mathematics

Article

CNN-CBAM-LSTM: Enhancing Stock Return Prediction Through Long and Short Information Mining in Stock Prediction

Peijie Ye †, **Hao Zhang** † and **Xi Zhou** *

Jiyang College, Zhejiang A&F University, Shaoxing 311800, China; 18861299777@163.com (P.Y.); zh8134717@gmail.com (H.Z.)
* Correspondence: zhouxi654321@126.com; Tel.: +86-180-6767-9115
† These authors contributed equally to this work.

Abstract: Deep learning, a foundational technology in artificial intelligence, facilitates the identification of complex associations between stock prices and various influential factors through comprehensive data analysis. Stock price data exhibits unique time-series characteristics; models emphasizing long-term data may miss short-term fluctuations, while those focusing solely on short-term data may not capture cyclical trends. Existing models that integrate long short-term memory (LSTM) and convolutional neural networks (CNNs) face limitations in capturing both long- and short-term dependencies due to LSTM's gated transmission mechanism and CNNs' limited receptive field. This study introduces an innovative deep learning model, CNN-CBAM-LSTM, which integrates the convolutional block attention module (CBAM) to enhance the extraction of both long- and short-term features. The model's performance is assessed using the Australian Standard & Poor's 200 Index (AS51), showing improvement over traditional models across metrics such as RMSE, MAE, R^2, and RETURN. To further confirm its robustness and generalizability, Diebold–Mariano (DM) tests and model confidence set experiments are conducted, with results indicating the consistently high performance of the CNN-CBAM-LSTM model. Additional tests on six globally recognized stock indices reinforce the model's predictive strength and adaptability, establishing it as a reliable tool for forecasting in the stock market.

Keywords: prediction; deep learning; stock prices; CNN; CBAM; LSTM

MSC: 68T07; 68T09; 62P05

Citation: Ye, P.; Zhang, H.; Zhou, X. CNN-CBAM-LSTM: Enhancing Stock Return Prediction Through Long and Short Information Mining in Stock Prediction. *Mathematics* **2024**, *12*, 3738. https://doi.org/10.3390/math12233738

Academic Editor: Anatoliy Swishchuk

Received: 11 October 2024
Revised: 20 November 2024
Accepted: 27 November 2024
Published: 27 November 2024

1. Introduction

In recent years, amid fluctuating financial markets and economic shifts, the gradual decline in bank deposit interest rates has prompted a surge of investors into the stock market [1]. As a result, accurately predicting stock prices has become a central focus of investor discourse and a significant challenge in this field [2]. This study presents a new algorithmic model for stock price forecasting at the theoretical level, aimed at enriching and advancing the relevant theoretical framework. On a practical level, this research aims to enhance investors' decision-making efficiency and investment returns. By developing reliable stock price forecasting models, investors can more precisely assess market risks and opportunities, enabling them to formulate smarter investment strategies and achieve improved returns on their investments.

The stock market operates within a dynamic and intricate economic system [3]. Stock prices are influenced by a multitude of factors, including a company's financial metrics, industry outlook, management practices, governance, and broader macroeconomic climate. Additionally, irrational elements such as market sentiment contribute to complexity, rendering stock prices highly nonlinear and stochastic and challenging to predict [4]. Early approaches predominantly relied on statistical and econometric methods such as moving averages and regression analysis. While these methods capture market trends to some extent, their capacity to handle nonlinear data is limited. With the advancement of machine-learning technology, data-driven approaches have emerged as effective tools for prediction across various domains [5]. Machine-learning algorithms such as support vector machines, random forests, and neural networks are used to construct prediction models. These algorithms can handle complex nonlinear relationships, but they usually require considerable feature engineering and are very sensitive to the selection of model parameters [6]. In recent years, deep-learning techniques have exhibited substantial promise across diverse domains, notably in areas such as image recognition [7], natural language processing [8], and reinforcement learning [9]. In financial technology, deep-learning models, particularly hybrid models that combine LSTM and CNN, have proven effective for capturing both long- and short-term dependencies within time-series data. The inclusion of attention mechanisms, such as the convolutional block attention module (CBAM), further enhances these models by facilitating adaptive feature refinement, a key advantage in the highly variable environment of financial forecasting [10,11]. As a result, deep-learning-based hybrid models incorporating attention mechanisms have gained significant research interest, offering a robust approach to managing the complexities inherent in stock market data.

This paper proposes a new method, CNN-CBAM-LSTM. This method uses multiple attention mechanisms and a CNN as its main component. The multi-head attention mechanism and CBAM capture the time correlation in the stock time series, whereas the CNN integrates the characteristics of stock data. CNN-CBAM-LSTM addresses the issue of feature extraction existing in traditional machine learning. There is no manual extraction from the data input to the prediction, which shortens the prediction time. This research utilizes Australia's Standard & Poor's 200 (AS51) daily trading data from 4 January 2005, to 26 April 2024, with the first 70% allocated for training and the remaining 30% allocated for testing.

This study makes the following contributions.

- Innovative Model Structure: The proposed model introduces a novel architecture capable of capturing stock information across different time scales, resulting in improved prediction accuracy. The model demonstrates a higher rate of return, with the experimental results confirming the significance of these performance gains.
- Efficient performance with CBAM: CBAM enhances the model performance while maintaining minimal additional computational costs, offering a balanced trade-off between efficiency and effectiveness.

In Section 3, the proposed model is introduced in detail. In Section 4, this study compares LSTM, CNN-LSTM, CNN-Bi-LSTM, random walk, random forest, and SVM across various stock datasets. Section 5 summarizes the advantages of the proposed deep model over existing models across different stock datasets and discusses its limitations.

2. Related Work

2.1. Challenges and Importance of Stock Price Prediction

The complexity and high volatility of financial markets pose significant challenges for stock price prediction. These challenges stem from numerous nonlinear and stochastic factors, including economic indicators, geopolitical events, and fluctuations in investor sentiment [12]. While traditional machine-learning models offer some predictive value, their accuracy is limited due to the high dimensionality and nonlinear nature of financial data [13]. In this context, recent advancements in deep learning, particularly through hybrid models and attention mechanisms, have introduced promising approaches to overcoming these challenges. By effectively capturing both short- and long-term dependencies and emphasizing essential features within complex datasets, deep-learning methods, such as the integration of CNN-LSTM with CBAM attention modules, have become widely utilized in financial forecasting. These methods not only enhance the prediction stability and accuracy but also provide valuable support for financial decision-making and risk management.

2.2. Hybrid Deep-Learning Models in Financial Forecasting

With the continuous improvement in the international financial system and the advent of the era of big data, machine-learning models have become one of the main analysis methods for stock forecasting. Machine-learning methods, such as neural networks, support vector machines, and random forests, are used to predict stock prices [14].

Deep learning represents a more advanced branch of machine learning, which is particularly dominant in predictive tasks involving random processes. It offers clear advantages in handling complex challenges such as image recognition and natural language processing. Minjun Kim et al. [15] verified its efficacy by constructing a complex network time series of the S&P 500 index, demonstrating that integrating network metrics into the ARIMA model enhances prediction accuracy. Lei Lin et al. [16] utilized an RNN model to develop a three-layer neural network to predict the Shanghai Stock Exchange Index's stock prices and analyze its convergence speed and feasibility. Daiyou Xiao et al. [17] introduced an ARIMA-LSTM hybrid model, which was evaluated using New York Stock Exchange data from 2010 to 2019. The model achieved a strong performance across three datasets, with MSE = 0.103, RMSE = 0.320, and MAE = 0.250 on the Develop set, highlighting its robustness and potential applicability for portfolio optimization. Patra et al. [18] constructed a hybrid LSTM-GRU model to predict the adjusted closing price of the S&P 500 index, which demonstrated an improved accuracy over baseline models in experimental trials. Shahzaheer et al. [19] adopted five deep-learning hybrid models and used six features of the Shanghai Composite Index to verify their effects. Through the experimental performance analysis, the single-layer RNN model is found to be superior to all other models. This finding verifies the feasibility of the model and helps investors make decisions at the right time to maximize profits.

