The S&P 500 Index Prediction Based on N-BEATS

Yichen Liu[1], Chengcheng Zhong[2(✉)], Qiaoyu Ma[3], Yanan Jiang[4], and Chunlei Zhang[5]

[1] Cyber Science and Engineering, Qufu Normal University, Jining 273165, China
[2] School of Science, China University of Geosciences, Beijing 100083, China
2119200033@cugb.edu.cn
[3] College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China
[4] School of Mathematical Sciences, Beijing Normal University, Beijing 100875, China
[5] Beijing Zhongdirunde Petroleum Technology Co. Ltd., Beijing 100083, China

Abstract. The stock market prediction has been a hot topic in the field of economics and finance. As a consequence of the complex and volatile nature of the stock market, it is challenging to accurately forecast the stock S&P 500 index. Currently, with the purpose of predicting stock market, intelligent algorithms via computer have been proved superior in recent studies. We have introduced the N-BEATS algorithm to precisely estimate the stock S&P 500 index which are tailored towards the drawbacks that most algorithms cannot incorporate with historical information for time-series data. The features extracted by the N-BEATS algorithm are more consistent with the temporal features through the forward and backward coefficients. On the basis of the comparison of four evaluation metrics obtained from the S&P 500 index corresponding to 500 base stocks in this study, the N-BEATS algorithm outperforms other estimators. It can be demonstrated that the N-BEATS is a more suitable and promising method for stock market prediction, which has widespread application value.

Keywords: Stock market prediction · S&P 500 · N-BEATS · Time-series

1 Introduction

Financial markets are very complex and volatile. As the stock market is one of the most important components of a country's economy, it is crucial to accurately forecast it to give an early warning of a stock market crash and subsequent recovery. Currently, there are several approaches that try to make a reliable prediction on stock markets, however, it is difficult to directly predict the future trend of stock due to the Efficient Market Hypothesis, namely the economic, political, environmental, and other factors that can affect stock changes. The S&P 500 index is an average record of the stock market [1], whose change can indicate a positive or negative return in the stock market. Therefore, constructing an accurate prediction model of the S&P 500 index plays an important role in evaluating the stock market.

J. Yen et al. (Eds.): ICBIS 2023, AHCS 14, pp. 923–929, 2023.
https://doi.org/10.2991/978-94-6463-198-2_96

The explosion of computing power and the amount of data have inspired wide interest in utilizing machine learning models to predict the change in financial markets. Artificial neural networks (ANN), support vector machines (SVM), genetic algorithms, and other techniques have been used to forecast and analyze financial markets over the past decade. These methods are easier to obtain higher accuracy without any assumptions of the data compared with traditional mathematical statistics methods [2–4]. Recently, deep learning has become a popular technique for predicting the stock market due to its power capacity of non-linear relationship mining and adaptive learning [5, 6]. A variety of deep learning methods have been applied to many important fields, such as financial markets [4]. However, most deep learning methods have limitations in time series data prediction. On the one hand, ANN, convolutional neural networks (CNN), and other models cannot combine the characteristics of time-series data in feature fusion, resulting in poor performance. On the other hand, recurrent-series models such as recurrent neural networks (RNN) cannot solve the problem of historical information transfer in time-series data, which still have flaws in time-series prediction in the financial market [7].

To cope with the above puzzles, the N-BEATS algorithm is introduced to process time-series data collected from 500 constituent stocks to predict the S&P 500 index [8]. The N-BEATS utilizes the forward and backward coefficients obtained from a double-branch structure, of which the feature extraction is more suitable with the characteristics of time-series data. Comparing the N-BEATS algorithm with benchmarks obtained from 7 deep learning models shows that the model error is considerably less than those models in terms of 4 evaluation metrics, which demonstrates the ability of the N-BEATS in effectiveness for financial time-series prediction tasks.

2 Model

2.1 N-BEATS

N-BEATS is a new univariate time series data prediction architecture proposed by Oreshkin in 2019. The structure of the model is shown in Fig. 1. As can be seen, the model has a simple and general architecture, which also has high expressiveness in depth. In addition, the model structure does not depend on specific feature engineering for time series data or input scaling.

