Theoretical Framework and Empirical Analysis of Stock Price and Trend Prediction Using LSTM Models

Xiaoyu Yang

Reading Academy, Nanjing University of Information Science & Technology, Nanjing, China

xiaoyu.yang@student.reading.ac.uk

Abstract. Changes in stock market prices are nonlinear, high-noise, and they change with time, which makes it hard for typical time series models to accurately capture their complicated patterns. This paper develops a forecasting model for stock prices using Long Short-Term Memory (LSTM) networks, with a focus on the A-share market to thoroughly investigate the effectiveness of LSTM models in predicting stock prices and trends. The research employs the Kaggle dataset titled "S&P 500 Stock Data" covering the years 2013 to 2018, which includes key daily trading indicators like opening price and trading volume. When comparing the predictive performance of LSTM models to that of traditional machine learning models like random forests, it is clear that LSTM models are better at predicting short-term trends and prices than both traditional machine learning models and traditional time series models. This research offers novel analytical insights for forecasting financial markets and possesses practical implications for investor decision-making and risk management.

Keywords: Long Short-Term Memory, Stock Price Prediction, Time Series, Deep Learning.

1. Introduction

The stock market is an important part of a contemporary nation's financial system. It shows how the economy is doing and has a direct effect on its growth. For a long time, predicting stock prices and trends has been a popular and difficult area of financial research. Investors need trustworthy prediction tools more and more as China's capital market expands. Most individual investors make bad investment choices because they're not very knowledgeable about the market and are readily affected by how people feel about it. For predicting stock price movements and trends, there are two major methodological types: linear and nonlinear prediction models. The linear prediction models mainly consist of the Autoregressive Integrated Moving Average (ARIMA) model and the Exponential Smoothing model. Still, due to the uncertain and noisy characteristics of financial time - series and considering the effects of various macro - economic factors like economic situations, policies, and corporate performance, achieving accurate forecasts is highly arduous [1]. In this context, a pressing issue that necessitates immediate attention is the development of strategies to reduce emotional influence and improve the scientific integrity of investment decisions. Recently, deep learning technology has shown that it offers distinct advantages for time series forecasting since it can fit nonlinear data very well. The LSTM (Long Short - Term Memory) model has turned into an essential tool in the domain of stock price forecasting studies. It owes its importance to the capacity of dealing with both long - term and short - term relationships in sequential data [2]. This particular study fortifies the theoretical architecture of financial time - series prediction. By means of empirical investigations, it verifies the usability of the LSTM model in predicting stock prices and their trends and also demonstrates the potential of deep - learning techniques in clarifying the rules that govern financial markets. In practical applications, the outcomes of this research can offer scientific decision - making guidance to investors. This guidance helps investors to steer clear of market perils. Simultaneously, it provides a reference for regulatory institutions to optimize market mechanisms and safeguard financial steadiness. Recent research predominantly centers on three aspects. To start with, the employment of traditional time - series models, like ARIMA (Autoregressive Integrated Moving Average) and GARCH (Generalized Autoregressive Conditional Heteroskedasticity), for stock price forecasting. Even though these models have assisted researchers in achieving more precise forecasts of the financial market [3], they are plagued by several major drawbacks. For one thing, they are incapable of depicting nonlinear time - series. For another, they demand the fulfillment of numerous assumptions prior to initiating the modeling. Third, they have a challenging time getting good results when it comes to predicting stocks. This shows that typical time series models don't work well with stock market data that is complicated and nonlinear [4]. To put it another way, ARIMA models can only be used to predict time series data if the data is stable. If the data is not stationary, it is impossible to find patterns. Second, machine learning models like random forests and SVM can pick up on nonlinear features, but they can't handle temporal correlations very well. The mean prediction of random forest decision trees struggles to accurately capture short-term sudden changes in time series, while SVM is relatively more flexible in handling nonlinear fluctuations, but it has not fundamentally resolved the modeling issue of temporal dependencies [5]; third, the exploration of deep learning models, such as the combination of CNN and LSTM, but most studies suffer from issues like single feature selection and limited sample intervals [6]. This work improves the depth and breadth of prior research by developing an LSTM model that combines several features.

2. Theoretical Basis and Method Design

2.1. LSTM Model Principle

Sepp Hochreiter and Jürgen Schmidhuber came up with LSTM as a better form of recurrent neural networks (RNNs). It effectively fixes the problems of gradient vanishing and gradient explosion that classic RNNs have by adding input gates, forget gates, and output gates [7, 8]. The architecture of the cell state is the most significant part. It can save critical information for a long time and forget information that isn't important, which makes it great for working with time series data with long-term dependencies, like stock prices [9].