Researchers such as Jithin Eapen [20] introduced a hybrid model that combines CNNs and Bi-LSTM, which achieves a 9% performance improvement over single-network models. Konark Yadav et al. [21] proposed FastRNN+CNN+Bi-LSTM, a high-speed, accurate hybrid deep-learning model that outperforms ARIMA, FBProphet, LSTM, and other hybrids in terms of the RMSE and calculation time. Jilin Zhang et al. [22] developed the CNN-BiLSTM-Attention model to overcome classical time-series prediction limitations, demonstrating its stability and effectiveness across three Chinese and eight international stock indices, outperforming the LSTM, CNN-LSTM, and CNN-LSTM-Attention models in accuracy. Man Li et al. [23] presented a clustering-enhanced deep-learning framework that uses logistic weighted dynamic time warping (LWDTW) to improve the stock similarity measurement and clustering effects. Their experimental results underline the practical significance of aiding optimal investment decisions. Junji Jiang et al. [24] proposed HMM-ALSTM, a model that combines hidden Markov models (HMMs) and attention-LSTM (ALSTM), which enhances the daily stock market status exploration and price estimation efficiency and generalizability through extensive experimentation.

Chaojie Wang et al. [25] used the transformer model to predict global stock indices, outperforming traditional models in terms of accuracy and returns by leveraging the encoder–decoder architecture and multi-head attention. Fuwei Yang et al. [26] proposed a stock price prediction model using an improved particle swarm optimization (PSO) algorithm with adaptive inertia weights, which demonstrated a strong performance in handling high-noise, nonlinear data. Jujie Wang et al. [27] developed a GRU-based stock selection model optimized with cuckoo search, integrating financial, technical, and sentiment factors, achieving a Sharpe ratio of 127.08% and an annualized return of 40.66%. Jinghua Zhao [28] applied attention-enhanced models (AT-RNN, AT-LSTM, and AT-GRU) for stock price prediction, with GRU and LSTM outperforming RNN, while attention mechanisms further improved its accuracy. Xuan Ji and colleagues [29] combined social media and financial data in a deep-learning model (Doc2Vec, SAE, and LSTM) with a wavelet transform for noise reduction, outperforming traditional models in capturing investor sentiment. Hadi Rezaei [30] proposed a CEEMD-CNN-LSTM hybrid model that excelled in handling complex financial data. Yu Lin and team [31] developed a CEEMDAN-LSTM model, which outperforms the SVM and BP models in terms of accuracy and robustness for S&P500 and CSI300 index prediction, confirming its strong potential for trend forecasting.

Deep learning outperforms traditional machine learning by effectively handling high-dimensional data and uncovering relevant information and patterns. It excels in prediction tasks, demonstrating superior accuracy, recall ability, robustness, and computational efficiency. Deep learning significantly enhances accuracy and applicability in classification tasks, driving advancements in mathematical and quantitative modeling [32].

3. Proposed Methodology

The structure of the proposed stock price forecasting method is depicted in Figure 1. It comprises a short-time-series information mining model, a long-time-series information mining model, and a linear layer. Given that stock information is typical time-series data, extracting insights solely from long sequences may overlook short-term data coupling, whereas focusing solely on short-term sequences may somewhat decrease periodic stock information characteristics. To balance these aspects, this paper proposes a deep-learning model that integrates both long- and short-time-series information mining to address these aspects that are unique to stocks. Section 3.1 details the extraction of stock data features, while Sections 3.2 and 3.3 introduce the short- and long-term time-series information mining models, respectively.

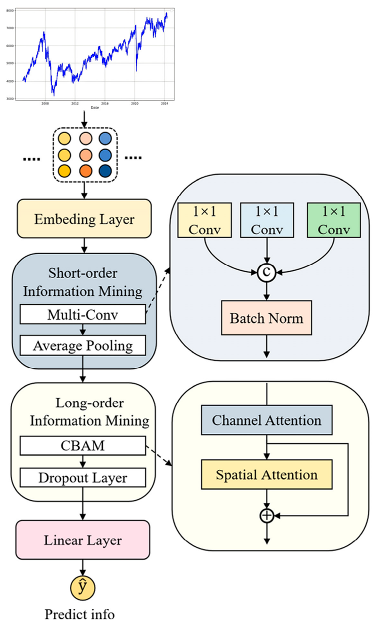

Figure 1. Architecture of the method proposed in this paper.

The method consists of an embedding layer, short-term sequence information mining, long-term sequence information mining, and a linear layer. The embedding layer enhances the relevance of stock information characteristics. The long- and short-time-series information mining models extract feature information from stock data across varying time scales. Finally, the linear regression model uses these extracted features for prediction and regression.

3.1. Data Features

3.1.1. Sliding Time Window Construction

The full-sequence stock price information cannot be directly input into the neural network. Given the variability of stock data over time, the most prevalent method for predicting the data for the first $T + 1$ day is to construct a sliding time window, utilizing the historical data from the past T days for prediction. Using these stock data characteristics of different subtime domains, the input data can be constructed as follows:

$$R = \left[R^1, R^2, \cdots, R^n\right] \tag{1}$$

where R^i represents the stock data information of the time length of T acquisition for the ith time. When F is used to represent the number of input factors, the formula for R^i is as follows:

$$R^i = \begin{bmatrix} p_{1,1} & \cdots & p_{1,F} \\ \vdots & \ddots & \vdots \\ p_{T,1} & \cdots & p_{T,F} \end{bmatrix} \tag{2}$$

3.1.2. Data Distribution Analysis

To comprehensively examine the distribution characteristics of the dataset, histograms and boxplots were generated for each feature (e.g., open, close, high, and low), as shown in Figure 2. The histograms illustrate the overall distribution and potential skewness of each variable, while the boxplots highlight any outliers. This analysis provides insight into inherent biases or skewness within the dataset, ensuring that the features are accurately represented for predictive modeling purposes.

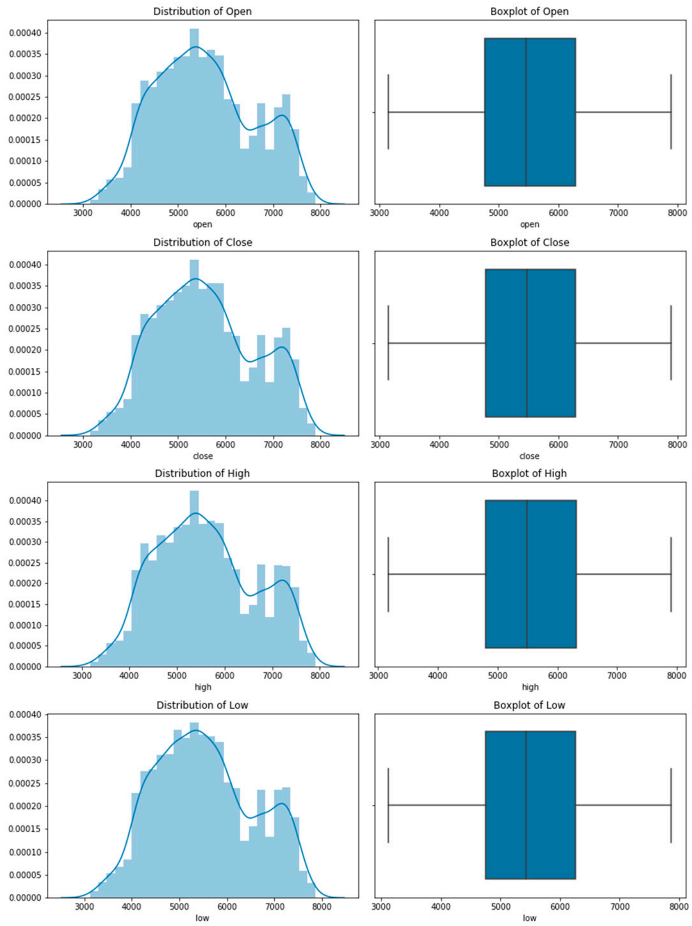

Figure 2. Feature distribution and outlier analysis for stock data.

3.2. Short-Order Information Mining

Stock series are a typical type of time-series information. A recurrent neural network (RNN) [33] structure is considered a mainstream solution for extracting the context features of sequence information. However, RNNs and their derivative structures (such as long short-term memory [LSTM]) [34], which are mostly used in this field to memorize time series, still have difficulty solving the problem of gradient disappearance. In this work, convolution is utilized to extract information from various sequences effectively. This approach captures local features without over-relying on long-term temporal information, thereby avoiding issues like gradient vanishing or exploding. Convolution also enhances parallel computing capabilities, speeding up both training and inference. Specifically, 1D convolution is applied to extract temporal features and identify local patterns and dependencies within sequences. A sliding window mechanism is logically constructed by stacking multiple convolution kernels of different sizes (e.g., 3, 5, and 7).

