

A Review of Stock Price Prediction Based on LSTM and TCN Methods

Ze Zheng^{1,a,*}

¹*Stony Brook Institute, Anhui University, Hefei, Anhui, 230000*

a. zz0904777@126.com

**corresponding author*

Abstract: In financial analysis, stock price prediction is a difficult and important problem that has received a lot of attention from researchers and practitioners in recent years. The application of machine learning and artificial intelligence algorithms to stock price forecasting has demonstrated significant potential for increasing forecasting accuracy. Long momentary memory (LSTM) and transient convolutional networks (TCN) are two famous profound learning calculations that have been generally utilized for stock cost expectation. The most recent approaches to stock price prediction using LSTM and TCN methods are reviewed in this paper. We highlight the most recent research trends in this field and talk about these methods' benefits and drawbacks. Additionally, we discuss potential future research directions in this field. The survey is expected to give a knowledge into the present status of exploration on stock value forecast and guide specialists and experts in working on the exactness of expectation.

Keywords: LSTM, TCN, machine learning, deep learning

1. Introduction

Predicting stock prices has always been a focus of attention for investors and traders. With the development of machine learning and deep learning, more and more people have begun to investigate the possibility of utilizing these technologies to assist in the prediction of stock prices in recent years. LSTM [1] and TCN [2] are two of the most reliable deep learning algorithms for capturing input-time dependencies and have shown great potential in improving stock price prediction accuracy. Studying stock price prediction based on LSTM [1] and TCN [2] methods has significant implications for the finance industry. Accurate predictions of stock prices using deep learning techniques can help reduce risks associated with investment decisions and improve financial performance. This review will introduce some methods and application cases for predicting stock prices based on machine and deep learning.

2. Predicting Stock Prices Based on Machine Learning

2.1. Time Series Model

Time series models are a common method for predicting stock prices. It uses previous time series data and fundamental factors to predict the future trend of stock prices. ARIMA model, SARIMA

model, VAR model, and MA model are common time series models that can be used to predict stock prices. These models can capture the trend of price fluctuations and provide price forecasts for a period of time in the future [3].

2.1.1. Application Cases of Time Series Model

Time series models are widely used in stock price prediction [4], which can use historical data to predict stock price trends in the future and offer investors reference opinions. Research has shown that time series models can effectively predict changes in stock prices, especially for seasonal and cyclical fluctuations in stock prices.

2.1.2. Advantages and Drawbacks of Time Series Model

The advantages of time series models lie in their simple modeling and high prediction accuracy [4], making them one of the important tools in the field of stock price prediction. Simultaneously, time series models likewise have a few disadvantages, for example, the solidness of time series information, the variety of impacting factors, and in some cases the expectation exactness can't completely meet the prerequisites.

2.2. Machine Learning Algorithms

Simultaneously, time series models likewise have a few disadvantages, for example, the solidness of time series information, the variety of impacting factors, and in some cases the expectation exactness can't completely meet the prerequisites [5].

3. Predicting Stock Prices Based on Deep Learning

Deep learning is a more advanced technology than machine learning, and due to its ability to process and analyze large-scale and high-dimensional data, more and more people are using deep learning to predict stock prices.

3.1. Recurrent Neural Network (RNN)

Recurrent neural network is time [1]. It can learn about time dependence from past stock prices and predict future trends in stock prices. This method can be used to predict stock prices in the coming days and effectively capture the periodicity and trend of price trends.

3.2. Long Short Term Memory Network (LSTM)

Long short term memory network is a neural network specifically used for processing sequence data [1]. LSTM can remember price trends and periodicity, and predict future stock price trends. LSTM performs better than traditional methods in predicting stock prices, especially in market environments with volatile stock prices, where it is more robust [1].

A special kind of RNN known as long short term memory (LSTM) is used to solve the problem of gradient explosion and disappearance during long sequence training [1]. In a nutshell, LSTM can outperform standard RNNs in longer sequences [1].

3.2.1. There are Three Main Stages within LSTM

Neglecting stage. This stage predominantly includes specifically failing to remember the info passed in from the past hub. "Forgetting the unimportant and remembering the important," as it is commonly known. The computed z_f — f for forget—is utilized as a forget gate to regulate the C_{t-1}

of the previous state, as depicted in Fig. 1 [6].