2.2. Model Construction and Experimental Design

Data Preprocessing Plan. The data processing workflow utilized in this investigation is as follows:

First, the research chose the A-share benchmark index from the Kaggle dataset "S&P 500 Stock Data" as the research sample. The time range was from January 2013 to December 2018. There were five main trading features: opening price (open), high price (high), low price (low), closing price (close), and trading volume (volume). These five features include price and trading volume over different time periods. This gives a completer and more multidimensional picture of stock trading conditions and market dynamics [10]. This gives more information to help with in-depth analysis of market behavior and the building of related models.

The next step is to standardize. To transfer the characteristics and labels (closing price) to the [0,1] interval in a consistent way, min-max standardization is utilized. A standardizer (Min-Max Scaler) is made by combining the training set's features and labels such that they are all processed on the same scale.

Next is time-series data partitioning, which employs time-series segmentation to split the data into training, validation, and test sets in a 7:2:1 ratio. This stops future data from leaking into past training.

Lastly, a sliding window is made. An input window of length 4 (input length=4) is utilized to get historical sequence features, and the next day's closing price (output length=1) is guessed. The obtain rolling window multi-step function makes the three-dimensional input format (number of samples × time steps × number of features) that the LSTM needs. Research shows that sliding window prediction is very accurate and more resilient [11], which is why it is used in our model.

LSTM Model Construction. The structure of the multi-feature LSTM model created in this study is as follows: The input layer gets a tensor with the shape (4, 5), where 4 is the length of the time step and 5 is the feature dimension (open/high/low/close/volume). The hidden layer has 2 LSTM units (num blocks=2), with 64 neurons in the first layer and 32 neurons in the second layer. These are used to find long-term dependencies in temporal features. The output layer has 1 neuron, which uses a linear transformation to give the predicted closing price for the next day. Also, a regularization method is added that uses gradient clipping (clip grad norm_=0.15) to stop gradient explosion and an early stopping strategy (patience=10) to stop over-fitting. This means that training stops when the validation loss doesn't go down for 10 epochs in a row.

Training Parameter Selection and Settings. Optimizer selection: AdamW, weight decay 0.005, with faster convergence speed and better learning performance [12]; Learning rate: initial 0.001, dynamically decayed using the ReduceLROnPlateau scheduler; loss function is MAE, which is more robust to abnormal stock price fluctuations [13]; batch size: 32; training epochs: 100, which are combined with early stopping to prevent over-fitting.

Comparison Model Design. To validate the superiority of multi-feature LSTM, the study selects the following traditional comparison models: Logistic Regression: Maximum iteration counts of 1000, which is used to predict the direction of stock price movements. Random Forest: It is setting up 100 decision trees and uses ensemble learning to capture nonlinear relationships. XGBoost: it uses the logloss evaluation metric, efficiently handling structured data. SVM: An RBF kernel function with probability outputs, behaving specific contrast. ARIMA: A classic time series model used for prediction tasks [12].

Evaluation Indicator Selection. This study adopts separate evaluation indicators for classification tasks and regression tasks.

For classification tasks, the indicators used include Accuracy, Precision, Recall, F1-score, and Area Under the ROC Curve (AUC); Accuracy reflects the overall classification correctness of the model (calculated by the ratio of correctly identified samples to total samples) but may be biased in imbalanced data (e.g., more rising than falling stock samples); Precision measures the reliability of positive predictions (the proportion of predicted positives that are actual positives) and is effective for reducing false positives in scenarios like stock anomaly detection; Recall evaluates the model's ability to capture true positives (the proportion of actual positives correctly predicted) and highlights false negative risks, requiring a balance with Accuracy in financial trend prediction; F1-score is the harmonic mean of Precision and Recall, balancing the two indicators to reflect the model's stable performance on positive samples; AUC (the area under the ROC curve, with false positive rate on the x-axis and true positive rate on the y-axis) indicates the model's ability to distinguish between positive and negative samples, and values closer to 1 mean better differentiation, making it suitable for financial classification tasks with complex data distributions (e.g., binary prediction of stock price rise or fall).

For regression tasks, the indicators used include Mean Squared Error (MSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Coefficient of Determination (R^2); MSE reflects the deviation between predicted and actual values, is sensitive to large errors and outliers, and is suitable for financial scenarios where error magnitude matters (e.g., stock price prediction); MAE measures the average absolute deviation between predictions and actuals, is less affected by extreme values than MSE, and helps determine the average error scale in tasks like financial asset yield prediction; MAPE represents the percentage difference between predictions and actuals, uses relative values to eliminate data scale impacts, and facilitates cross-dataset performance comparison (e.g., comparing prediction errors of stocks with different market capitalizations); R^2 indicates the model's ability to explain dependent variable changes (ranging from 0 to 1), with values closer to 1 meaning better data fitting, and is effective for evaluating the model's capture of overall trends in financial time series forecasting (e.g., exchange rate trend prediction).