The convolution process, given W_d as the convolution kernel weight, t as the input dimension, and u as the output dimension, is expressed as follows:

$$O^i(t,u) = \sum_{i=1}^{k} W_d(u,i)\cdot\left(R^i\right)^T\left(\left[t+i-\frac{[k+1]}{2}\right],u\right) \tag{3}$$

where k is the size of the convolution kernel and d is the number of output channels. The output results are subsequently spliced together, and the formula is as follows:

$$O = concat\left(O^1, O^2, \ldots, O^k\right) \tag{4}$$

The data are input into the batch norm structure for numerical normalization. To accelerate the convergence of the network, the data are input into the average pooling module for pooling. The formula is as follows:

$$O_1 = AvgPool(BatchNorm(O)) \tag{5}$$

3.3. Long-Order Information Mining

To effectively extract information across long stock sequences, this study incorporates a convolutional block attention module (CBAM) [35], which integrates both channel attention and spatial attention mechanisms. In this context, O denotes the initial feature map produced after convolution operations, which captures essential temporal and spatial characteristics. This feature map is then used as input for the channel and spatial attention mechanisms. Likewise, M_s represents the spatial attention feature map within CBAM, emphasizing spatially important regions in O to improve the model's focus on relevant areas.

A residual structure is added to this module to preserve effective data. The channel attention module, the first component of CBAM, is illustrated in Figure 3. Initially, data undergo average pooling and max pooling to generate two spatial descriptors for each channel. These descriptors are then fed into a shared, fully connected network where spatial information is updated and combined. The resulting channel-focused feature data are obtained through an activation function. This process can be formulated as follows:

$$G_1 = \sigma(MLP(AvgPool(O_1)) + MLP(MaxPool(O_1))) \tag{6}$$

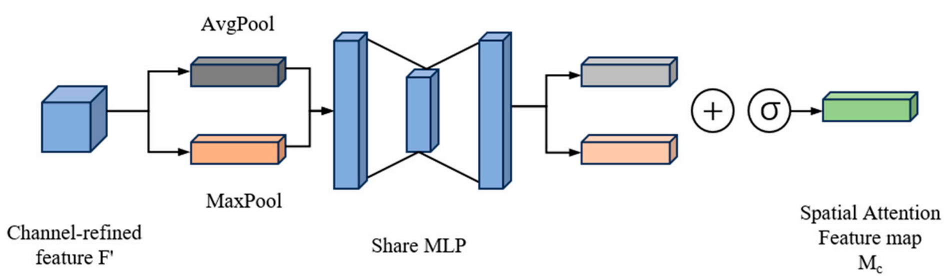

Figure 3. Channel attention mechanism diagram.

The second component of CBAM, which is designed to examine internal relationships within the spatial dimensions of feature information, is the spatial attention module, which enhances the interactive perception ability of the model toward input features. Its architecture, depicted in Figure 4, begins with average pooling and max pooling operations, followed by concatenation of the pooled results. A standard 7 × 7 convolutional layer and sigmoid activation function are subsequently applied to generate the spatial attention feature map. This process can be outlined as follows:

$$G_2 = \sigma(f_{7\times7}(concat(AvgPool(G_1), MaxPool(G_1)))) \tag{7}$$

To enhance network stability and mitigate overfitting issues, the spatial attention module incorporates a residual mechanism. This involves adding the input feature mapping to the output feature from the spatial attention mechanism, followed by prediction through a linear layer. The process is represented as follows:

$$\hat{y} = Linear(G_1 + G_2) \tag{8}$$

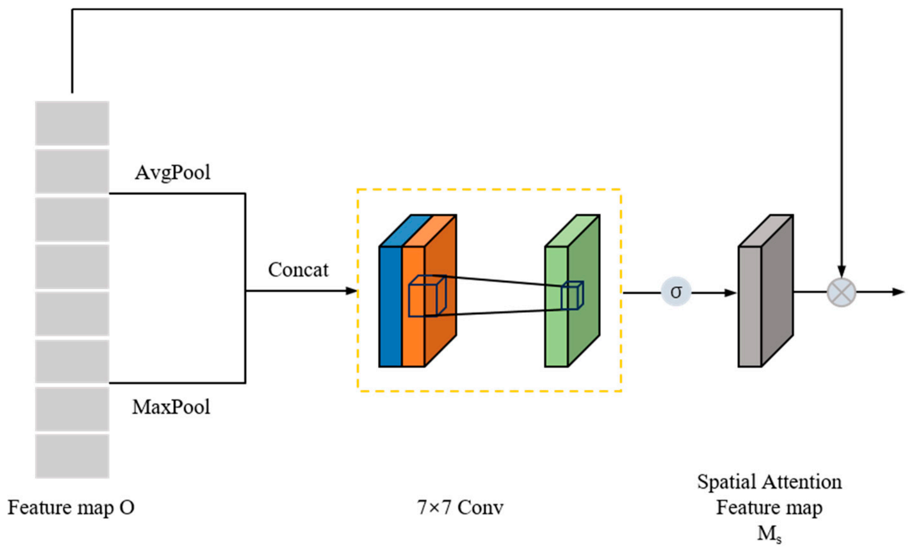

Figure 4. Feature fusion module.

4. Experiment

To better show the model's efficacy, this study compares it with the LSTM, CNN-LSTM, CNN-BiLSTM, random walk, random forest, and SVM models trained on identical datasets. The inputs consist of the opening price, highest price, lowest price, closing price, and trading volume, aimed at predicting the cumulative rate of return. Solely relying on historical single-stock indicators for future predictions may limit the model's ability to capture comprehensive market trends. The incorporation of multidimensional data inputs enhances the model's information retrieval capabilities.

Moreover, macroeconomic indicators exhibit a lag and weak correlation with short-term asset price fluctuations. Moreover, macro data can be subject to artificial manipulation. For this reason, for short-term price predictions, greater emphasis should be placed on technical indicators and market trading behaviors.

4.1. Dataset

This study focuses on the Australian stock market, specifically the AS51 index, also known as the ASX200. This index reflects the diverse characteristics of the Australian economy and its connections to the global market. The ASX200, based on market capitalization, includes the top 200 companies listed on the Australian Stock Exchange, covering sectors such as pharmaceuticals, banking, mining, and emerging industries like telecommunications technology. The total assets represented by the ASX200 exceed 130 billion US dollars, making it a stable and representative indicator of the Australian stock market.

The dataset utilized in this study spans from 4 January 2005, to 26 April 2024, with a total of 4874 trading days. It includes historical data on the opening price, highest price, lowest price, closing price, and trading volume. In the process of data collection, strict quality inspection was carried out to ensure the integrity and reliability of the dataset. No missing values were found in the selected indicators, and the data showed a stable trend without obvious fluctuations. Therefore, no additional preprocessing techniques, such as interpolation or smoothing, are required.

For the experiment, a random selection of a 600-day period from this dataset was made to analyze the most predictive price change indicators. The daily changes in opening price, highest price, lowest price, and closing price were the primary variables used for analysis in the deep-learning model training and testing.

4.2. Experimental Setup

The objective of this experiment is to predict daily price changes for the Australian S&P/ASX200 Index. A random subset of 600 trading days was selected from the dataset for analysis. The dataset includes historical data on the opening price, highest price, lowest price, closing price, and trading volume. These variables were used as input features to predict the cumulative return. A deep-learning model was employed for end-to-end training and testing on these data.

The results show that the daily changes in opening price significantly outperform other price change indicators in terms of prediction accuracy. To illustrate the model's performance, fit plots were generated to compare the actual and predicted values (Figure 5). These plots clearly highlight the predictive capability of the opening price changes over different periods. Furthermore, a detailed analysis of a subset of 100 days from the 600-day sample (Figure 6) further confirmed the stability and reliability of the predictions.

