILSTM-Attention model, and the improved BILSTM-Attention model.

4.2. Verification of MCDropout effectiveness and control of experimental stability

To verify the effectiveness of MCDropout in quantifying uncertainty, this paper conducted 50 stochastic forward passes with Dropout on 20 samples from the test set. The results show that the mean error of the samples is 0.0272, and the mean predicted standard deviation is 0.0279, indicating that the model's predictions are stable and the errors are small. At a 95% confidence level, the

DOI: 10.54254/2754-1169/2026.LD32715

coverage of the prediction interval is 90%, significantly higher than the random level (50%), confirming that MCDropout can reasonably quantify predictive uncertainty.

Although MCDropout is effective in quantifying uncertainty, there are certain issues: stochastic sampling is variable, and the predictive distribution generated each time differs to some extent; and the uncertainty estimation has limitations, as the model's uncertainty estimates for a small number of samples may be biased. Therefore, to control experimental stability and reduce the impact of random fluctuations on the experiments, all experiments involving MCDropout in this paper were repeated independently 15 times, and the average was taken as the final result.

4.3. Prediction results and performance analysis

This subsection uses the standard LSTM model, the baseline BILSTM-Attention model, and the improved BILSTM-Attention model to predict three stocks. The results are shown in Table 1, Table2 and Table 3.

Table 1. 603986's predictive indicators

Stock:603986	MSE	RMSE	MAE	MAPE	R^2
LSTM	0.0039	0.0574	0.0447	6.92%	0.9454
Basic BILSTM-Attention	0.0051	0.0681	0.0579	10.61%	0.8903
Improved BILSTM-Attention	0.0023	0.0441	0.0366	5.68%	0.9274

Table 2. 300223's predictive indicators

Stock:300223	MSE	RMSE	MAE	MAPE	R^2
LSTM	0.0018	0.0415	0.0336	5.31%	0.9376
Basic BILSTM-Attention	0.0036	0.0623	0.0477	8.41%	0.8929
Improved BILSTM-Attention	0.0018	0.0307	0.0314	5.41%	0.9334

Table 3. 605178's predictive indicators

Stock:605178	MSE	RMSE	MAE	MAPE	R^2
LSTM	0.0047	0.0648	0.0433	9.06%	0.9348
Basic BILSTM-Attention	0.0085	0.0973	0.0759	18.79%	0.8311
Improved BILSTM-Attention	0.0031	0.0502	0.0392	8.37%	0.9523

From Table 1, Table 2, and Table 3, it shows that the baseline BILSTM-Attention model performs the worst, with its error metrics (RMSE, MAPE, etc.) generally higher than LSTM, indicating that the basic attention mechanism may introduce overfitting or noise. In contrast, the improved model outperforms the baseline model on all stocks and surpasses LSTM on most metrics: for example, on 603986, the MSE of the improved model decreased by 41% compared to LSTM, and MAPE dropped from 6.92% to 5.68%; on 605178, the R^2 of the improved model reached 0.9523, far higher than LSTM's 0.9348.

Comparing the performance of the three models in terms of goodness of fit (R^2), the improved model shows cases that it lags behind the standard model. The reason for this may be that the attention mechanism of the improved model filters out some minor noise, and the bidirectional

DOI: 10.54254/2754-1169/2026.LD32715

LSTM structure makes the model more focused on trends rather than short-term fluctuations. Therefore, the improved model has an advantage in predicting trends but is slightly less sensitive to short-term fluctuations. Subsequent experiments on directional prediction accuracy also support this conjecture.

In all, the improved model's error metrics are comprehensively superior, indicating that the model, through Sigmoid attention and MCDropout, reduces prediction bias while maintaining good fitting ability, resulting in overall better predictive performance.

4.4. Backtesting results of simple investment strategy

This section combines different prediction models with a simple investment strategy to verify the investment performance of the predictions from the three models. The backtest was conducted over 20 trading days for Gigadevice (603986) from December 4 to December 31, 2025. The total return during the buy-and-hold strategy period was 1.83%. In this paper, the backtest set the transaction fee to 0.2% and slippage to 0% to simply mimic real trading.