3. Empirical Results and Analysis

3.1. Stock Trend Assessment

Stock Price Trend Dimension. The blue curve in the stock price trend chart (Figure 1) shows the closing price of the stock, which makes it easy to see how the price has changed over time. The yellow 7-day moving average (7-day MA) and green 21-day moving average (21-day MA) are used to smooth out short-term noise and show medium-term trends.

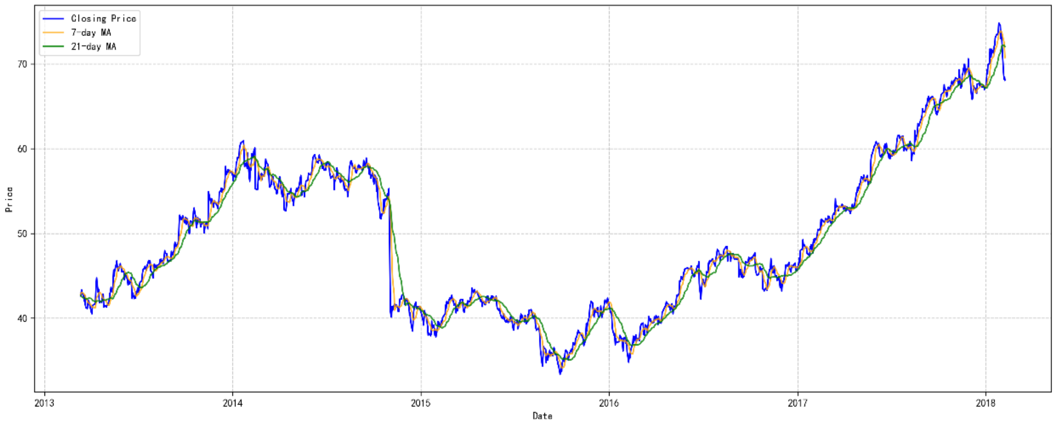

Fig. 1 A Stock Price Trend (Picture credit: Original)

From time series point of view, stock prices and moving averages went up together from 2013 to 2014, with a strong correlation between the two. This showed a clear upward trend and a strong bullish market mood. From 2014 to 2016, stock prices went down and stayed there, with moving averages going down and crossing. The 21-day moving average showed more clear lagging features because it had a longer period. During this time, the trends were all over the place, with bullish and bearish forces fighting hard against each other. From 2016 to 2018, stock prices started to rise again as moving averages lined up in a positive way, setting a long-term upward path. These variations in trends show how things like how people see the value of stocks, the state of the economy, and how well a firm is doing all work together to help us understand how stock prices vary.

Trading Volume Dimension. The volume chart (Figure 2) shows purple bars. During the time when stock prices were going up from 2013 to 2014, volume often hit record highs. This meant that people were trading a lot and had enough buying power to keep the trend going, which was confirmed by the rise in volume and stock prices. Between 2014 and 2016, during the decline in stock prices, the trading volume decreased overall. There were occasional minor upticks in volume. These were largely due to panic - driven selling or short - term capital speculation. However, these increases were short - lived as there was insufficient buying pressure. This indicated that the market had recognized the downward price trend. After 2016, when stock prices began to climb once more, the trading volume did not return to its previous peak levels. Instead, as stock prices ascended, the trading volume stabilized. This suggested that capital was gradually re - entering the market. It verified the upward trend in terms of volume, echoing the market adage "volume precedes price." It also helped figure out when trends were going to change or continue.

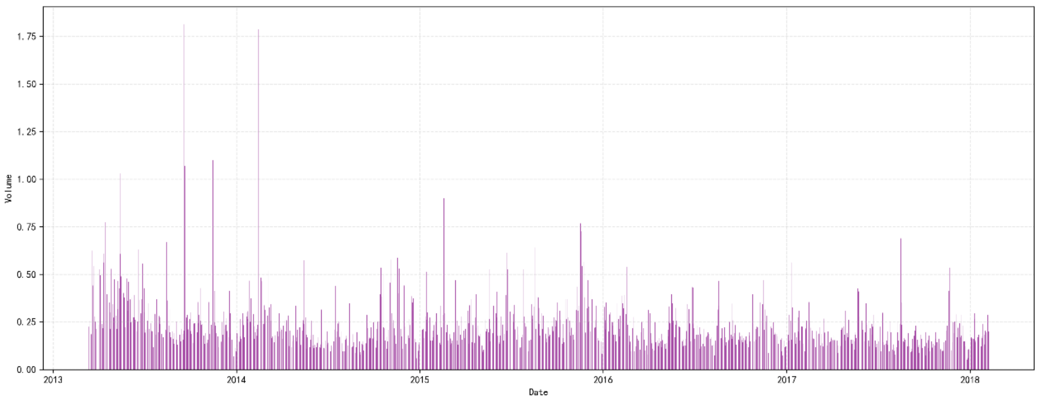

Fig. 2 Trading Volume (Picture credit: Original)