The core part of the model is composed of standard fully connected layers, that is θ^f, in which data are nonlinearly mapped by stacking fully connected layers. The output sequence is divided into two branches: forward predicted result and backward predicted result. The prediction process of the two branches is simulated with the forward coefficient θ^f and backward coefficient θ^b for the basic expansion, respectively. Specifically, the model optimizes the prediction of y'^b by mapping the forward expansion coefficient θ^f through the base vector of g^f on the one hand. The model on the other hand generates an estimate of the historical data x by taking the backward expansion coefficient θ^b utilized in g^b and removing them in each block. The residuals are then employed as input to the downstream module and the residual features are utilized to assist the prediction of downstream blocks.

The peripheral part maps the expansion coefficients θ^f and θ^b learned above to the overall output, which also adopts the two-branch structure $y^b = g^f\left(\theta^f\right)$ and $x^b = g^b\left(\theta^b\right)$.

This allows their respective outputs to be adequately represented by different expansion coefficients θ^f and θ^b. The forward coefficient θ^f and backward coefficient θ^b from different stages can be learned by the functions g^b and g^f. In this way, g^b and g^f in each step can effectively reflect the inductive bias of the forward or backward features in sequence data, which can be added or removed in the input of the next step to constrain the downstream structure appropriately.

The output of the model organizes the blocks into stacks using double residual stacking for the output results of each step and shares the results of g^b and g^f in the process. The predictions of their residual transfer are aggregated hierarchically, eventually forming a neural network with interpretable outputs. For other blocks in the model, the posterior residual branch can be thought of as performing a time series analysis of the input signal. The results obtained in the data flow through the upstream block remove the approximate part of the signal, allowing the information to be simplified and the prediction in the downstream block to be easy. In addition, each block in the model outputs a partial prediction, and the predictions of this hierarchical decomposition are first aggregated at the stack level and then at the entire network level, with the final prediction being the sum of all partial predictions. In a general model context, it makes the network more transparent to gradient flow when the stack is allowed to have arbitrary g^b and g^f structures for each layer. In the special case of intentional structures enforced in g^b and g^f via stack sharing, it is critical to achieving interpretability by aggregating meaningful partial predictions.

In this way, the N-BEATS structure can simultaneously produce values that affect the prediction result and the next basic block in each basic block. The two-branch structure not only effectively retains the historical information but also quickly learns in combination with the feature information learned before the two-branch structure, thereby achieving efficient and accurate prediction.

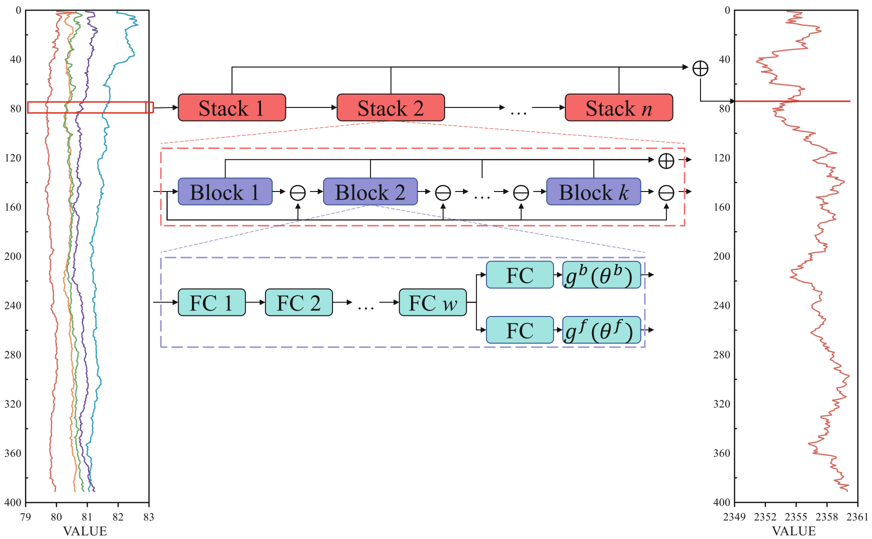

Fig. 1. The network structure of N-BEATS

2.2 Model Specifications

The network structure of N-BEATS is shown in Fig. 1. The whole structure of the model consists of n stacks, and each stack includes k blocks, which are organized into stacks by employing the doubly residual stacking principle. Each block is built from a multi-layer FC network with RELU nonlinearity, which has a two-branch structure. The model consists of a 15×501-dimensional input layer and a one-dimensional output layer. The input layer represents the 500 stocks and S&P 500 indices for the previous 15 min, while the output layer represents the predicted S&P 500 indices for the next minute. In this study, we set the stack structure of the model as 2, each layer has 4 blocks, and each block is composed of a two-branch structure with 3 fully connected layers. In addition, in order to keep the input and output of the module consistent with the sequence length and the number of nodes in each layer, we set the number of fully connected structures before the two-branch to 64.