The rules of the simple strategy are: go all-in if an increase is predicted, stay out of the market if a decrease is predicted. The results are shown in Table 4.

Table 4. Simple strategy backtest result

	Total return	Excess return	Win rate	Direction prediction accuracy
LSTM	0.95%	-0.88%	51.73%	60.75%
Basic BILSTM-Attention	2.54%	0.71%	54.58%	55.14%
Improved BILSTM-Attention	3.64%	1.81%	56.29%	64.79%

As can be seen from Table 4, the improved BILSTM-Attention model has the greatest advantage in backtesting with simple strategies. The excess return of the improved model is 1.81%, higher than the baseline model's 0.71% and the standard model's -0.88%. At the same time, the directional prediction accuracy of the improved model is 64.79%, higher than the other two models. It can be seen that the improved model, while enhancing prediction fit, filters out short-term fluctuations and noise, performing the best under this strategy.

4.5. Ablation experiments on the intelligent investment framework

This section uses two sets of ablation experiments to verify the effectiveness of the intelligent investment framework. By introducing confidence levels and dynamic thresholds, it analyzes the performance of this framework and its strategies in actual investments, verifying the advantages of this framework from multiple data perspectives.

4.5.1. Ablation experiment 1: introducing confidence

This experiment is based on an improved model, comparing two strategies: "only using predicted returns" and "predicted returns with confidence." The strategy of only using predicted returns involves going fully long if an increase is predicted and holding cash if a decrease is predicted. The confidence strategy uses the decision rules from Section 3.2.3. The results are shown in Table 5 and Figure 3.

DOI: 10.54254/2754-1169/2026.LD32715

Table 5. Backtesting results with confidence

	Total return	Excess return	Win rate	Sharpe ratio	Maximum drawdown	Number of trades
Only returns	3.64%	1.81%	56.29%	1.5823	-4.11%	10.6
With confidence	5.12%	3.29%	58.73%	1.7208	-3.76%	11.9

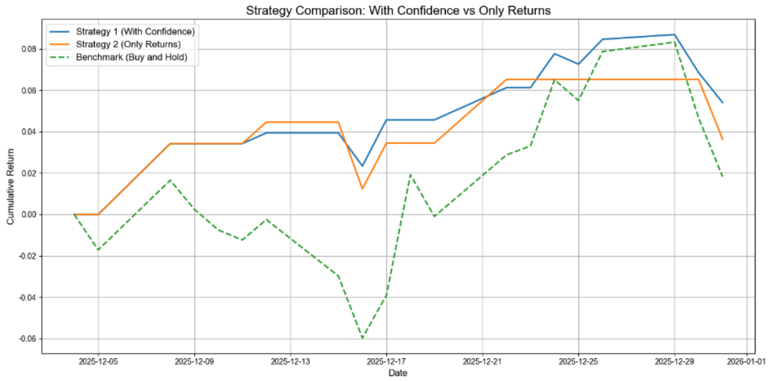

Figure 3. Backtesting results with confidence (picture credit: original)

From Figure 3, it can be seen that after introducing the confidence strategy, excess returns increased from 1.80% to 3.29%, and the win rate improved from 56.29% to 58.73%. In addition, both risk indicators have improved: the Sharpe ratio increased from 1.5823 to 1.7208, and the maximum drawdown decreased from -4.11% to -3.76%, indicating that risk is effectively controlled.

From Figure 3, it can be seen that the return curve corresponding to the introduction of the confidence strategy (blue solid line) is higher than or coincides with the return curve without the confidence strategy (orange solid line) for most of the time, and both are higher than the buy-and-hold strategy for the majority of the period.

This result demonstrates that the model's uncertainty information has significant incremental value for investment decisions. By adopting a conservative position during low-confidence periods, the strategy effectively avoids losses caused by erroneous signals while retaining offensive capability when confidence is high. However, the robustness of the strategy needs improvement, as it failed to reduce positions in time to avoid risk when stock prices suddenly dropped.