RSI Indicator Dimensions. The red line on the Relative Strength Index (RSI) chart (Figure 3) shows the RSI value. The green and yellow dashed lines show the 30 and 70 levels, which are usually regarded as the oversold and overbought levels. The RSI stayed mostly between 50 and 70 during the rise in stock prices from 2013 to 2014, which meant that there was enough upward momentum. During the period from 2014 to 2016, when stock prices were going up and down, the RSI often fell below 30 and entered the oversold zone, which meant that stock prices were too low. The RSI went back above 50 in 2016, which was when the stock price hit its lowest point and started to rise again. This confirmed the signal for a trend reversal. During the upward phase from 2016 to 2018, the RSI stayed between 50 and 70 without being in overbought territory, which showed that this upward trend was strong. The RSI adds to price and volume trend analysis by assessing the relative strength of bullish and bearish forces. This gives you a way to judge trends based on momentum.

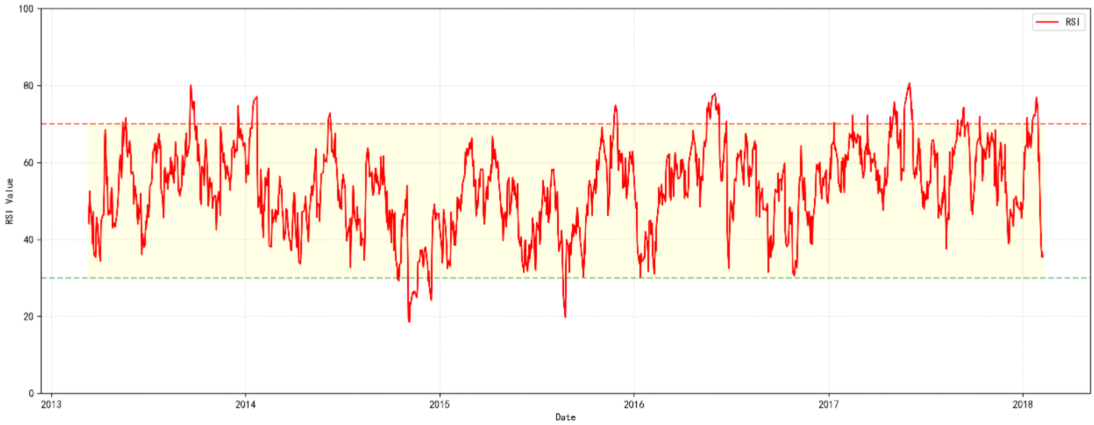

Fig. 3 Relative Strength Index (Picture credit: Original)

The three criteria are looked at together. Price trends show the definite direction, trade volume shows how strong the trend is and how investors feel about it, and RSI shows how strong the bulls and bears **are**. They work together to provide a method for looking at market trends that shows how stock trends formed, changed, and confirmed between 2013 and 2018 from multiple points of view. This sets the stage for later model analysis.

3.2. Model Performance Evaluation

Comparison of Model Price Fluctuation Predictions. This study employs a radar chart (Figure 4) to do a multi-dimensional analysis of the efficacy of logistic regression, random forest, XGBoost, support vector machine (SVM), and long short-term memory network (LSTM) in the classification of financial data. The main distinctions are as follows:

The chart indicates that the LSTM radar chart has the largest polygon area, which means it works well. It has an advantage in the AUC metric, which means it can figure out the differences between positive and negative samples of changes in financial stock prices and successfully capture complicated data aspects. It also does well with F1 value and recall. It has a low chance of "false negatives" and can recognize true positive samples well, like when stock prices are going up. By balancing precision and recall, it keeps positive sample predictions stable. This is helpful for complicated financial situations where you need to find a balance between "false negatives" and "false positives."

On radar charts, both the XGBoost and Random Forest models exhibit comparable patterns and demonstrate high accuracy. This makes them suitable for scenarios where minimizing false positives is crucial, such as detecting abnormal stock transactions, as they can strictly control mislabeling. However, in terms of overall metrics like AUC and F1 scores, they don't perform as effectively as the LSTM model. When dealing with financial data characterized by unevenly distributed or non - linear features, these models struggle to distinguish between positive and negative samples and maintain the stability of positive - class predictions. As a result, it becomes challenging to achieve an appropriate balance between false negatives and false positives.