Choose the stage of memory. The inputs from this stage are selectively "remembered" by this stage. The primary objective is to choose and remember the input x^t . Record only the important things, and forget less about the less important things. The previously calculated value of z is used to represent the content of the current input and controls the selected gating signal (i for information).

The C^t transmitted to the next state is the sum of the results obtained in the previous two steps. The first formula in the figure above is this one.

Stage of output. Which outputs will be considered the current state will be determined at this stage. The majority of its control comes from z^o . A tanh activation function was used to change the scale of the c^o obtained in the previous stage.

The output y^t is frequently obtained through h^t changes, just like with normal RNN [1].

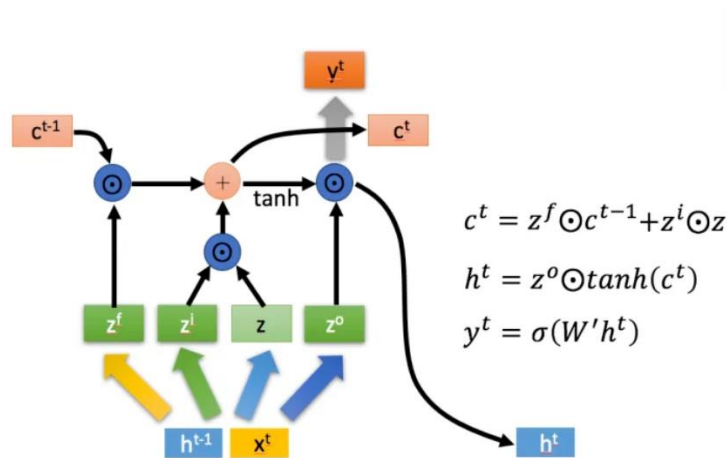


Figure 1: LSTM neural network structure diagram [6].

3.3. Time Convolutional Network(TCN)

Combining RNN and CNN architectures, TCN [2] is a variant of the convolutional neural network utilized for sequence modeling tasks [7]. On a variety of tasks and datasets, the preliminary evaluation of TCN demonstrates that simple convolutional structures perform better than typical cyclic networks (like LSTM) and have a longer effective memory [8].

3.3.1. Residual Block

The most significant change to the basic TCN model is the switch from a one-dimensional causal convolution layer to a two-layer residual block with the same expansion factor and residual connection as the basic building block. Let's take into consideration a layer from the basic model with an expansion factor d_2 of 2 and a kernel size k_3 of 3.

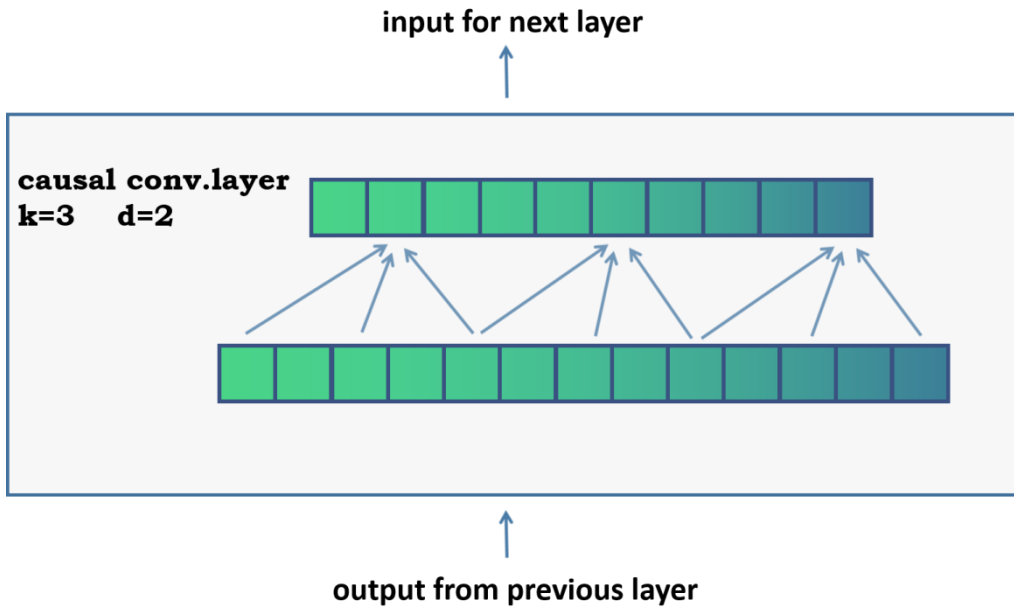


Figure 2: Basic model [8].