4.5.2. Ablation experiment 2: introducing dynamic thresholds

This experiment is based on the confidence strategy, comparing the static threshold strategy with the dynamic threshold strategy. The dynamic threshold strategy adjusts in real-time according to market volatility (see formula in Section 3.4.2). The results are shown in Table 6 and Figure 4:

Table 6. Backtest results with dynamic thresholds

	Total return	Excess return	Win rate	Sharpe ratio	Maximum drawdown	Number of trades
Static threshold	5.12%	3.29%	58.73%	1.7208	-3.76%	11.9
Dynamic threshold	5.97%	4.14%	61.54%	2.1746	-3.09%	12.2

DOI: 10.54254/2754-1169/2026.LD32715

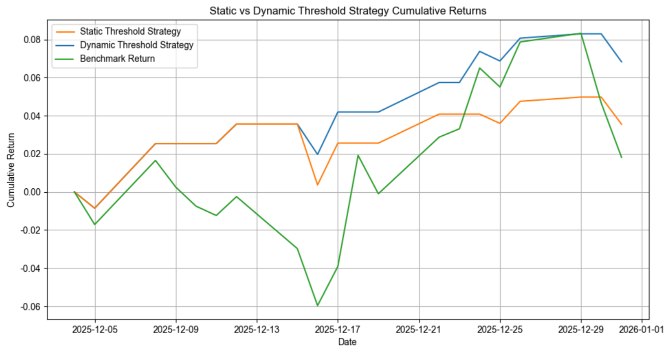

Figure 4. Backtest results with dynamic thresholds (picture credit: original)

From Table 6, it can be seen that the dynamic threshold strategy, based on the static threshold strategy, increases excess returns to 4.14% and raises the win rate from 58.73% to 61.54%. Both risk indicators have also improved, with the Sharpe ratio increasing from 1.7208 to 2.1746 and the maximum drawdown decreasing from -3.76% to -3.09%.

From Figures C and D, although both static and dynamic strategies have misjudgments, the loss of returns for the static threshold strategy (orange solid line) is noticeably higher than that of the dynamic strategy (blue solid line). This indicates that in highly volatile markets with high model uncertainty, a dynamic threshold can promptly raise the threshold, prompting the strategy to adopt a more conservative position (for example, adjusting from full position buying to half position observation), thereby effectively limiting losses. Therefore, the dynamic threshold strategy shows better robustness and stronger risk resistance.

4.5.3. Analysis of result distribution range

Figure 5 is the distribution chart of results from 15 consecutive experiments, from which can see: the results of the basic strategy are the most concentrated, with the smallest range and the lowest average return; the results of the static threshold strategy are the most dispersed, with the largest range and relatively higher average return; the results of the dynamic threshold strategy are relatively concentrated compared to the static threshold strategy, with a relatively smaller range and the highest average return.

This phenomenon reflects three advantages of the dynamic threshold strategy: first, this strategy buffers the random fluctuations caused by MCDropout, smoothing short-term volatility in confidence scores through the market state awareness mechanism; second, the dynamic adjustment mechanism automatically calibrates decision boundaries according to market volatility, improving decision consistency and stability; finally, while capturing high-return market opportunities, this strategy effectively balances risk and return by filtering out low-confidence signals through adaptive threshold control.