For the logistic regression and SVM models, the extent of radar chart coverage is restricted, and their performance is evidently subpar. When handling financial data with imbalanced and non - linear relationships, the overall classification accuracy can be easily disrupted. This leads to poor performance in comprehensive metrics such as AUC and F1 scores, as well as difficulties in separating positive and negative samples and ensuring stability in positive - class predictions. Logistic regression faces difficulties in accommodating non - linear features, while the SVM is influenced by kernel functions and parameters, making it less adaptable for generalization. Thus, it is more suitable as a basic model or for straightforward applications. It is clear that LSTM is the better choice if you need to balance "false negatives" and "false positives" (for example, while predicting stock price patterns).

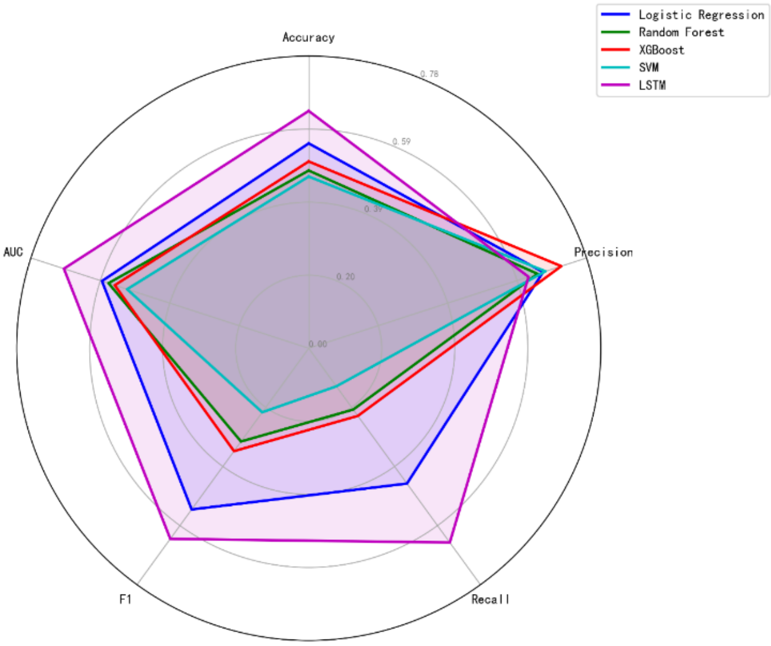

Fig. 4 Radar chart for model performance comparison (Picture credit: Original)

Comparison of model stock price fits. To visually compare the fitting effects of different models on stock price time series data, random forest and long short-term memory network (LSTM) were selected for visualization analysis.

The blue dotted line in figure 5 shows the real stock price, and the green dotted line shows the random forest prediction value. In general, the random forest model can show the long-term trend of stock prices, like when prices are going up for a long time and the prediction curve matches the direction of the actual curve. But it has big problems fitting short-term changes. When prices change quickly (as at sample index 80), the predicted values are very different from the actual values. The random forest model can't find "sequential dependencies" in time-series data very well, which makes it hard to accurately reproduce the short-term volatility of stock prices.

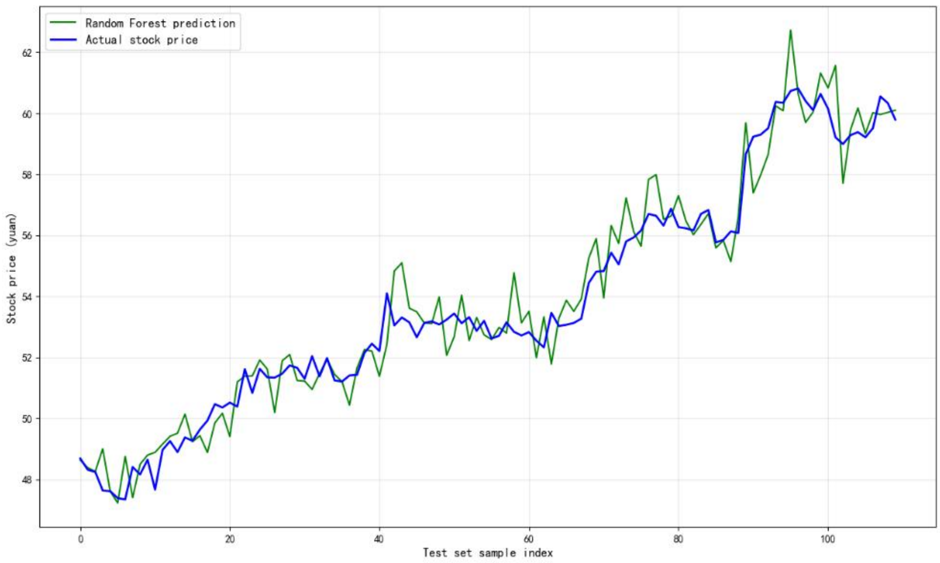

Fig. 5 Random Forest prediction results (Picture credit: Original)

The green line in figure 6 displays the LSTM prediction, and the red line shows the real stock price. LSTM is a time-series-specific model that does a better job of fitting the trends and volatility of stock price sequences. For example, when stock prices are stable (e.g., sample indices 40–60), the prediction curve closely matches the actual curve, showing that it can effectively capture temporal dependencies. In situations with extreme volatility (e.g., near sample index 90), the LSTM model does a better job of capturing the "peaks and valleys" of stock prices than the random forest model, even though prediction errors still happen.