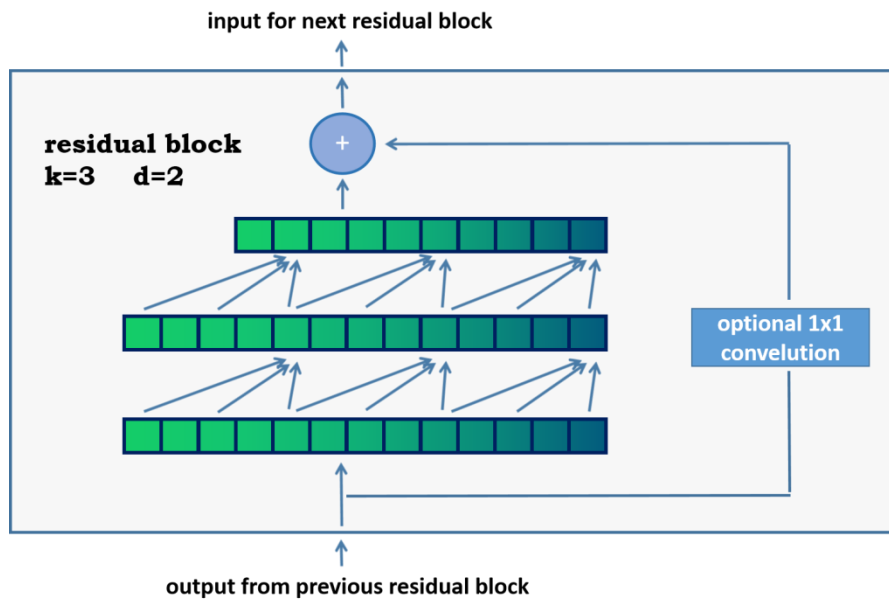


Figure 3: Advanced model [8].

In order to generate the input for the subsequent block, the inputs of the two convolutional layers will be combined with the input of the residual block. For all interior blocks of the organization, or at least, all blocks with the exception of the first and last, the info and result channel widths are something very similar, i.e., num_channels. Since the first convolution layer of the first residual block and the second convolution layer of the last residual block might have different widths of the input and output channels, it might be necessary to adjust the residual tensor's width with 1 Convolution is finished. As depicted in Fig. 3.

The calculation of the minimum number of layers required for full coverage is affected by this change. Presently we should consider the number of leftover blocks that are expected to accomplish total responsive field inclusion. Because it consists of two such layers, adding a residual block to a

TCN will increase the receptive field width by twice as much as adding a basic causal layer [2]. As a result, the following formulae can be used to determine the total size of the sensing region r of a TCN with an expanding base b , the kernel size kb , and the number of residual blocks: $n: r = 1 + \sum_{i=1}^{n-1} 2 * (k - 1) * b^i = 1 + 2 * (k - 1) * \frac{b^n - 1}{b - 1}$

This causes input_lenth of the minimum number of residual blocks n for the full history coverage of length is:

$$n = \log_b \left(\frac{(1-1)*(b-1)}{(k-1)*2} + 1 \right)$$

3.3.2. Activation, Normalization, Regularization

An activation function that introduces nonlinearity above the convolution layer is required for TCN to be more than just an overly complicated linear regression model. After the two convolution layers, the ReLU activation is added to the residual block. Weight normalization is applied to each convolution layer to normalize the input of hidden layers, which can prevent issues like gradient explosions. Dropout is used to introduce regularization after each convolution layer of each residual block to prevent overfitting. The final residual block is shown in Fig. 4.

