entropy

Article

A TCN-Linear Hybrid Model for Chaotic Time Series Forecasting

Mengjiao Wang * and Fengtai Qin

School of Automation and Electronic Information, Xiangtan University, Xiangtan 411105, China; 202121622930@smail.xtu.edu.cn
* Correspondence: wangmj@xtu.edu.cn or wangmj@smail.xtu.edu.cn

Abstract: The applications of deep learning and artificial intelligence have permeated daily life, with time series prediction emerging as a focal area of research due to its significance in data analysis. The evolution of deep learning methods for time series prediction has progressed from the Convolutional Neural Network (CNN) and the Recurrent Neural Network (RNN) to the recently popularized Transformer network. However, each of these methods has encountered specific issues. Recent studies have questioned the effectiveness of the self-attention mechanism in Transformers for time series prediction, prompting a reevaluation of approaches to LTSF (Long Time Series Forecasting) problems. To circumvent the limitations present in current models, this paper introduces a novel hybrid network, Temporal Convolutional Network-Linear (TCN-Linear), which leverages the temporal prediction capabilities of the Temporal Convolutional Network (TCN) to enhance the capacity of LSTF-Linear. Time series from three classical chaotic systems (Lorenz, Mackey–Glass, and Rossler) and real-world stock data serve as experimental datasets. Numerical simulation results indicate that, compared to classical networks and novel hybrid models, our model achieves the lowest RMSE, MAE, and MSE with the fewest training parameters, and its R^2 value is the closest to 1.

Keywords: chaos prediction; time series forecasting; neural networks

1. Introduction

Chaotic research constitutes an interdisciplinary field [1], encompassing theories of dynamical systems, nonlinear dynamics, and complex systems [2]. The significance of chaos theory lies in its elucidation of non-periodic behaviors and unpredictable characteristics within numerous systems [3]. Notably, chaotic systems demonstrate extreme sensitivity to initial conditions, where minor initial discrepancies can lead to significant deviations in system trajectories, resulting in long-term unpredictability while retaining short-term predictability [4]. This phenomenon marks chaos as a key feature of complex system behaviors, offering new perspectives for our understanding and analysis of these systems [5].

Time series forecasting involves predicting future outputs using historical information and future input signals [6], reflecting the dynamic changes of the system in the future, and holds broad research prospects [7]. In the classical forecasting domain, mathematical models with rigorous derivations offer good interpretability; however, unknown parameters in the system increase the difficulty of modeling [8]. Linear or single-degree-of-freedom dynamic systems are easier to predict, whereas, due to their sensitivity to initial conditions—characteristic of chaos—chaotic time series are more challenging to forecast [9].

The advent of data-driven modeling techniques has posed challenges for researchers, while the evolution of neural networks, particularly deep learning, bolstered by advancements in computer hardware, has opened new avenues for automated data analysis. Amaranto et al. [10], for instance, developed B-AMA (Basic dAta-driven Models for All), a flexible and easy-to-use tool for both non-expert users and more experienced developers. As for deep learning, traditional Convolutional Neural Networks (CNNs) [11] have

check for updates

Citation: Wang, M.; Qin, F. A TCN-Linear Hybrid Model for Chaotic Time Series Forecasting. *Entropy* **2024**, *26*, 467. https://doi.org/10.3390/e26060467

Academic Editor: Mohammad Reza Rahimi Tabar

Received: 17 April 2024
Revised: 25 May 2024
Accepted: 27 May 2024
Published: 29 May 2024

been proven effective in the domain of image recognition, while Recurrent Neural Networks (RNNs), due to their unique neural unit structure, exhibit superior performance in processing sequential data [12]. Despite many RNN variants optimizing their performance by introducing gating mechanisms, issues such as vanishing or exploding gradients and high computational costs due to the inability to compute in parallel still persist [13]. Consequently, since the introduction of the Transformer [14] network in 2017, its parallel processing capabilities based on the attention mechanism and its ability to capture long-distance dependencies have demonstrated remarkable abilities in natural language processing tasks, making it and its variants the focus of current neural network research. However, Zeng et al. questioned the Transformer network's capabilities in time series forecasting [15]. They pointed out that the attention mechanism could lead to the loss of temporal scale information, which is crucial for time series forecasting. They introduced a structurally simple LSTF-Linear network that outperformed the Transformer network on multiple test sets.