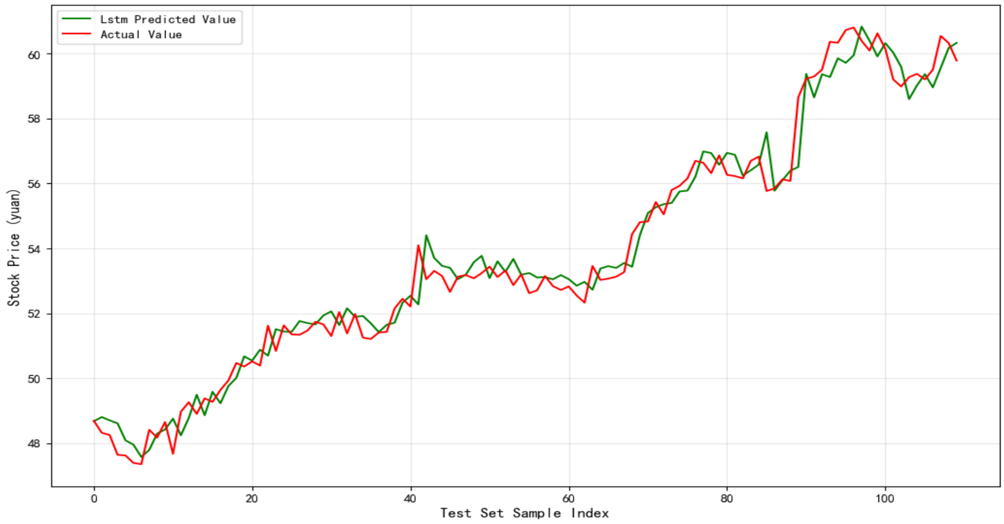

Fig. 6 Stock LSTM prediction results (Picture credit: Original)

The blue dotted line in figure 7 shows the ARIMA forecast value, and the red dashed line shows the real stock price. ARIMA is a standard time series model that has a lot of problems when it comes to fitting stock price sequences. This is because its main logic is based on linear autoregression and moving average principles. It can only momentarily match actual values during the first few phases (e.g., around sample indices 0–5) since stock prices change so often. When stock prices go through a period of dynamic change (particularly non-cyclical variations and trend reversals after sample index 10), the prediction values don't take into account non-linear dependencies, which means they stay steady over time and don't match the actual stock price patterns at all. This shows that ARIMA is not good at dealing with "non-cyclical fluctuations and complex sequence dependencies" in financial time series data.

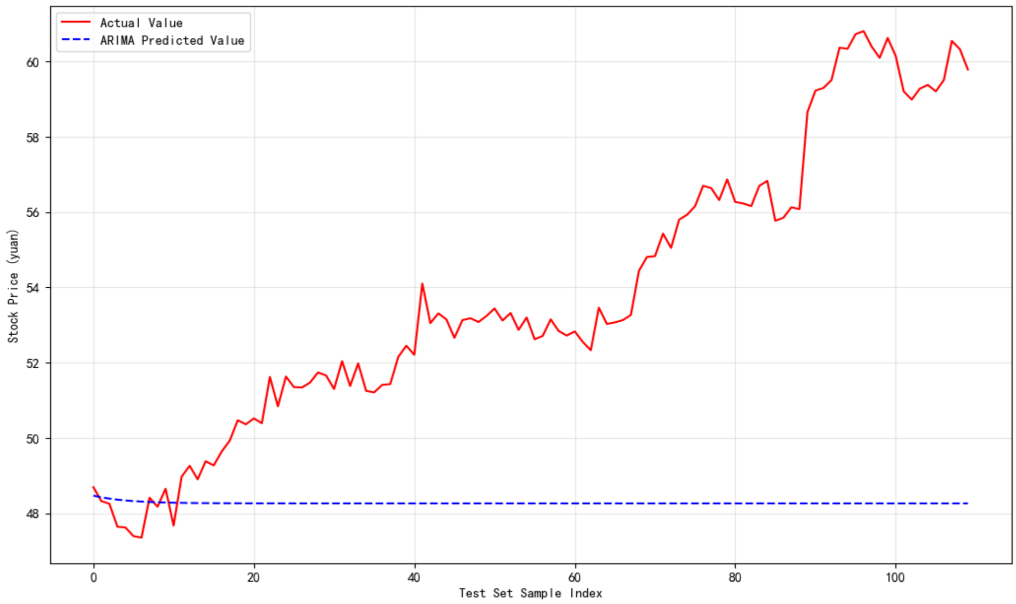

Fig. 7 ARIMA Model prediction result (Picture credit: Original)