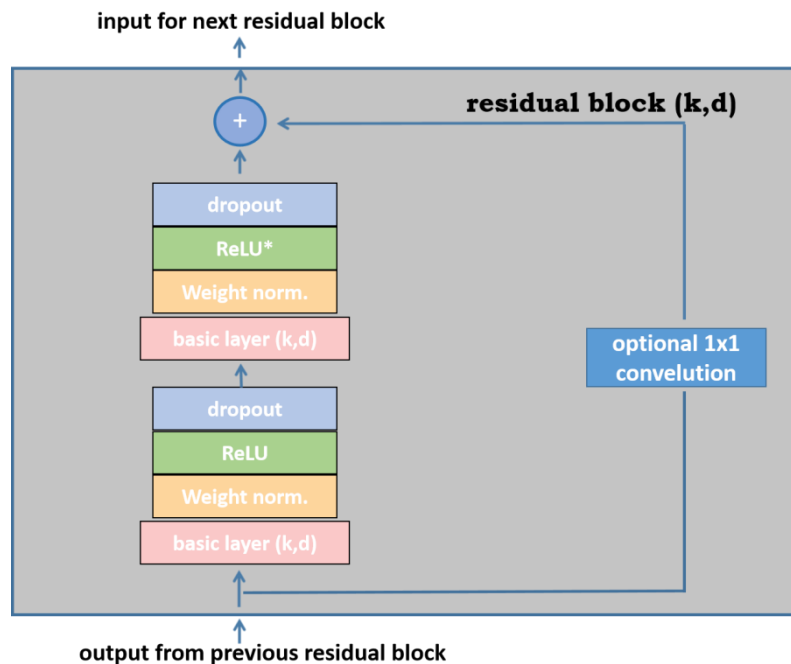


Figure 4: Final residual block [8].

The asterisk in the second ReLU unit in the figure above indicates that it exists in every layer except the last one, because we hope that the final output can also be negative.

3.3.3. Final Model

Fig.5 shows our final TCN model, where l equals input_ Length, k equals kernel_ Size, b equals dimension_ Base, $k \geq b$ and has a minimum residual block number of complete historical coverage n , where n can be calculated based on the above parameters.

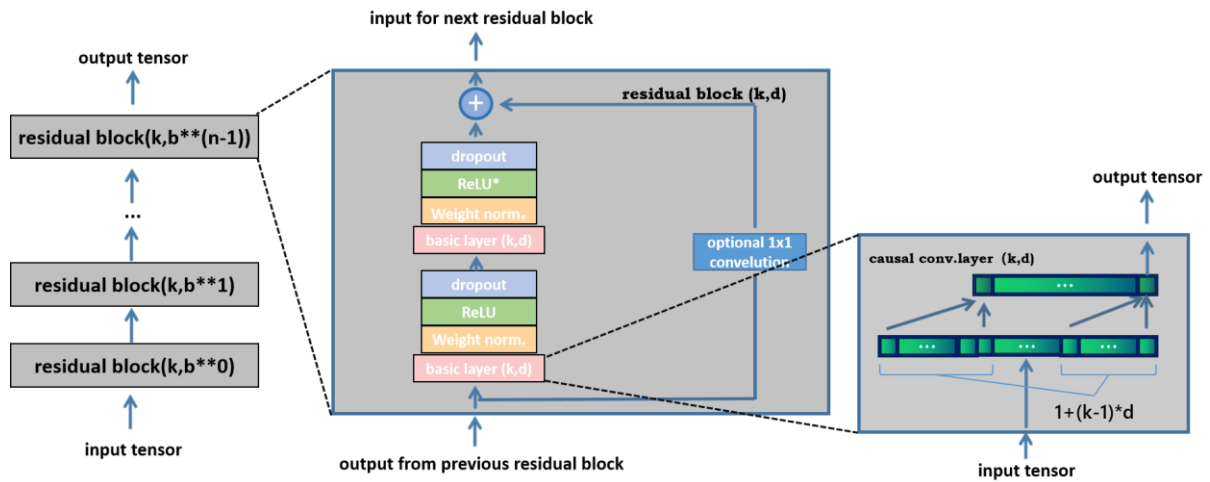


Figure 5: final model [8].

4. Application Cases of LSTM and TCN

The first is the prediction of a single stock: By analyzing the historical price data of a certain stock, predict its future price trend [9]. The second is the prediction of stock market indices: By analyzing the historical price data of a certain market index (such as the Dow Jones Index, NASDAQ Index, etc.), predict the future price trend of the index. And the next is the prediction of multiple stocks: By analyzing the historical price data of multiple stocks and identifying their correlations, predict their future price trends [9]. There are also the prediction of stock volatility: Predict future volatility and stock trading volume by analyzing a stock's historical price data: Predict the stock's future trading volume trend by analyzing its historical price data.

