Dean&Francis

ISSN 2959-6130

Stock Price Forecasting Using Deep Learning: A Comparative Study of CNN, LSTM, and Transformer Models with Feature Engineering Insights

Jiyu Ning

South China University of Technology, Guangzhou, Guangdong, 510641, China
202230322062@mail.scut.edu.cn

Abstract:

Accurate forecasting of stock prices remains a challenging yet essential task in financial modeling. This paper investigates the effectiveness of deep learning architectures—convolutional neural networks (CNN), long short-term memory networks (LSTM), and transformer models—for predicting the average stock price over a 30-day horizon. Historical market data from 12 representative Hong Kong-listed companies are employed to evaluate the performance of each model, both with and without the inclusion of traditional technical indicators (Type 1) and derived statistical features (Type 2). Results show that the Transformer model trained on raw price and volume data achieves superior performance, recording the lowest Root Mean Square Error (RMSE = 0.3799), Mean Absolute Error (MAE = 0.2909), and the highest directional accuracy (51.00%), outperforming other models and exceeding the random baseline. In contrast, the integration of technical indicators results in decreased performance, suggesting potential overfitting or feature redundancy. The research underscores the effectiveness of attention-based architectures in financial time series prediction and emphasizes the importance of cautious feature selection when incorporating domain-specific indicators. These findings provide practical guidance for quantitative analysts and financial data scientists regarding model architecture choices and feature engineering strategies in stock forecasting tasks.

Keywords: LSTM, Transformer, CNN, Stock Forecasting, Feature Engineering

1 Introduction

Forecasting trends in the financial market remains an important yet complex area for researchers and industry professionals, given its significance for strategic investments and economic decision-making. In recent years, deep learning (DL) models have gained considerable popularity due to their powerful ability in handling intricate temporal relationships and capturing nonlinear dynamics inherent in financial datasets.

Bansal et al. explored various DL techniques specifically for predicting stock movements, underscoring the advantages of these methods in financial forecasting tasks [1]. Additionally, Yang and Wen combined DL approaches with sentiment analysis, demonstrating notable predictive improvements for stock market analysis [2]. Moreover, a combined approach using LSTM and CNN architectures was proposed by Wang and Li, effectively leveraging the sequential modeling strengths of LSTM along with CNN's capacity to extract local patterns, resulting in enhanced prediction accuracy [3].

LSTM networks have been widely recognized for their proficiency in modeling sequential data due to their unique memory mechanisms. Hochreiter and Schmidhuber initially introduced LSTM networks, which have since become foundational in handling long-term dependencies in financial data [4]. Jiang et al. further explored the use of LSTM specifically in stock market forecasting, demonstrating its effectiveness in capturing complex temporal dynamics [5].

Linear regression is a commonly used fundamental analytical tool in financial forecasting; however, Singh and Srivastava indicated it typically struggles to effectively model the complex and dynamic characteristics of financial markets [6]. Shaikh et al. carried out a comparative study examining traditional statistical methods alongside machine learning techniques, ultimately suggesting that machine learning methods generally yield better accuracy and robustness in predicting stock price movements [7]. Additionally, Vaswani et al. proposed the Transformer architecture featuring a self-attention mechanism, substantially enhancing the modeling of long-range dependencies within sequential data [8]. Hu further explored various deep learning techniques, including the Transformer, emphasizing their strong potential for capturing sophisticated market behaviors in financial prediction tasks [9].

This study contributes to the existing literature by systematically evaluating the performance of CNN, LSTM, and Transformer models specifically within the context of the Hong Kong stock market, providing insights into their respective strengths, limitations, and optimal usage scenarios.

2 Methodology

2.1 Data Collection and Preprocessing

The paper collects historical daily trading data (2019–2024) of 12 representative Hong Kong-listed companies via AkShare, including Close, High, Low, Shares_Traded, and Volume. The data set consists of daily trading records from 2019 to 2024 for 12 representative Hong Kong stocks, obtained via the Akshare financial database. Each record contains the basic K-line indicators: open, high, low, close, volume, and amount (Table 1).