The comparison shows that ARIMA, which is a standard time series model, has a linear framework and a fixed-order autocorrelation assumption. This makes it hard to discern long-term non-cyclical trends in stock prices and makes it impossible to respond to short-term changes. Because of this, its overall fitting performance is very different from real trends. Random Forest, which is built on an ensemble learning framework, can figure out the general direction of stock prices. However, its model structure makes it less flexible when it comes to time series features, which means it doesn't fit short-term changes as well. LSTM, on the other hand, uses gating mechanisms to look more closely at the time dependencies in stock price sequences. This makes it better at fitting both trends and volatility. This difference confirms that time series-specific models (like LSTM) are inherently suitable for financial time series data and gives a clear reason for choosing and improving stock price prediction models. For stock price data that is very dynamic and strongly dependent, models that can capture complex time series relationships should be given priority. Table 1 compares the results of different models.

Table 1. Detail data

Model	MSE	MAE	MAPE	R^2
LSTM	0.3542	0.4661	0.86%	0.9758
RandomForest	0.6720	0.6293	1.16%	0.9523
ARIMA	44.956	5.5626	10.04%	-2.1936

4. Discussion

4.1. Theoretical Explanation of Research Findings

Three theoretical variables can explain why LSTM models do better at predicting stock prices than other models.

The input gate, forget gate, and output gate structure of LSTM solves the gradient vanishing problem that happens in regular RNNs. This lets LSTM keep important information in stock price sequences for a long time (like how past highs and lows affect current prices) while forgetting short-term noise. This is quite similar to the market rule that "historical prices have a lagged effect on future trends," which shows that deep learning models can find long-term relationships in time series data.

Multi-feature synergy effect: The characteristics chosen for this study, like opening price and trading volume, show information on market supply and demand and trading activity. According to the efficient market hypothesis, prices change because investors trade based on this knowledge that is available to everyone. The LSTM model transforms the implicit connections among various characteristics into predictive signals by nonlinear fitting, reflecting the theoretical principle that “price is a collection of information” [14].

Behavioral finance provides an additional viewpoint: investor sentiment influences stock price variations alongside fundamental factors. The LSTM model's acute detection of features like irregular trading volume variations and transient price surges and declines indirectly illustrates the transmission mechanism of market sentiment, corresponding with Kahneman's prospect theory that “investors' asymmetric reactions to losses and gains exacerbate price volatility” [15], thereby offering empirical validation for behavioral finance theory.

4.2. Practical Implications

For individual investors: The LSTM model's short-term predictions can help you decide whether to buy and sell stocks, especially when the market is unstable (like it was from 2014 to 2016). By integrating the model's study of trade volume with the RSI indicator, we can lessen the impact of emotions. However, it's crucial to remember that model forecasts can be wrong and shouldn't be the only thing you rely on. To reduce risk, it is best to use both "prediction signals" and "stop-loss strategies."

For institutional investors: Multi-feature LSTM models can be added to quantitative trading systems to improve arbitrage tactics by taking advantage of their capacity to find temporal relationships. But you should be careful about the risks of overfitting. You can improve generalization by broadening the sample range (for example, by adding data from stocks in different industries) or by making regularization stronger to keep the model from failing in live trading because of "over-optimization of historical data."

For regulatory authorities: Studies show that market sentiment, as shown by trading volume and RSI indicators, has a big effect on how volatile stock prices are. So, to cut down on insider trading and the spread of misleading information, we need to make information disclosure rules stronger. This will help investors make smart choices. Also, LSTM models can be used to keep an eye on unusual trading patterns, like short-term rapid drops in volume, to give early warnings of systemic concerns.

4.3. Research Limitations and Future Prospects

This study has certain limitations:

Single feature dimension: The exclusion of macroeconomic indicators like the GDP growth rate and interest rates, as well as firm fundamentals data such as net profit and price - earnings ratio, is a significant shortcoming. By leaving out these factors, it's likely that vital elements influencing stock prices have been omitted. These macro - and micro - economic factors usually play a crucial part in determining stock price fluctuations, and their absence may curtail the comprehensiveness of the analysis.

Limited sample coverage: The study's scope is restricted to A - share market data from 2013 to 2018. This limited time span and market focus pose limitations. The analysis doesn't include data from other major markets, such as those in the US and Hong Kong. Moreover, extreme market conditions like the market shock caused by the pandemic in 2020 were not taken into account. Consequently, the generalizability of the model remains untested. To be certain of the model's efficacy across different market situations and regions, a more diverse and extensive dataset, covering various market conditions, would be required.

The current LSTM model does not take into account the changes in the weights of different features, which means it doesn't explain "which features are more important in certain market conditions" well enough.