The study of time series forecasting for dynamic system states holds practical value [16]. In recent decades, a considerable amount of research has been dedicated to applying neural networks to chaotic time series forecasting and related application domains. Sun et al. [17] trained an encoder–decoder with LSTM units to perform chaos forecasting on a five-degree-of-freedom duffing oscillator system. The numerical simulations revealed that the LSTM ED model can accurately predict chaotic time series with limited data, achieving a prediction window twice the size of the observation window. Uribarri et al. [18] found out that under certain conditions, Long Short Term Memory networks can learn to forecast time series from chaotic systems by generating an embedding in their inner state that is topologically equivalent to the original strange attractor. Sangiorgio et al. [19] implemented a multi-step approach to predict an entire interval of future values, and compared the performances of various neural network architectures in real-world cases. Ref. [20] recommends training LSTM networks without the teacher forcing them to improve accuracy and robustness, ensuring a more uniform distribution of the predictive power within the chaotic attractor. Ref. [21] found that recurrent architectures of networks are more suitable for learning the non-stationary dynamics caused by structural noise. Pathak et al. [22] presented a parallel scheme with an example implementation based on the Reservoir Computing (RC) paradigm that offered a new direction in the field of chaos prediction. Ref. [23] found that RNNs trained via backpropagation through time show superior forecasting abilities and capture the dynamics of reduced order systems well. Patel et al. [24] found that machine learning shows excellent performance in predicting the long-term behavior of a non-stationary dynamical system.

In recent studies, hybrid models that leverage the strengths of various models through parallel or sequential combinations have demonstrated superior performance, which is emerging as a new trend in research. Xia et al. [25] presented a stacked GRU-RNN for the prediction of renewable energy generation and electricity load; the experimental results demonstrated that the models achieved an accurate energy prediction for effective smart grid operation. Lazcano et al. [26] combined the characteristics of a GCN and a Bi-LSTM network to forecast the price of oil and reached a low error of RMSE, MSE, and MAPE. Cao et al. [27] established a new hybrid time series forecasting method by combining the EMD and CEEMDAN algorithms with the LSTM neural network for financial time series, which exhibited a more accurate predictive performance than other similar models. Fu et al. [28] designed a deep temporal inception module and gated recurrent unit network (DTIGNet), which demonstrated superior accuracy and performance in predicting chaotic time series, outperforming other methods based on six evaluation metrics. The significance of chaotic time series prediction also extends to its application in fields such as weather forecasting [29,30], the stock market [31,32], wind power [33,34], and traffic flow analysis [35,36], highlighting its critical role across various domains.

Building on the analysis provided, we introduce a novel hybrid neural network architecture, the TCN-Linear model. This model harnesses the advantages of dilated and causal convolutions within the TCN network, thereby expanding the model capacity of LSTF-Linear and enhancing its specialized learning capabilities for diverse time series data. It circumvents the gradient issues associated with RNNs and the loss of temporal scale information attributed to the attention mechanisms in Transformers, achieving a higher prediction accuracy with a reduced parameter count for training. The remainder of the paper is organized as follows. Section 2 introduces the theoretical underpinnings and the development of the model. Section 3 describes the experimental setup and results. Section 4 concludes the work and offers perspectives on future research.

2. Proposed Model

2.1. TCN

Bai et al. [37] introduced a novel Temporal Convolutional Network (TCN) that adapts convolutional networks for the processing of time series data. This model leverages a proposed causal convolution method to capture local dependencies within sequence data, ensuring temporality, and employs dilated convolutions to expand its receptive field for better learning of data correlations. Additionally, it utilizes convolutional operations for efficient parallel computation, making it suitable for large-scale data processing.

2.1.1. Causal Convolution

Causal convolution, as shown in Figure 1, is one of the core concepts of TCN. To ensure that convolution operations only utilize past information, causal convolution employs zero-padding at the beginning of the sequence. This technique ensures that the output at each time step is influenced solely by that point and its preceding inputs. Such an approach prevents forward leakage of information and maintains temporal alignment between the input and output sequences.