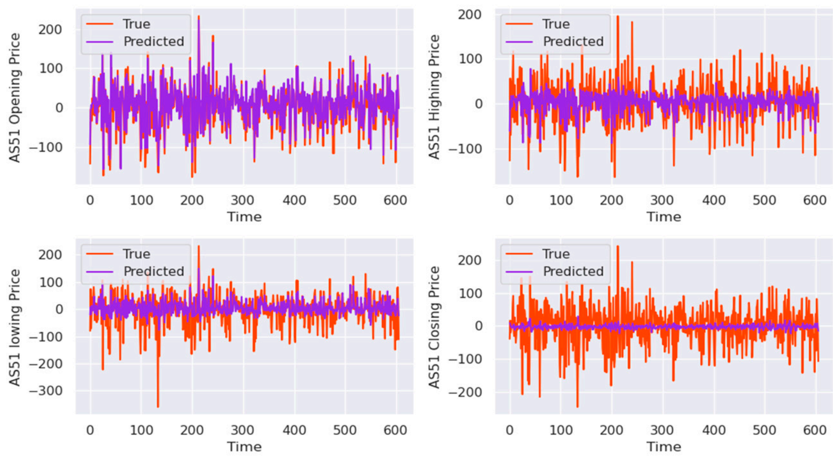

Figure 5. Actual vs. predicted price changes.

We use the following strategy to simulate actual stock trading: when the model predicts that the opening price will rise on the second day, we buy on the current day and sell on the second day (regardless of whether it actually rises or falls); otherwise, we do not take any action.

The findings of this study confirm the effectiveness of deep-learning models in predicting daily changes in opening prices and provide empirical support for the development of future trading strategies.

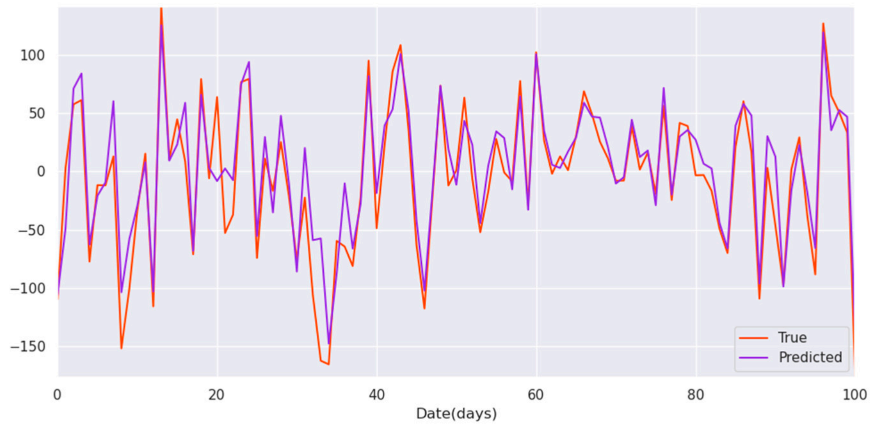

Figure 6. Opening price change predictions (100-day subset).

4.3. Performance Metrics

Model evaluation plays a pivotal role in model development and analysis, particularly in regression tasks. The key metrics typically employed include the cumulative rate of return (RETURN) [36], root-mean-squared error (RMSE) [37], mean absolute error (MAE) [38], and R^2 score (R^2), which facilitate an accurate assessment of the model performance for optimization and refinement. The cumulative rate of return is calculated under the premise of the aforementioned trading strategy, serving as a measure of the model's profitability. Moreover, other indicators are compared with the model's predictive ability. The calculation formulae are outlined below:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}, \in [0, +\infty) \tag{9}$$

$$RETURN = \prod_{t=1}^{n}(1 + R_t \cdot I_t) - 1 \tag{10}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|, \in [0, +\infty) \tag{11}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y_i})^2} \tag{12}$$

where n represents the number of samples, y_i represents the true value, and $\hat{y}_i$ represents the predicted value. In the return metric, R_t represents the return rate on day t, and n represents the total number of trading days. I_t is an indicator function. If it is predicted that the price will rise on day $t+1$, $I_t = 1$; otherwise, $I_t = 0$.

4.4. Ablation Experiment

We conducted a series of ablation experiments to evaluate each component's contribution to the proposed method. The main purpose of this section is to analyze the impact of removing different model components on the prediction accuracy to verify the role of these components in model performance.

First, we gradually removed key components from the model and evaluated each ablation model. Specifically, we removed the embedding layer, SIM, and LIM separately. All the ablation models were trained and tested on the same dataset (ASX200) using five-fold cross-validation to ensure the robustness and reliability of the results.

In the ablation experiment, we observed the following results: after removing the embedding layer, the R^2 value of the model decreased to 91.41%. This finding indicates that, although the embedding layer has little impact on the overall accuracy, it is still crucial for improving the model's accuracy. After removing the SIM module, the R^2 value of the model significantly decreased to 57.03%, highlighting the critical role of the SIM module in the model's predictive performance. On the other hand, after removing the LIM module, the (R^2) value of the model was 72.44%, indicating that CBAM, as an essential component of this part, can effectively integrate information across different scales, fully utilize multi-scale features, and further improve prediction performance.

Finally, we demonstrated the performance of the complete model (i.e., the proposed method) that includes all the components, with an R^2 value of 97.23%, as shown in Table 1. This result indicates that the integrated model of the embedding layer, SIM, and LIM can significantly improve the fitting ability of the data, verifying the necessity and effectiveness of these components in the model.

Table 1. Comparison of the forecasting results corresponding to the different evaluation metrics.

Model	RETURN
Remove Spatial_attention	3.8659
Remove Channel_attention	3.8507
Remove CNN	3.7147
The proposed method	4.1678

Through these ablation experiments, we verified each component's independent effects in the proposed method and provided important references for further research and application.

4.5. S&P/ASX200 Forecast

This dataset contains 4874 data points from the past 19 years. The first 70% of the daily historical data are used as the training set, and the last 30% are used as the test set; that is, 3411 data points from 4 January 2005 to 12 July 2018 are used as the training set, and 1463 data points from 13 July 2018 to 26 April 2024 are used as the test set for later model training. The above performance indices are used to evaluate the prediction results of different models. In order to reduce excessive quasi-consolidation and ensure the robust evaluation of the prediction performance of the model, the future verification method is adopted. In this method, the model is trained only on the early part of the dataset (training set), and the model is tested on the non-overlapping section (test set). This setting ensures that the model is evaluated based on the invisible future data, closely simulates the real-world prediction scenarios, and enhances its generalization ability to new market conditions.

The look-back period (days) refers to the historical data range used for prediction, typically focusing on recent days. In this experiment, the data encompass the five daily trading technical indicators mentioned earlier. To assess the performance of the four models across various review periods, this study chose five different review periods: 5 days, 10 days, 20 days, 40 days, and 60 days. Each model was evaluated using seven performance indicators across these five review periods, with the best results highlighted in bold in Table 2.

Table 2 shows that the prediction accuracy does not increase with an increasing number of review days. The experiment revealed that, when the number of review days was 20, the effect of the model was the greatest. The follow-up experiments with different datasets still use a 20-day review cycle.

In the 20-day review period experiment, the performance of the CNN-CBAM-LSTM model is significantly superior to that of the other models. Its RMSE is 21.03, which is 0.63 lower than that of the LSTM model (21.66), 5.32 lower than the CNN-Bi-LSTM model (26.35), and 2.05 lower than the CNN-LSTM model (23.08). The MAE of CNN-CBAM-LSTM is 14.81, 0.30 lower than LSTM (15.11), 5.29 lower than CNN-Bi-LSTM (20.10), and 1.44 lower than CNN-LSTM (16.25). For R^2, CNN-CBAM-LSTM is 85.15%, 0.90% higher than LSTM (84.25%), 8.46% higher than CNN-Bi-LSTM (76.69%), and 3.05% higher than CNN-LSTM (82.10%). Finally, the return rates of CNN-CBAM-LSTM were 416.78% and 18.72% higher than those of LSTM (398.06%), 59.86% higher than those of CNN-Bi-LSTM (356.92%), and 24.67% higher than those of CNN-LSTM (392.11%). These results highlight the advantages of CNN-CBAM-LSTM in terms of various indicators. All the experiments demonstrate that our model is highly competitive in terms of prediction accuracy and profitability.

Table 2. Comparison of the forecasting results corresponding to the different evaluation metrics.