Table 1. Selected Hong Kong stocks and their industry sectors

Stock code	Company name	Sector
00700.HK	Tencent Holdings Ltd.	Technology
03690.HK	Meituan	Consumer Services / E-commerce
01211.HK	BYD Company Limited	Automobiles / New Energy
00388.HK	Hong Kong Exchanges and Clearing	Financial Services
02382.HK	Sunny Optical Technology	Technology / Optical Equipment
02318.HK	Ping An Insurance	Insurance
00981.HK	Semiconductor Manufacturing Int'l Corp.	Semiconductors
06060.HK	CITIC Securities	Financial Services / Brokerage
06823.HK	EC Healthcare	Healthcare Services
01044.HK	Hengan International Group	Consumer Goods / Hygiene
01093.HK	CSPC Pharmaceutical Group	Pharmaceuticals
02020.HK	Anta Sports Products Limited	Consumer Goods / Apparel

To facilitate model training, all close price series are independently standardized using z-score normalization to eliminate scale differences between stocks. The standardized trends are visualized in Fig. 1, where each line rep-

ISSN 2959-6130

resents one stock.

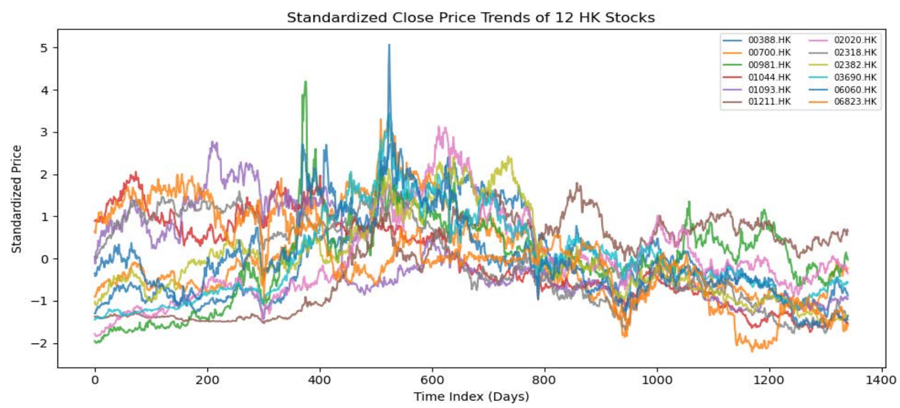


Fig. 1. Standardized Close Price Trends of 12 HK Stocks (Photo/Picture credit: Original).

From Fig. 1, Standardized Close Price Trends of 12 HK Stocks, the author observes that while all stocks fluctuate around zero (due to standardization), their volatility and local patterns differ significantly. Some exhibit sharp spikes (indicative of outlier movements), while others are relatively stable. This underscores the heterogeneity in price dynamics between companies, which deep learning models must take into account.

To enhance learning, the author computes two groups of derived features. The first group consists of technical indicators, commonly used in financial analysis: Type 1 features (technical indicators):

Moving Average (MA) over n days smooths the price trend by averaging the past n closing prices:

$$MA_n(t) = \frac{1}{n}\sum_{i=0}^{n-1} P(t-i) \tag{1}$$

where $P(t)$ is the price at time t. Rolling Standard Deviation (STD) captures recent volatility:

$$STD_n(t) = \sqrt{\frac{1}{n}\sum_{i=0}^{n-1}(P(t-i) - MA_n(t))^2} \tag{2}$$

It measures the dispersion of prices around the moving average. Relative Strength Index (RSI) is a momentum oscillator:

$$RSI(t) = 100 - \frac{100}{1 + RS(t)} \tag{3}$$

Where $RS(t) = \frac{EMA^+(t)}{EMA^-(t)}$ and EMA^+, EMA^- are the exponential moving averages of gains and losses, respectively. On-Balance Volume (OBV) reflects cumulative volume flow:

$$\mathrm{OBV}(t) = \begin{cases} \mathrm{OBV}(t-1) + V(t), & \text{if} P(t) > P(t-1) \\ \mathrm{OBV}(t-1) - V(t), & \text{if} P(t) < P(t-1) \\ \mathrm{OBV}(t-1), & \text{otherwise} \end{cases} \tag{4}$$

Where $V(t)$ is the trading volume at time ttt. Moving Average Convergence Divergence (MACD) is calculated as:

$$MACD(t) = EMA_{12}(t) - EMA_{26}(t) \tag{5}$$

With the signal line defined as $Signal(t) = EMA_9(MACD(t))$, and histogram: $Hist(t) = MACD(t) - Signal(t)$. Here, EMA_k denotes the exponential moving average over k periods.

$$\mathrm{OBV}(t) = \begin{cases} \mathrm{OBV}(t-1) + V(t), & \text{if} P(t) > P(t-1) \\ \mathrm{OBV}(t-1) - V(t), & \text{if} P(t) < P(t-1) \\ \mathrm{OBV}(t-1), & \text{otherwise} \end{cases} \tag{6}$$

Type 2 features (statistical descriptors over a 30-day window): To capture the statistical characteristics of stock prices, the following features are computed over a fixed-size window of $w = 30$ days: Trend Slope: The slope β_1 from linear regression over the price series is used to quantify the directional trend. It is calculated by fitting a line to the past www prices, where $\overset{\text{i¥}}{x}$ and $\overset{\text{i¥}}{P}$ are the means of the time indices and prices, respectively.