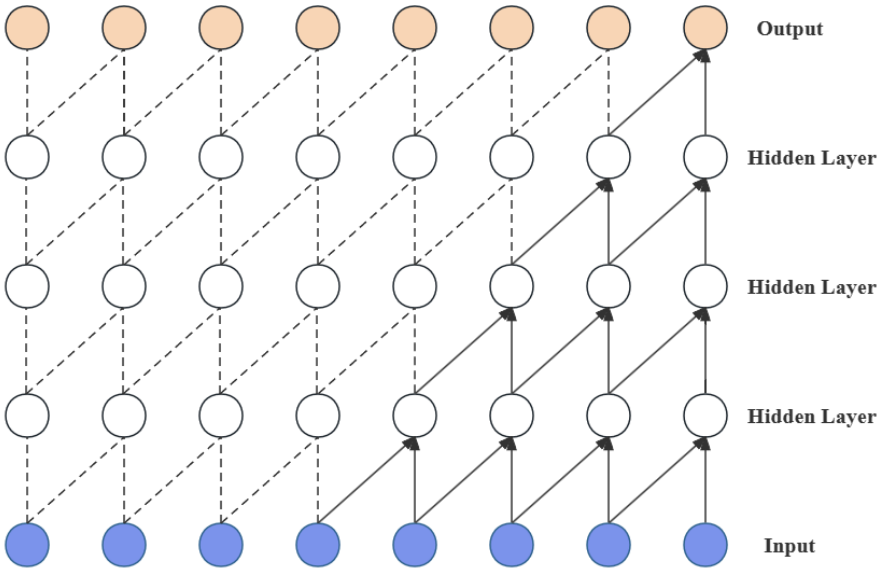


Figure 1. Structure of causal convolution.

2.1.2. Dilated Convolution

Dilated convolution represents another crucial component within TCN, serving as an extension of traditional convolution aimed at enlarging the receptive field of the convolutional layers. This enlargement enables the network to capture temporal dependencies over longer ranges. In dilated convolution, zeros are inserted between the elements of the convolution kernel (i.e., the dilation rate), allowing the network to cover a larger input area without an increase in the number of parameters or the computational complexity; the structure of dilated convolution is shown in Figure 2.

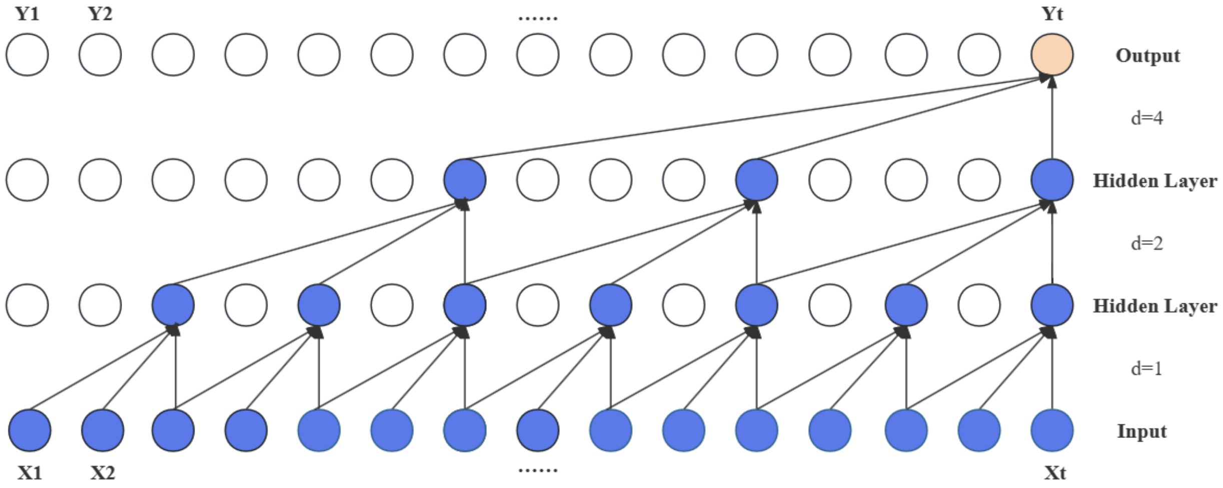


Figure 2. Structure of dilated convolution.

With the increase in network depth, the dilation rate can be progressively increased, enabling deeper convolutional layers to possess an extensive receptive field. Consequently, the network can effectively learn long-term temporal dependencies without sacrificing temporal resolution.

2.1.3. Residual Connection

TCN mitigates the effects of gradient vanishing and explosion in deep networks to some extent. This model introduces straightforward direct connection channels, allowing the network to learn identity mapping, as shown in Figure 3. This ensures that the performance of deep networks does not degrade more than that of their shallower counterparts, and prevents the initial data weight increase caused by dimension changes during the input processing phase.