Look-Back (Days)	Model	RMSE	MAE	R^2 (%)	RETURN (%)
5	CNN-CBAM-LSTM	**22.4853**	**16.5815**	**82.7398**	**416.6013**
5	LSTM	21.9569	15.6231	83.5414	408.6176
5	CNN-LSTM	29.7536	22.2981	69.7775	347.7133
5	CNN-Bi-LSTM	39.7168	30.6344	46.1483	240.8474
5	Random Walk	303.9385	52.3066	−12.7287	3.0846
5	Random Forest	22.8900	15.4864	82.1128	393.8533
5	SVM	31.1021	19.1712	66.9759	398.0802
10	CNN-CBAM-LSTM	**21.0038**	**14.5103**	**85.0408**	**415.1590**
10	LSTM	22.9625	16.8651	82.1206	413.5471
10	CNN-LSTM	23.5179	16.7758	81.2453	402.4180
10	CNN-Bi-LSTM	22.9564	16.8524	82.1301	392.9585
10	Random Walk	304.6167	52.5227	−12.8847	3.2745
10	Random Forest	22.9621	15.5427	82.1213	390.8926
10	SVM	31.2150	19.2601	66.9600	395.0941
20	CNN-CBAM-LSTM	**21.0257**	**14.8079**	**85.1536**	**416.7809**
20	LSTM	21.6573	15.1139	84.2483	398.0633
20	CNN-LSTM	23.0843	16.2489	82.1041	392.1141
20	CNN-Bi-LSTM	26.3461	20.0950	76.6894	356.9154
20	Random Walk	299.8871	52.4538	−8.4948	5.8192
20	Random Forest	23.1425	15.7282	82.0138	384.1477
20	SVM	31.4624	19.4984	66.7568	388.2915
40	CNN-CBAM-LSTM	**21.2100**	**14.7332**	**85.0272**	**380.0213**
40	LSTM	24.5280	18.8498	79.9763	353.9234
40	CNN-LSTM	21.8224	15.2465	84.1501	382.4799
40	CNN-Bi-LSTM	21.6754	15.0017	84.3631	385.6259
40	Random Walk	311.6648	53.6068	−14.0334	3.4455
40	Random Forest	23.2080	15.6797	82.0735	366.3835
40	SVM	31.7496	19.6352	66.4498	368.5855
60	CNN-CBAM-LSTM	**21.1959**	**14.5356**	**84.3838**	**363.6592**
60	LSTM	23.3538	17.8295	81.0424	344.0035
60	CNN-LSTM	23.5733	16.9221	80.6843	353.4915
60	CNN-Bi-LSTM	25.1808	18.3487	77.9601	334.0270
60	Random Walk	293.4858	52.1406	−0.0038	11.9090
60	Random Forest	23.5102	15.8988	80.7876	343.4913
60	SVM	30.7820	19.1231	67.0646	345.5852

Figure 7 presents the index prediction results of the six models applied to the same dataset (AS51) over a 50-day period. Evidently, the predictions from CNN-CBAM-LSTM are closer to the actual values.

This experiment compares the performance of the CNN-CBAM-LSTM model and three other models in terms of the AS51 index from 13 July 2018 to 26 April 2024. The results show that the random walk model performs the worst in this time period, whereas the CNN-CBAM-LSTM model performs significantly better than the other models.

The partially enlarged subgraph in Figure 8 shows that the CNN-CBAM-LSTM prediction is very close to the true value curve. Even when the opening price fluctuates too much (absolute value greater than 50), it is difficult for the model to fit to the peak, but the general direction of the rise and fall is correct, which proves that the model successfully captures the changes in the opening price.

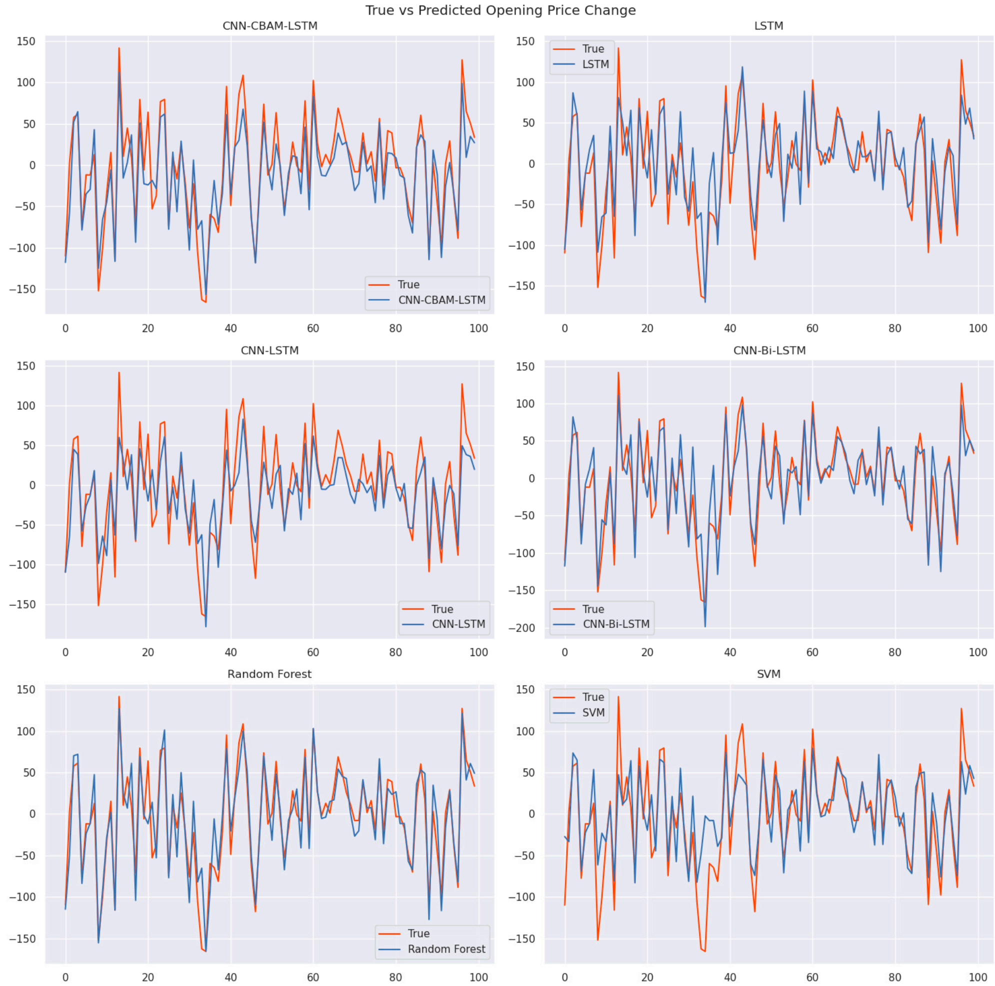

Figure 7. Forecasting results of different models on the S&P/ASX200 index for the 100-day test period (look-back = 20 days).

Figure 8. Prediction results of different models for the AS51 index of the whole test set (look-back = 20 days).

4.6. Diebold–Mariano Test for Predictive Performance Evaluation

After quantitatively comparing the predictive performance of different models, we conducted the Diebold–Mariano (DM) test to further validate the significant differences in predictive ability among the models [39]. The DM test is a statistical method widely used for comparing the accuracy of two forecasting models, particularly in time-series prediction.

In this study, we employed the DM test to compare the CNN-CBAM-LSTM model with other models, including LSTM, CNN-LSTM, CNN-Bi-LSTM, random forest, and support vector machine (SVM), to assess their predictive performance in terms of actual returns. The hypotheses for the DM test are as follows:

Hypotheses:

Null Hypothesis (H0): *There is no significant difference in predictive performance between the two models.*

Alternative Hypothesis (H1): *There is a significant difference in predictive performance between the two models.*

We calculated the DM statistics and their corresponding p values for each model comparison, which are presented in Table 3.

Table 3. DM statistics and p values for different predictive models.

Model	DM Statistic	p Value
LSTM	-12.283	4.06×10^{-31}
CNN-LSTM	-6.390	3.32×10^{-10}
CNN-Bi-LSTM	-6.275	6.69×10^{-10}
Random Forest	-5.460	6.95×10^{-8}
SVM	-4.024	6.44×10^{-5}

The DM test results highlight the significant advantage of the CNN-CBAM-LSTM model over the other models. For example, the DM statistic for the LSTM model is -12.283, with an extremely low p value (4.06×10^{-31}), indicating that the LSTM model's predictive performance is significantly weaker than that of the CNN-CBAM-LSTM. This pattern is consistent across the other models—CNN-LSTM, CNN-Bi-LSTM, random forest, and SVM—whose DM statistics and p values similarly demonstrate inferior predictive abilities compared with those of the CNN-CBAM-LSTM model.