DOI: 10.54254/2754-1169/2026.LD32715

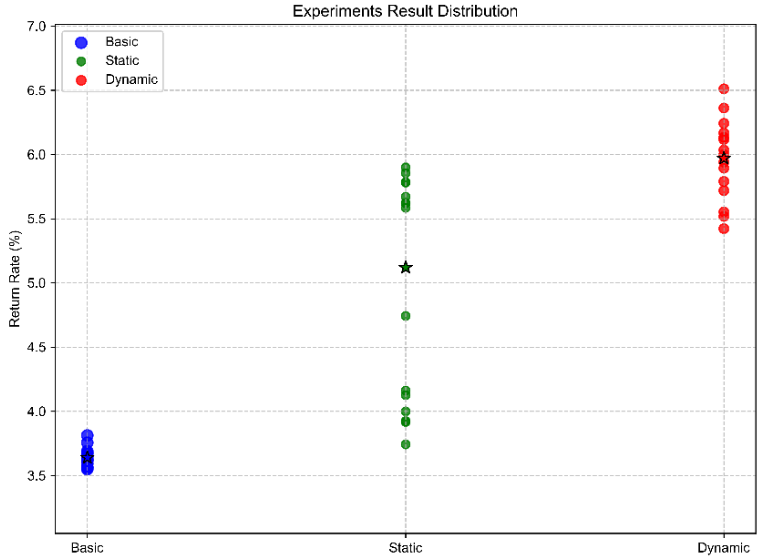

Figure 5. Distribution of results range

5. Conclusion

This paper addresses three core issues in stock price prediction and quantitative investment: insufficient prediction accuracy, lack of uncertainty quantification, and the disconnect between prediction and investment, and proposes an end-to-end intelligent investment framework.

This paper improves the baseline BILSTM-Attention model through innovatively introducing a Sigmoid-gated attention mechanism and adopting end-to-end feature fusion. What's more, this paper incorporates the Monte Carlo Dropout mechanism into the model. The prediction uncertainty is also quantified into intuitive confidence levels. Lastly, this paper designs a dynamic adaptive investment module. This module not only uses confidence to filter trading signals but also innovatively introduces a dynamic threshold adjustment mechanism.

Although this framework has achieved good results in prediction and investment, there are still the following limitations and future research directions.

Firstly, the model sacrifices the ability to fit certain fluctuations when pursuing trend prediction accuracy. Secondly, the inherent sampling variability of MCDropout leads to a dispersed distribution of strategy results, posing a challenge for stable deployment of the model. Finally, this paper has only been validated in volatile market environments, and its adaptability to other environments such as bull and bear markets remains to be tested.

Based on the above limitations, further research can focus on the following directions: Comprehensive validation of the framework: applying the framework to other asset classes and testing the model's generalization ability across different market environments, including bull and bear markets. Refining the strategy: designing more detailed portfolio adjustment strategies to enhance risk resistance. Exploring more stable uncertainty quantification methods: attempting techniques such as deep ensemble Bayesian neural networks to obtain stable uncertainty estimates and provide more reliable risk evidence.

References

[1] Gudelek, M. U., Boluk, S. A., & Ozbayoglu, A. M. (2017). A deep learning based stock trading model with 2-D CNN trend detection. In 2017 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 1–8). IEEE.

DOI: 10.54254/2754-1169/2026.LD32715

[2] 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.

[3] Alqarafi, A., Batool, H., Abbas, T., et al. (2024). Estimating uncertainty in deep learning methods and applications. In 2024 International Conference on Computer and Applications (ICCA) (pp. 1–6). IEEE.

[4] Lu, W., Li, J., Wang, J., et al. (2021). A CNN-BiLSTM-AM method for stock price prediction. Neural Computing and Applications, 33(10), 4741–4753.

[5] Luo, A., Zhong, L., Wang, J., et al. (2024). Short-term stock correlation forecasting based on CNN-BiLSTM enhanced by attention mechanism. IEEE Access, 12, 29617–29632.

[6] Ranjan, P., Khan, P., Kumar, S., et al. (2023). Log-Sigmoid activation-based long short-term memory for time-series data classification. IEEE Transactions on Artificial Intelligence, 5(2), 672–683.

[7] Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In International Conference on Machine Learning (pp. 1050–1059). PMLR.

[8] Hassan, M. M. (2024). Bitcoin price prediction using deep Bayesian LSTM with uncertainty quantification: A Monte Carlo dropout–based approach. Stat, 13(3), e70001.

[9] Reis, P. D., Serra, A. P. S. F. M., & Gama, J. M. P. (2025). The role of deep learning in financial asset management: A systematic review [Preprint]. arXiv: 2503.01591.

[10] Saberi, E., Pirgazi, J., & Ghanbari Sorkhi, A. (2024). A machine learning approach for trading in financial markets using dynamic threshold breakout labeling. The Journal of Supercomputing, 80(17), 25188–25221.

[11] McAleer, M., & Da Veiga, B. (2008). Single-index and portfolio models for forecasting value-at-risk thresholds. Journal of Forecasting, 27(3), 217–235.