Mean and Standard Deviation: The average price $\mu(t)$ and volatility $\sigma(t)$ over the window are given by:

$$\mu(t) = \frac{1}{w}\sum_{i=0}^{w-1} P(t-i), \sigma(t) = \sqrt{\frac{1}{w}\sum_{i=0}^{w-1}(P(t-i) - \mu(t))^2} \tag{7}$$

where P(t) denotes the price at time t.

Maximum and Minimum Prices: The highest and lowest prices within the window are:

$$P_{\max}(t) = \max_{0 \leqslant i < w} P(t-i), P_{\min}(t) = \min_{0 \leqslant i < w} P(t-i) \tag{8}$$

Amplitude (Range): Defined as the price range in the window:

$$A(t) = P_{\max}(t) - P_{\min}(t) \tag{9}$$

Skewness and Kurtosis: These higher-order moments capture the asymmetry and tail behavior of the price distribution:

$$\text{Skew}(t) = \frac{1}{w}\sum_{i=0}^{w-1}\left(\frac{P(t-i)-\mu(t)}{\sigma(t)}\right)^3, \text{Kurt}(t) = \frac{1}{w}\sum_{i=0}^{w-1}\left(\frac{P(t-i)-\mu(t)}{\sigma(t)}\right)^4 \tag{10}$$

All features are standardized per stock using:

$$x_{i}^{-} = \frac{x-\mu}{\sigma} \tag{11}$$

This standardization is applied after feature computation to preserve temporal dynamics while ensuring comparability across stocks.

2.2 LSTM Model

The LSTM architecture is designed to process sequential data by preserving and updating memory states over time. It effectively learns long-distance temporal relationships while reducing issues such as the vanishing gradient, which often hinder standard recurrent networks. Through sequential processing of historical stock data, LSTM selectively retains important past signals and filters out noise, supporting the learning of time-based patterns essential for prediction. Its strength lies in modeling nonlinear and intricate temporal behaviors within financial datasets, making it a robust candidate for stock market forecasting applications.

2.3 CNN Model

CNNs extract local temporal patterns from time-series data by applying convolution operations across input sequences, enabling efficient feature learning without relying on sequential processing. Through stacked convolution and pooling layers, CNNs summarize short-term dependencies and highlight significant local trends in historical stock data. Their primary advantage lies in fast training, efficient parallel computation, and effective capture of short-term patterns, making them suitable for tasks where recent fluctuations carry critical predictive information.

2.4 Transformer Model

Originally developed for handling sequential data, the Transformer architecture has since been adapted for time-series analysis, primarily due to its strength in modeling distant temporal relationships through self-attention mechanisms. By dynamically weighting different time steps, the model can extract meaningful temporal patterns from the entire historical dataset without relying on recurrent structures. Its main advantages include parallel processing and the ability to capture global dependencies, making it especially suitable for tasks involving complex temporal sequences where long-term historical trends are essential for precise prediction. Transformer+Type1: Uses technical indicators. Transformer+Type2: Uses statistical pattern descriptors [10].

3 Results

This section visualizes and interprets the training dynamics and prediction performance of the proposed models, including LSTM, CNN, and Transformer variants. To evaluate model performance on stock trend forecasting, three key metrics are reported: RMSE, MAE, and directional accuracy (trend prediction accuracy). The summary of results is presented in Table 2.

Among the tested models, the Transformer, when applied using only raw input features without any additional preprocessing or engineered indicators, produced the most favorable outcomes. It recorded the lowest root mean square error (RMSE) and mean absolute error (MAE), while also reaching a directional accuracy of 51.00

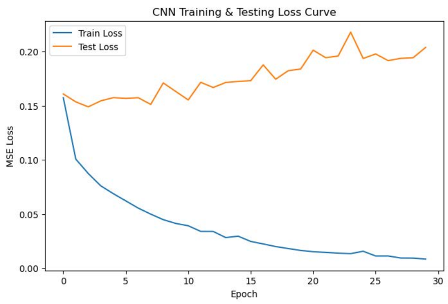
The LSTM model performed reasonably well and showed a level of stability across its predictions. However, when measured against the Transformer, it fell short by a small margin. This gap may be attributed to the way LSTM handles data in a step-by-step sequence, which could limit its ability to recognize patterns spread over longer time intervals.