These findings reinforce the conclusions from earlier model comparisons and underscore the importance of choosing the right model for practical applications. The DM test quantitatively assesses differences in predictive performance, offering a solid foundation for more informed decision-making.

4.7. Model Confidence Set (MCS) Robustness Test

To further evaluate the robustness of the models, we conducted a model confidence set (MCS) test. The MCS is a statistical procedure that identifies a set of models that are statistically indistinguishable in terms of predictive performance at a specified confidence level [40]. In other words, the MCS test helps determine which models are likely to be among the "best" performers on the basis of their predictive capabilities.

In this experiment, we compared the performance of seven models: CNN-CBAM-LSTM, LSTM, CNN-LSTM, CNN-Bi-LSTM, SVM, random forest, and random walk. The MCS test was conducted at the 95% confidence level, and the resulting p values are presented in Table 4.

The MCS results indicate that the CNN-CBAM-LSTM model is the only model with a p value of 1.000, making it the most likely candidate for inclusion in the confidence set of best-performing models. This finding suggests that the CNN-CBAM-LSTM model is more robust than the other models. In contrast, the p values for the remaining models—LSTM (0.042), CNN-LSTM (0.091), CNN-Bi-LSTM (0.091), SVM (0.091), random forest (0.091), and random walk (0.091)—indicate that their performances are statistically indistinguishable from one another but inferior to that of the CNN-CBAM-LSTM model.

Table 4. *p* values of different models based on MCS testing.

Model	*p* Value
LSTM	0.042
CNN-Bi-LSTM	0.091
SVM	0.091
CNN-LSTM	0.091
Random Forest	0.091
Random Walk	0.091
CNN-CBAM-LSTM	1.000

These findings affirm the robustness of the CNN-CBAM-LSTM model and reinforce its superior performance observed in earlier tests. By identifying CNN-CBAM-LSTM as the only model included in the MCS with a high degree of confidence, this test underscores its strong predictive capability across varying conditions.

4.8. Experimental Comparison of Different Stock Indices

Six well-known indices, namely, the HSI, N255, SPX, FTSE, IXIC, and TWII, were obtained for the experiments to verify the robustness and generalizability of the model.

The Hang Seng Index (HSI) is a crucial indicator for tracking stock market movements in Hong Kong. The Nikkei 225 Index (N225) monitors the performance of 225 stocks listed on the Tokyo Stock Exchange in Japan, encompassing metrics such as opening price, highest price, lowest price, closing price, and trading volume. The Standard & Poor's 500 Index (SPX) is a widely recognized benchmark in the U.S. stock market, comprising 500 large-cap companies weighted by their market capitalizations. The FTSE 100 Index is a benchmark for the largest companies in the UK listed on the London Stock Exchange, comprising 100 blue-chip stocks. The Nasdaq Composite Index (IXIC) tracks all stocks in the U.S. on the Nasdaq Stock Exchange, primarily focusing on technology and growth companies. The Taiwan weighted index (TWII) is an important index compiled by the Taiwan Stock Exchange that reflects the overall performance of the Taiwan stock market.

For consistency in the experimental results, the six aforementioned stock indices were chosen to coincide with the same time frame as the Australian Standard & Poor's 200 (AS51). Similarly, the initial 70% of the data was allocated for model training, whereas the remaining 30% was reserved for testing purposes.

Figures 9–14 illustrates how well the model predicts outcomes across six diverse stock indices in this study. The red line signifies the actual observed values, whereas the purple line denotes the values predicted from the CNN-CBAM-LSTM model. The experimental results are shown in Table 5.

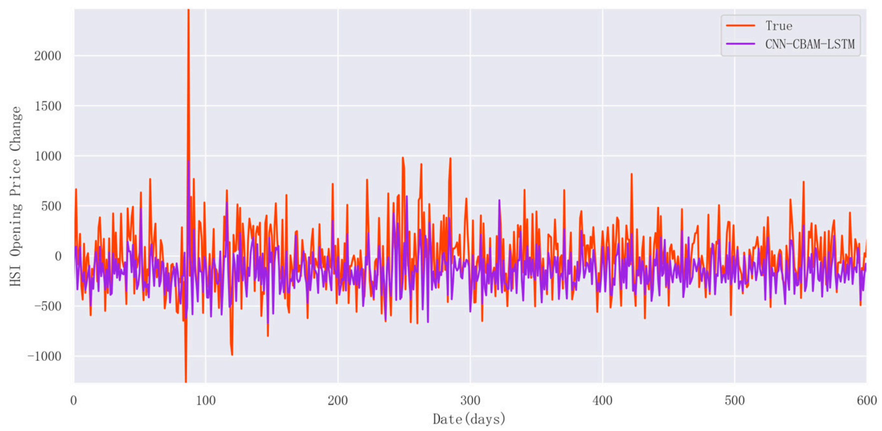

Figure 9. CNN-CBAM-LSTM model's forecasting capability for HSI indices.

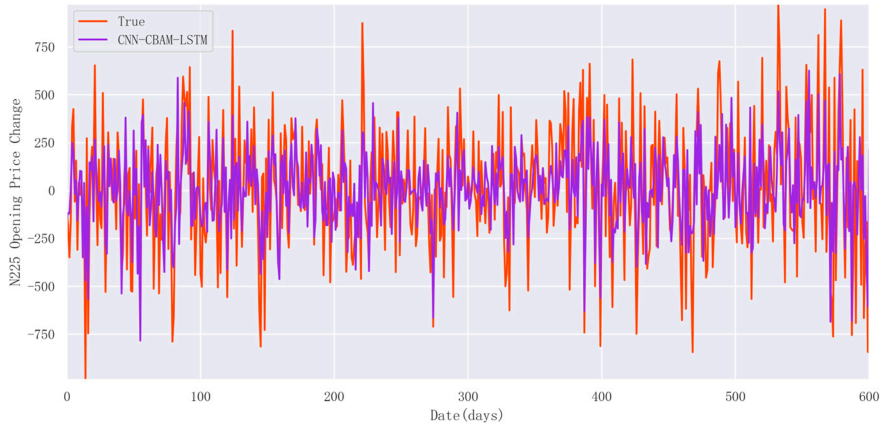

Figure 10. CNN-CBAM-LSTM model's forecasting capability for N225 indices.

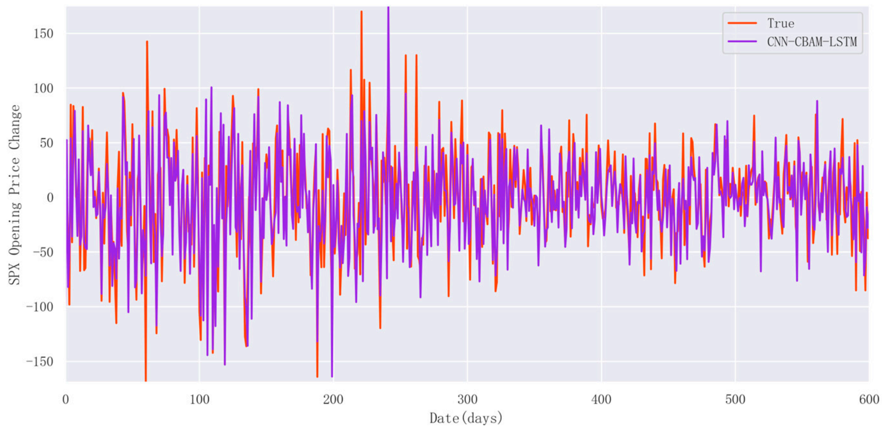

Figure 11. CNN-CBAM-LSTM model's forecasting capability for SPX indices.