5. Advantages and Drawbacks

5.1. Several Advantages of Using LSTM and Obvious Drawbacks to Using LSTM

High accuracy: LSTM can learn and capture long and short-term dependencies data, thus achieving high prediction accuracy [10]. **Able to handle nonlinear relationships:** Due to the influence of many factors on stock prices, LSTM [1] can handle nonlinear relationships between different variables, which traditional linear models are difficult to handle [10]. **Strong applicability:** LSTM can be applied to various types of stock price prediction problems, including single stock prediction, stock market index prediction, multi stock prediction, volatility prediction, and trading volume prediction. **Model interpretability is good:** Compared to other models (such as neural networks), the LSTM model has better interpretability and can be visualized to understand the predicted results.

High training difficulty: LSTM training and adjustment involve many hyperparameters, such as learning rate, Dropout ratio, etc., which need to be adjusted repeatedly to obtain a better prediction effect. **High resource consumption:** Due to the need for multiple iterations in the training process of LSTM, the required computational resources are relatively large and the running time is relatively long. **High quality requirements for datasets:** LSTM is suitable for processing high-quality datasets. If the input data quality is not high, LSTM will be affected, leading to inaccurate prediction results [10].

5.2. Several Advantages of Using TCN and Obvious Drawbacks to Using TCN

TCN can capture long-term dependencies in time series and has good predictive ability for complex

time series data. TCN uses convolutional neural networks, which can process temporal information in time series through convolutional operations, avoiding numerical instability caused by step-by-step calculations in traditional RNN models. Capable of automatically learning features in time series data, thereby reducing dependence on feature engineering.

Data storage during the evaluation period: TCN needs to use the original sequence until the valid historical record length, so more storage space may be required during the evaluation period.

Potential parameter changes for domain transfer: The number of historical records required for predicting models in different fields may vary. Therefore, when transferring the model from a domain that only requires minimal memory (i.e., smaller k and d) to a domain that requires longer memory (i.e., larger k and d), TCN may not have a sufficiently large receiving field.

6. Conclusion

LSTM and TCN, as deep learning methods, have been widely used. Research has shown that the application of methods such as LSTM and TCN can effectively analyze time series data and improve the accuracy and generalization ability of stock price prediction. At the same time, for the existing problems such as overfitting and low data quality, this paper also proposes some coping strategies and methods. It should be noted that deep learning methods such as LSTM and TCN still face many challenges, such as balancing prediction accuracy and efficiency, processing multimodal data, and the influence of complex factors. Future research directions include improving the stability and generalization ability of stock price forecasting by introducing enhanced learning, further optimizing the hyperparameter of deep learning methods, and so on. We believe that in future development, deep learning methods such as LSTM and TCN will receive more in-depth research and application.

References

- [1] Hochreiter S, Schmidhuber J. Long-term memory networks for neural machine translation[J]. *Neural Networks*, 1997, 9(8):507-536
- [2] Mei P, Zhang J, Liu Z, et al. Transformer-based continuous convolutional network for image classification[C]//*Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017: 779-788.
- [3] Liu, Y., Du, B., Dong, Y., Liang, L., & Wang, Z. (2019). Empirical study of stock price prediction using ARIMA model and machine learning techniques. *SHS Web of Conferences*, 67, 04010. doi: 10.1051/shsconf/20196704010
- [4] Bollerslev, E. H., & Engle, R. F. (1986). On the existence of autoregressive conditional heteroskedasticity. *Econometrica*, 54(2), 370-392.
- [5] Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3), 210-229.
- [6] Chen cheng (2018). LSTM that everyone can understand. *Quora*
- [7] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.
- [8] Shaojie Bai, J. Zico Kolter, Vladlen Koltun.(2018). An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *ArXiv:1803.01271v2 [cs.LG]*, 2018, 4.
- [9] Hossain, M. S., Islam, M. T., & Ahmad, M. (2019). A review on stock market prediction using artificial neural networks. *Cogent Business & Management*, 6(1), 1694492. doi: 10.1080/21642583.2019.1694492
- [10] Venkatesh, B., & Nayyar, A. (2016). A review of time series forecasting using neural networks applied to stock market data. *Expert Systems with Applications*, 46, 164-181. doi: 10.1016/j.eswa.2016.04.032