Table 2. Performance metrics

Model	RMSE	MAE	Direction Acc (%)
CNN	0.4538	0.3432	50.03
LSTM	0.4016	0.2940	50.86
Transformer	0.3799	0.2909	51.00
Transformer + Type1	1.2234	0.9952	49.87
Transformer + Type2	0.7007	0.5321	50.20

ISSN 2959-6130

3.1 Training and Testing Loss Curves

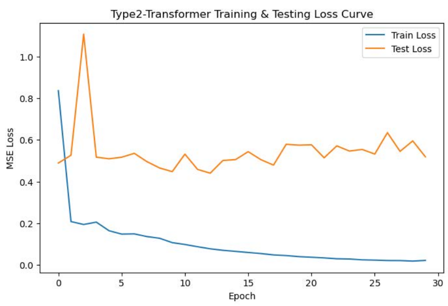


(a)CNN Training and Testing loss Curve

(b)LSTM Training and Testing loss Curve

(c) Transformer Training and Testing loss Curve

(d)Type1-Transformer Training and Testing loss Curve

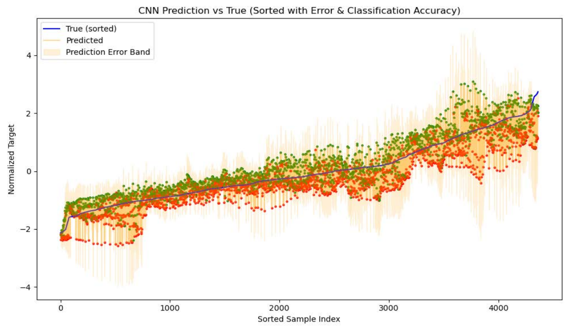


(e)Training and Testing loss Curve of Type2-Transformer

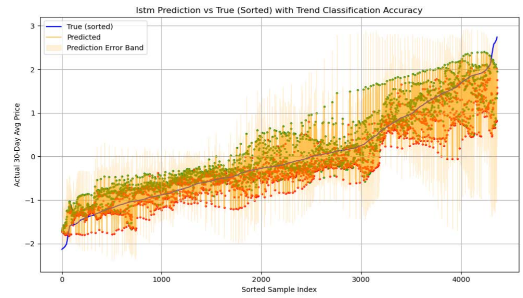
Fig. 2. Training and testing curve (Photo/Picture credit: Original).

Fig. 2 illustrates the curves representing the MSE loss values throughout the training and testing phases. The training loss is represented by the blue line, while the testing loss is shown by the orange line. A well-generalizing model should exhibit convergence of both curves with minimal overfitting, whereas divergence suggests poor generalization. Notably, the LSTM and Transformer models (top-right and middle-left, respectively) demonstrate relatively stable convergence.

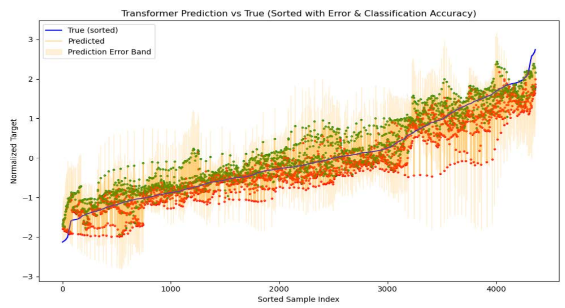
3.2 Prediction Performance and Classification Visualization