3 Experiment and Result

3.1 Dataset

The experimental dataset in our experiments comes from a contest run by STATWORX, which fetched detailed data on 500 stocks and their S&P 500 indices from the Google Finance API from April to August 2017. The input data is 500 stocks and their corresponding S&P 500 indices for the previous 15 min. Upon reshaping the data size is (41,251, 15, 501), where 41,251 is the minutes, 15 is the length of each input sequence considering the historical information, and 501 is the feature dimension.

3.2 Settings

The experiments were implemented by Python 3.7, and PyTorch 1.3.0 on a PC platform with an Intel I CoITM) i5-9400 CPU @ 2.90 GHz, NVIDIA GeForce RTX 2060 environment. We compared the N-BEATS with 7 deep learning models, namely CNN, Temporal Convolutional Networks (TCN), RNN, Long Short-Term Memory (LSTM), Gate Recurrent Unit (GRU), Vision Transformer (ViT) [9], and MLP [10]. We adopted four evaluation metrics to evaluate the performance of the different models, namely correlation coefficient (COR), coefficient of determination (R^2), mean absolute percentage error (MAPE), and root mean square error (RMSE). COR and R^2 can help judge the degree of similarity between the variation trend of the predicted results and the actual values. MAPE and RMSE are employed to evaluate the numerical gap between the predicted results and the actual values.

3.3 Analysis

Table 1 shows the performance of different deep learning models. As can be noted, the MAPE of the N-BEATS is the smallest (0.02%) compared with others, indicating that the estimated values using the N-BEATS are closer to the actual values. On the other hand, its COR between the predicted and the actual values reaches the highest (0.9998), demonstrating that the estimated trend is highly consistent with the actual one.

Figure 2 visualizes the predicted results of a certain day in the dataset to further analyse the performance of models. It is obvious that the error between the predicted values and the actual values based on the N-BEATS is smaller than that of other algorithms, and the curve shows a higher coincidence degree without a large prediction deviation, indicating the evident strength of the N-BEATS for the prediction of time-series data that contains spatial structural features such as stock price.

The RMSE of CNN and TCN is relatively large, and the predicted values have significant fluctuations in a major segment of this case study, which indicates that the convolutional structure cannot effectively capture the features of time-series data when predicting the future state, resulting in poor performance. Although the predicted results of RNN do not diverge greatly from the overall trend, most of the results have a certain difference from the original values. This is because RNN tends to make the gradient too small when learning the long dependence relationship, which cannot save the information before a long-term step. The LSTM and GRU models can use the "gate" structure to retain the information of the long-term step, while their prediction of short-term mutation

Table 1. Prediction performance of deep learning models

Models	COR	R^2	MAPE (%)	RMSE
TCN	0.9996	0.9991	0.04	1.1749
CNN	0.9944	0.8810	0.46	13.6938
MIXER	0.9991	0.9979	0.06	1.7979
ViT	0.9995	0.9990	0.04	1.2545
RNN	0.9990	0.9978	0.06	1.8776
LSTM	0.9995	0.9990	0.04	1.2861
GRU	0.9996	0.9991	0.04	1.1769
N-BEATS	0.9998	0.9996	0.02	0.8242