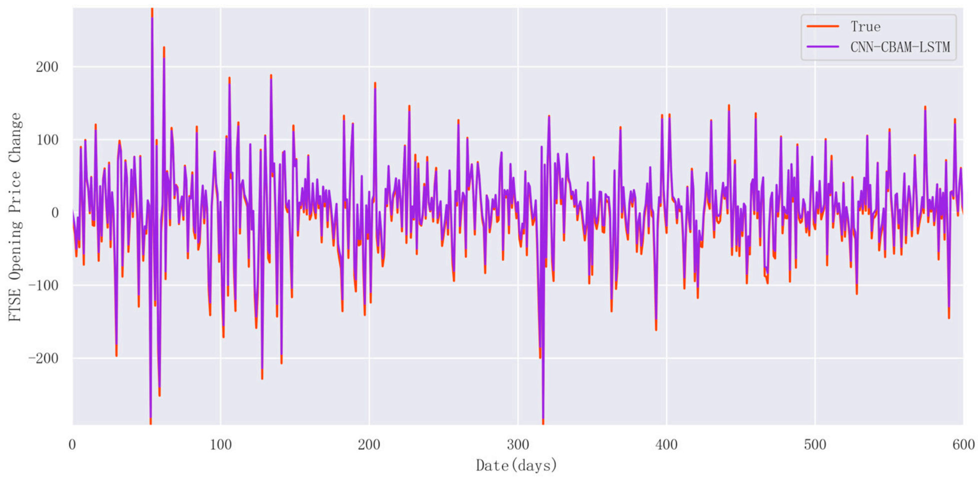

Figure 12. CNN-CBAM-LSTM model's forecasting capability for FTSE indices.

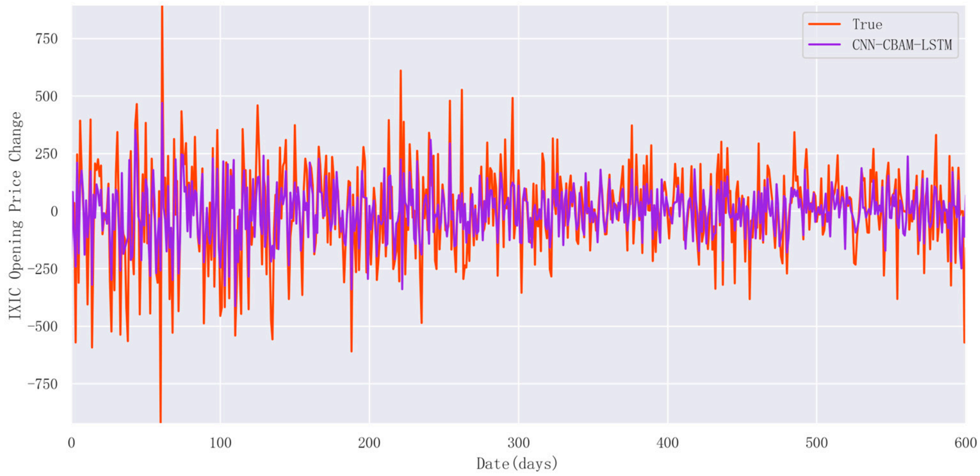

Figure 13. CNN-CBAM-LSTM model's forecasting capability for IXIC indices.

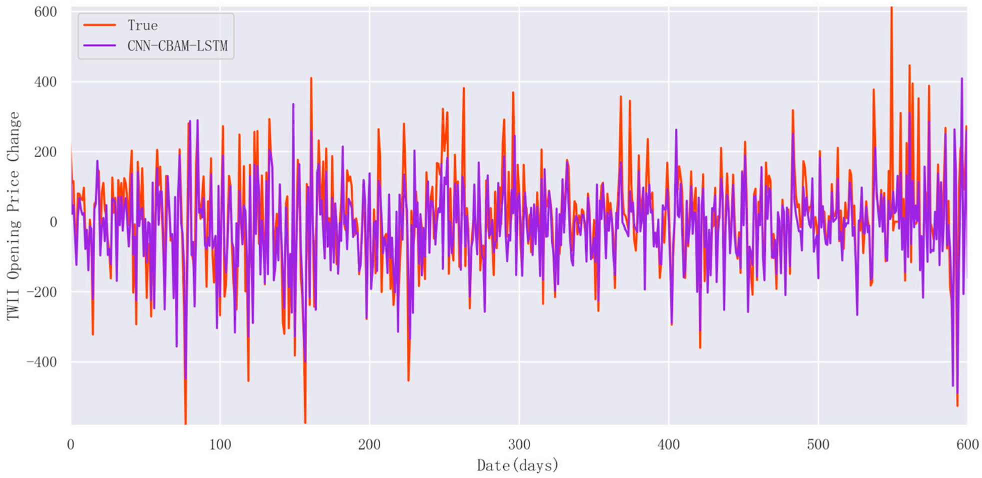

Figure 14. CNN-CBAM-LSTM model's forecasting capability for TWII indices.

Table 5. Comparison of the experimental results corresponding to different stock indices and evaluation metrics.

Index	Look-Back (Days)	Model	RMSE	MAE	R^2 (%)	RETURN (%)
HSI	20	CNN-CBAM-LSTM	191.91	142.23	59.32	1388.87
N225	20	CNN-CBAM-LSTM	213.73	192.55	44.72	498.15
SPX	20	CNN-CBAM-LSTM	23.91	18.11	73.86	766.15
FTSE	20	CNN-CBAM-LSTM	515.27	417.04	99.33	555.09
IXIC	20	CNN-CBAM-LSTM	116.86	87.25	61.90	1705.14
TWII	20	CNN-CBAM-LSTM	76.95	54.27	73.21	527.69

The experimental results show that the predicted price change curves for the six stock indices closely match the actual values, highlighting not only the universal applicability of the CNN-CBAM-LSTM model but also, more importantly, its significant advantage in predicting stock returns.

While the results demonstrate the versatility and reliability of CNN-CBAM-LSTM in forecasting these indices, certain limitations should be acknowledged. In particular, the relatively small dataset used in the experiment may limit the model's generalizability and robustness. To provide a more comprehensive evaluation of the model's performance, future research could expand the scope by incorporating a broader and more diverse range of stock data for validation. Additionally, exploring other factors that may affect the model's effectiveness—such as changing market conditions, data frequency, and feature selection—could further enhance its ability to predict returns and its practical utility. More extensive studies are needed to thoroughly assess and optimize the stock return prediction performance of the CNN-CBAM-LSTM model.

5. Conclusions

To predict stock price changes, this study introduces the SANet deep-learning network model with a feature fusion module. The method utilizes five key features from time-series data: opening price, highest price, lowest price, closing price, and trading volume. Four performance indicators (RMSE, MAE, R2, and RETURN) are employed to assess the model's effectiveness. The experiment focuses on Australia's S&P 200 (AS51) stock index to validate the CNN-CBAM-LSTM model's predictive performance. The results indicate that, compared with the other models, the proposed CNN-CBAM-LSTM model has the lowest RMSE and MAE, an R^2 (%) value closer to 100%, and a significantly higher RETURN (%). This highlights the model's superior accuracy in predicting stock price changes while also demonstrating its advantage in generating higher returns, making it highly effective for stock market forecasting applications. Furthermore, to test the model's robustness, the study applies it to six well-known global stock indices. The findings consistently demonstrate the superior performance of the proposed CNN-CBAM-LSTM model. This research underscores the effectiveness of CNN-CBAM-LSTM and its potential applicability across diverse global stock market indices.

Although the results are promising, several limitations should be acknowledged. First, this study focuses on a limited set of stock-specific indicators, which may not fully capture the broader market dynamics. Additionally, the model's reliance on historical price data may result in a lower accuracy during high-volatility scenarios with sudden market shifts. These factors may limit the model's adaptability to predict unpredictable events.

Future work will focus on enhancing the model by incorporating additional factors such as macroeconomic indicators and industry-specific data, which could improve its adaptability to external environmental changes. Investigating other feature-engineering techniques may also strengthen the model's robustness under conditions of high fluctuation. Furthermore, the application of ensemble-learning strategies could enhance the model's versatility across different markets. A quantitative analysis approach will be used to assess the impact of these extended features on prediction accuracy, with the goal of developing a more general and reliable financial prediction model.

Author Contributions: Writing—original draft preparation, P.Y. and H.Z.; writing—review and editing, X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Statistical Science Research Project (2023LY098), the National College Students' Innovation and Entrepreneurship Training Program (202313283009), and the Talent Launch Project of Scientific Research Development Fund of Jiyang College of Zhejiang A&F University (RC2021D10).