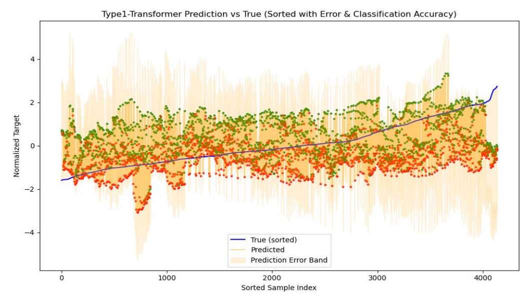
(a)CNN



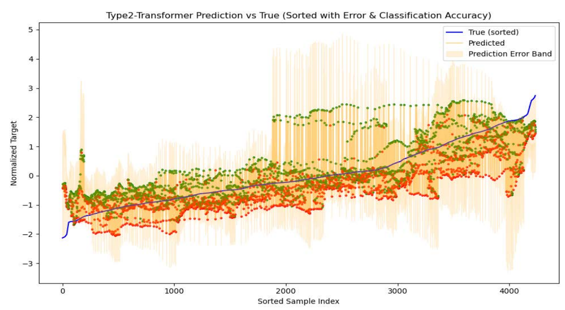
(b)LSTM



(c)Transformer



(d)Type-1 Transformer



(e)Type-2 Transformer

Fig. 3. Prediction Curve of 5 Models (Photo/Picture credit: Original).

Fig. 3 presents the sorted prediction results for each model, overlaid with the true (ground truth) target curve. The y-axis represents the normalized 30-day average target (i.e., the prediction target), while the x-axis denotes the sorted test sample index.

4 Conclusion

This research provides a comparative analysis among three deep learning architectures—CNN, LSTM, and Transformer—evaluating their capabilities in forecasting stock trends within the Hong Kong market. Experimental outcomes demonstrate superior predictive effectiveness of the Transformer architecture when utilizing basic market data, achieving notably low forecasting errors (RMSE = 0.3799, MAE = 0.2909) alongside improved accuracy in predicting market directions (51.00%). Conversely, CNN and LSTM architectures showed comparatively inferior predictive results.

Analyzing the strengths and weaknesses of each model, the Transformer's self-attention mechanism enables it to effectively capture long-range dependencies and global patterns, which is particularly advantageous in sequential prediction tasks. CNN models are effective at capturing local patterns but typically lack explicit mechanisms for modeling long-range temporal dependencies. LSTM models, known for their gated mechanisms, excel in capturing temporal dependencies and have demonstrated high accuracy in short-term stock price predictions.

ISSN 2959-6130

Regarding model optimization strategies, this study found that directly incorporating traditional technical indicators (Type 1) or statistical features (Type 2) did not enhance performance and instead introduced redundancy or noise, negatively affecting model accuracy. This highlights the importance of carefully selecting relevant features to enhance predictive performance.

This study also has limitations in its experimental design. The dataset used was limited to 12 companies listed in the Hong Kong market, potentially restricting the generalizability of the findings. Additionally, external economic and non-economic factors, recognized as critical influencers on stock prices, were not extensively explored, although these factors significantly impact real-world market conditions.

Future research could expand dataset scale and market diversity, integrate broader economic and non-economic indicators, and explore hybrid model architectures. Combining different neural network architectures, such as hybrid CNN-LSTM models, shows promise for further improving prediction accuracy. Furthermore, investigating flexible time windows and diverse forecasting tasks could contribute to developing more robust and generalizable stock market prediction models.

In conclusion, this study confirms the superior performance of the Transformer model in stock trend prediction and highlights the importance of careful feature selection and model architecture design, offering valuable guidance for developing and applying future financial forecasting models.

References

1. Bansal, M., Goyal, A., Choudhary, A.: Stock market prediction using deep learning. International Journal of Creative Research Thoughts (2022)

2. Yang, Y., Wen, J.: Forecasting Chinese stock market using deep learning and sentiment analysis. Humanities and Social Sciences Communications 11(1), 104 (2024)

3. Wang, H., Li, M.: Hybrid LSTM-CNN model for stock market prediction. Humanities and Social Sciences Communications 12(1), 129 (2025)

4. Hu, Y.: Deep learning based stock market prediction and analysis. SSRN Electronic Journal (2018)

5. Jiang, Q., Wang, X., Chen, Z.: Stock market prediction based on LSTM neural network. Procedia Computer Science 199, 1043–1050 (2022)

6. Singh, A.K., Srivastava, A.: Stock market analysis using linear regression. International Journal of Computer Applications 183(45), 25–28 (2022)

7. Shaikh, M., Shaikh, S.A., Talha, M.: Comparative analysis of machine learning and deep learning approaches for stock price prediction. Procedia Computer Science 200, 529–536 (2022)

8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation 9(8), 1735–1780 (1997)

9. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., et al. (eds.) Advances in Neural Information Processing Systems, vol. 30. Curran Associates, Red Hook (2017)

10. Khaniki, M. A. L., & Manthouri, M.: Enhancing price prediction in cryptocurrency using transformer neural network and technical indicators. arXiv preprint arXiv:2403.03606 (2024)