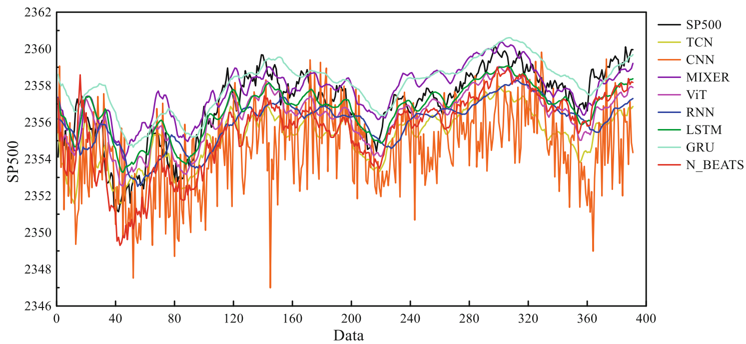

Fig. 2. The comparison of predicted results from various deep learning models

information is not accurate enough, so the results have a large deviation at the mutation, which does not affect the overall performance. The ViT and MLP models are suitable for feature extraction of sequential data, but they are unable to extract features of time-series data and lack the ability to extract and fusion of historical and future information, resulting in poor performance. The forward and backward coefficients used in the N-BEATS can simulate the historical and future information from the extracted temporal features and then remove the historical factors in the information before the two-branch structure and combine the rest part with the future one, which can quickly learn the extracted features and predict more precisely. Therefore, the N-BEATS achieves the optimal results in 4 evaluation indicators (COR:0.9998, R^2:0.9996, MAPE: 0.02%, and RMSE:0.8242). The above experimental results fully verify the N-BEATS effectiveness in financial time-series data prediction, which can construct a high-precision model for forecasting the stock market in line with the actual financial market condition and pay more attention to features extracted from the time-series data itself.

4 Conclusion

In this paper, we have introduced the N-BEATS architecture for time series prediction and successfully utilized it to predict the next-minute values of the S&P 500 index. The experimental results indicate that N-BEATS is a feasible and applicable deep learning method for financial market prediction. Compared to the current mainstream deep learning models, the N-BEATS network is more suitable for stock market prediction as it can mine the historical information and future trends separately when extracting the serial features of stock data. For the prediction results of different models, the N-BEATS model is optimal for COR, R2, MAPE, and RMSE, which are 0.9998, 0.9996, 0.02%, and 0.8242, respectively. This study demonstrates that N-BEATS is an effective computational tool in stock market forecasting. Our future work will combine the idea of causal statistical learning to enhance the interpretability of time series prediction.

References

1. KAWALLER, I. G., KOCH, P. D., & KOCH, T. W. (1987). The Temporal Price Relationship between S&P 500 Futures and the S&P 500 Index. *The Journal of Finance*, *42*(5), 1309–1329. https://doi.org/10.1111/j.1540-6261.1987.tb04368.x
2. Strader, T. J., Rozycki, J. J., Root, T. H., & Huang, Y.-H. (John). (2020). Machine Learning Stock Market Prediction Studies : Review and Research Directions. *Journal of International Technology and Information Management*, *28*(4), 63–83.
3. Cavalcante, R. C., Brasileiro, R. C., Souza, V. L. F., Nobrega, J. P., & Oliveira, A. L. I. (2016). Computational Intelligence and Financial Markets: A Survey and Future Directions. *Expert Systems with Applications*, *55*, 194–211. https://doi.org/10.1016/j.eswa.2016.02.006
4. Thakkar, A., & Chaudhari, K. (2021, September 1). A comprehensive survey on deep neural networks for stock market: The need, challenges, and future directions. *Expert Systems with Applications*. Elsevier Ltd. https://doi.org/10.1016/j.eswa.2021.114800
5. Shen, Z., Zhang, Y., Lu, J., Xu, J., & Xiao, G. (2020). A novel time series forecasting model with deep learning. *Neurocomputing*, *396*, 302–313. https://doi.org/10.1016/j.neucom.2018.12.084

6. Althelaya, K. A., Mohammed, S. A., & El-Alfy, E. S. M. (2021). Combining deep learning and multiresolution analysis for stock market forecasting. *IEEE Access*, *9*, 13099–13111. https://doi.org/10.1109/ACCESS.2021.3051872
7. Kumar, K., & Haider, M. T. U. (2021). Enhanced Prediction of Intra-day Stock Market Using Metaheuristic Optimization on RNN–LSTM Network. *New Generation Computing*, *39*(1), 231–272. https://doi.org/10.1007/s00354-020-00104-0
8. Oreshkin, B. N., Carpov, D., Chapados, N., & Bengio, Y. (2019). N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. https://arxiv.org/pdf/1905.10437. Oreshkin, B. N., Dudek, G., Pełka, P., & Turkina, E. (2021). N-BEATS neural network for mid-term electricity load forecasting. *Applied Energy*, 293, 116918.
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X.H., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N. (2020). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv preprint, arXiv:2010.11929.
10. Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X.H., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lucic, M., Dosovitskiy, A. (2021). MLP-Mixer: An all-MLP Architecture for Vision. arXiv preprint, arXiv:2105.01601.