Data Availability Statement: The data will be made available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Dash, R.K.; Nguyen, T.N.; Cengiz, K.; Sharma, A. Fine-tuned support vector regression model for stock predictions. *Neural Comput. Appl.* **2023**, *35*, 23295–23309. [CrossRef]
2. Kamalov, F. Forecasting significant stock price changes using neural networks. *Neural Comput. Appl.* **2020**, *32*, 17655–17667. [CrossRef]
3. Yiming, W.; Xun, L.; Umair, M.; Aizhan, A. COVID-19 and the transformation of emerging economies: Financialization, green bonds, and stock market volatility. *Resour. Policy* **2024**, *92*, 104963. [CrossRef]

4. Sheth, D.; Shah, M. Predicting stock market using machine learning: Best and accurate way to know future stock prices. *Int. J. Syst. Assur. Rng. Manag.* **2023**, *14*, 1–18. [CrossRef]
5. Strielkowski, W.; Vlasov, A.; Selivanov, K.; Muraviev, K.; Shakhnov, V. Prospects and challenges of the machine learning and data-driven methods for the predictive analysis of power systems: A review. *Energies* **2023**, *16*, 4025. [CrossRef]
6. Li, C.; Chen, J. An End-to-End structure with novel position mechanism and improved EMD for stock forecasting. In Proceedings of the International Conference on Neural Information Processing, Changsha, China, 20–23 November 2023.
7. Ruan, Y.; Zheng, M.; Qian, F.; Meng, H.; Yao, J.; Xu, T.; Pei, D. Fault detection and diagnosis of energy system based on deep learning image recognition model under the condition of imbalanced samples. *Appl. Therm. Eng.* **2024**, *238*, 122051. [CrossRef]
8. Lauriola, I.; Lavelli, A.; Aiolli, F. An introduction to deep learning in natural language processing: Models, techniques, and tools. *Neurocomputing* **2022**, *470*, 443–456. [CrossRef]
9. Jang, J.; Seong, N. Deep reinforcement learning for stock portfolio optimization by connecting with modern portfolio theory. *Expert Syst. Appl.* **2023**, *218*, 119556. [CrossRef]
10. Liang, Y.; Lin, Y.; Lu, Q. Forecasting gold price using a novel hybrid model with ICEEMDAN and LSTM-CNN-CBAM. *Expert Syst. Appl.* **2022**, *206*, 117847. [CrossRef]
11. Li, S.; Tang, G.; Chen, X.; Lin, T. Stock index forecasting using a novel integrated model based on CEEMDAN and TCN-GRU-CBAM. *IEEE Access* **2024**, *12*, 122524–122543. [CrossRef]
12. Belhoula, M.M.; Mensi, W.; Naoui, K. Impacts of investor's sentiment, uncertainty indexes, and macroeconomic factors on the dynamic efficiency of G7 stock markets. *Qual. Quant.* **2024**, *58*, 2855–2886. [CrossRef]
13. Zheng, H.; Wu, J.; Song, R.; Guo, L.; Xu, Z. Predicting financial enterprise stocks and economic data trends using machine learning time series analysis. *Appl. Comput. Eng.* **2024**, *87*, 26–32. [CrossRef]
14. Mintarya, L.N.; Halim, J.N.M.; Angie, C.; Achmad, S.; Kurniawan, A. Machine learning approaches in stock market prediction: A systematic literature review. *Procedia Comput. Sci.* **2023**, *216*, 96–102. [CrossRef]
15. Kim, M.; Sayama, H. Predicting stock market movements using network science: An information theoretic approach. *Appl. Netw. Sci.* **2017**, *2*, 35. [CrossRef] [PubMed]
16. Lin, L. Stock prediction and analysis based on RNN Neural Network. In Proceedings of the 3rd International Symposium on Economics, Management, and Sustainable Development (EMSD), Hangzhou, China, 29–30 November 2022.
17. Xiao, D.; Su, J. Research on stock price time series prediction based on deep learning and autoregressive integrated moving average. *Sci. Program.* **2022**, *2022*, 4758698. [CrossRef]
18. Patra, G.R.; Mohanty, M.N. An LSTM-GRU based hybrid framework for secured stock price prediction. *J. Stat. Manag. Syst.* **2022**, *25*, 1491–1499. [CrossRef]
19. Zaheer, S.; Anjum, N.; Hussain, S.; Algarni, A.D.; Iqbal, J.; Bourouis, S.; Ullah, S.S. A multi parameter forecasting for stock time series data using LSTM and deep learning model. *Mathematics* **2023**, *11*, 590. [CrossRef]
20. Eapen, J.; Bein, D.; Verma, A. Novel deep learning model with CNN and bi-directional LSTM for improved stock market index prediction. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019.
21. Yadav, K.; Yadav, M.; Saini, S. Stock values predictions using deep learning based hybrid models. *CAAI Trans. Intell. Technol.* **2022**, *7*, 107–116. [CrossRef]
22. Zhang, J.; Ye, L.; Lai, Y. Stock price prediction using CNN-BiLSTM-Attention model. *Mathematics* **2023**, *11*, 1985. [CrossRef]
23. Li, M.; Zhu, Y.; Shen, Y.; Angelova, M. Clustering-enhanced stock price prediction using deep learning. *World Wide Web* **2023**, *26*, 207–232. [CrossRef]
24. Jiang, J.; Wu, L.; Zhao, H.; Zhu, H.; Zhang, W. Forecasting movements of stock time series based on hidden state guided deep learning approach. *Inform. Process. Manag.* **2023**, *60*, 103328. [CrossRef]
25. Wang, C.; Chen, Y.; Zhang, S.; Zhang, Q. Stock market index prediction using deep Transformer model. *Expert Syst. Appl.* **2022**, *208*, 118128. [CrossRef]
26. Yang, F.; Chen, J.; Liu, Y. Improved and optimized recurrent neural network based on PSO and its application in stock price prediction. *Soft Comput.* **2023**, *27*, 3461–3476. [CrossRef]
27. Wang, J.; Zhuang, Z.; Feng, L. Intelligent optimization based multi-factor deep learning stock selection model and quantitative trading strategy. *Mathematics* **2022**, *10*, 566. [CrossRef]
28. Zhao, J.; Zeng, D.; Liang, S.; Kang, H.; Liu, Q. Prediction model for stock price trend based on recurrent neural network. *J. Amb. Intel. Hum. Comp.* **2021**, *12*, 745–753. [CrossRef]
29. Ji, X.; Wang, J.; Yan, Z. A stock price prediction method based on deep learning technology. *Int. J. Crowd Sci.* **2021**, *5*, 55–72. [CrossRef]
30. Rezaei, H.; Faaljou, H.; Mansourfar, G. Stock price prediction using deep learning and frequency decomposition. *Expert Syst. Appl.* **2021**, *169*, 114332. [CrossRef]
31. Lin, Y.; Yan, Y.; Xu, J.; Liao, Y.; Ma, F. Forecasting stock index price using the CEEMDAN-LSTM model. *N. Am. J. Econ. Financ.* **2021**, *57*, 101421. [CrossRef]
32. Zheng, Y.; Xu, Z.; Xiao, A. Deep learning in economics: A systematic and critical review. *Artif. Intell. Rev.* **2023**, *56*, 9497–9539. [CrossRef]

33. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 1–74. [CrossRef]
34. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [CrossRef] [PubMed]
35. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
36. Ang, A.; Bekaert, G. Stock return predictability: Is it there? *Rev. Financ. Stud.* **2007**, *20*, 651–707. [CrossRef]
37. Kumar, A.; Alsadoon, A.; Prasad, P.W.C.; Abdullah, S.; Rashid, T.A.; Pham, D.T.H.; Nguyen, T.Q.V. Generative adversarial network (GAN) and enhanced root mean square error (ERMSE): Deep learning for stock price movement prediction. *Multimed. Tools Appl.* **2022**, *81*, 3995–4013. [CrossRef]
38. Chen, Y.; Fang, R.; Liang, T.; Sha, Z.; Li, S.; Yi, Y.; Zhou, W.; Song, H. Stock price forecast based on CNN-BiLSTM-ECA model. *Sci. Program.* **2021**, *2021*, 2446543. [CrossRef]
39. Diebold, F.X. Comparing predictive accuracy, twenty years later: A personal perspective on the use and abuse of Diebold–Mariano tests. *J. Bus. Econ. Stat.* **2015**, *33*, 1. [CrossRef]
40. Hansen, P.R.; Lunde, A.; Nason, J.M. The model confidence set. *Econometrica* **2011**, *79*, 453–497. [CrossRef]