Future research may be augmented in three domains:

Add more data to the feature system by combining macroeconomic, corporate fundamentals, and market sentiment data to create a "multi-source information fusion" prediction framework. This will make the model better at explaining complex markets.

Improve the model structure by adding an attention mechanism that lets the model automatically learn how important different features and time steps are. This will make the model more predictable. Alternatively, try a hybrid model of LSTM and Transformer to make it even better at capturing long-term dependencies.

Cross-market and extreme market conditions testing: Add major global stock markets to the sample to **see** how the model works in different markets. At the same time, add data from extreme market conditions to see how well the model holds up during black swan events. This will teach you how to manage risk across markets.

5. Conclusion

This study examines the A-share market index from the Kaggle dataset “S&P 500 stock data” spanning 2013 to 2018, establishing a comparative experiment between standard models and multi-feature fusion, while thoroughly investigating the efficacy of LSTM in stock price prediction. The primary conclusions are as follows:

The LSTM model's regression metrics, like MSE, MAE, and MAPE, are much lower than those of traditional models when it comes to predicting short-term stock price trends (the closing price for the next day). The R^2 value is close to 1. In the binary classification task of predicting stock price rises or falls, the LSTM model showed outstanding results, boasting high AUC and F1 scores. From the radar charts, it's clear that its overall performance surpasses that of models like logistic regression and random forest. This is particularly evident in its capacity to capture complex non - linear relationships and both long - and short - term temporal associations. The LSTM model takes into account features such as opening price, high price, low price, closing price, and trading volume. This multi - feature inclusion offers a more thorough understanding of market operations compared to single - feature models. It implies that technical indicators can contribute to stock price prediction, and that the movement of stock prices is a consequence of the interplay of various elements. Additionally, The LSTM model also showed relatively stable changes in prediction error under different market conditions, such as the upward trend from 2013 to 2014, the volatile period from 2014 to 2016, and the recovery phase from 2016 to 2018. This shows that it is better able to adapt to complex and changing market conditions. This gives investors a steady base to make decisions in different market scenarios.

References

[1] Gandhmal, D. P., Kumar, K. Systematic analysis and review of stock market prediction techniques. Computer Science Review, 34:100190 (2019).

[2] Di Persio, L., Honchar, O. Recurrent Neural Networks Approach to the Financial Forecast of Google Assets. International Journal of Mathematics and Computers in Simulation, 11:7-13 (2017).

[3] Liu, G. Q. Research on the application of the nonlinear GARCH model in predicting Chinese stock market activities. Statistical Research, 2000 (1):49-52 (2000).

[4] Shi, Z. MambaStock: Selective state space model for stock prediction. arXiv (2024). https://doi.org/10.48550/arXiv.2402.18959

[5] 5) Shi, T. Study of short-term load forecasting based on random forest regression algorithm. Master's thesis, Zhengzhou University (2017).

[6] Hu, D. G., Ma, Y., Pang, X. D., Wu, F., Niu, Z., Li, H., Ji, Y. Q., Feng, W. T. New energy power generation prediction based on the CNN-LSTM-Attention model and risk detection analysis of isolation forest algorithm. Journal of Image and Signal Processing, 14 (1):45-61 (2025). https://doi.org/10.12677/jisp.2025.141005

[7] Hochreiter, S., Schmidhuber, J. Long Short-term Memory. Neural Computation, 8:1735-1780 (1997).

[8] Gers, F. A., Schmidhuber, J., Cummins, F. Learning to forget: Continual prediction with LSTM. In Artificial Neural Networks (pp. 850-855). IEE (1999). https://doi.org/10.1049/cp:19991117

[9] Jiang, S. Y. Stock price prediction based on LSTM model. Jiangsu Commercial Forum, 2025 (1):83-86 (2025). https://doi.org/10.13395/j.cnki.issn.1009-0061.2025.01.013

[10] Murphy, J. J. Technical analysis of the financial markets. Seismological Press (1999).

[11] Chou, J. S., Nguyen, T. K. Forward Forecast of Stock Price Using Sliding-Window Metaheuristic-Optimized Machine-Learning Regression. IEEE Transactions on Industrial Informatics, 14 (7):3132-3142 (2018).

[12] Yang, Q., Wang, C. A study on forecast of global stock indices based on deep LSTM neural networks. Statistical Research, 36 (3):65-78 (2019). https://doi.org/10.19343/j.cnki.11-1302/c.2019.03.006

[13] Xu, Q. H. A Transformer stock price prediction model based on CNN-GRU. Master's thesis, Jiangxi University of Finance and Economics (2025). CNKI.

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

[15] Kahneman, D., Tversky, A. Prospect theory: An analysis of decision under risk. Econometrica, 47 (2):263–291 (1979).