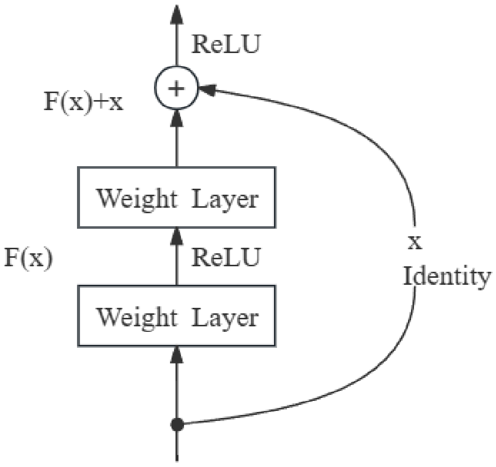


Figure 3. Structure of residual connection.

2.2. *LSTF-Linear*

LSTF-Linear is a simple direct multi-step model that operates via a temporal linear layer. The fundamental approach of LTSF-Linear employs a weighted sum operation to directly predict future values by regressing on historical time series data (as illustrated in Figure 4). The mathematical expression is $\hat{X}_i = WX_i$, where $W \in \mathbb{R}^{T \times L}$ is a linear layer along the temporal axis. $\hat{X}_i$ and X_i are the prediction and the input for each i variate.

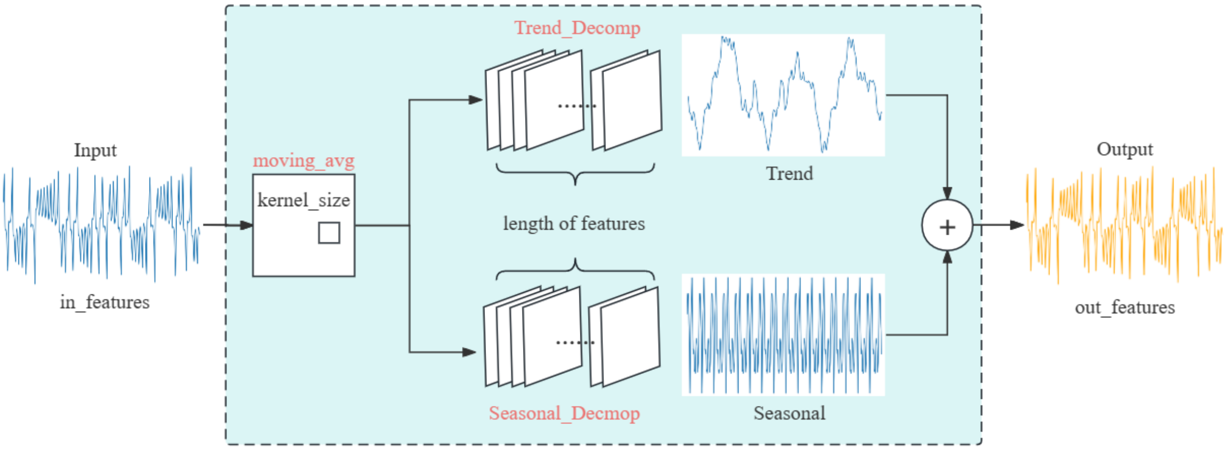


Figure 4. Structure of D-Linear.

Specially, D-Linear (shown in Figure 4) is a hybrid of a seasonal decomposition encoder–decoder and the Linear network which decomposes the original data into seasonal and trend components through a moving average kernel. Subsequently, each component is processed using a single linear layer, and the output results are summed to obtain the final prediction. This strategy enhances the model's performance when the data exhibit clear trends, which coincidentally can be identified within the phase diagrams of chaotic systems.

2.3. TCN-Linear

To further improve model capacity and prediction accuracy, a new hybrid model for chaotic time series prediction is proposed in this paper, named TCN-Linear, which is shown in Figure 5. We tried to improve the structure of D-Linear by fusing it with TCN. This model is constructed with several Residual Block modules and a D-Linear network, where each Residual Block contains two Dilated Causal Conv, two WeightNorm layers, two ReLu layers, and two Dropout layers. In the Dilated Causal Convolution within these blocks, the dilation factor (d) is set to values in the set {1,2,4} and the output of each current layer will serve as the input of the next layer. Finally, the prediction is output by the combination of the Decomposition scheme and the Linear layers.

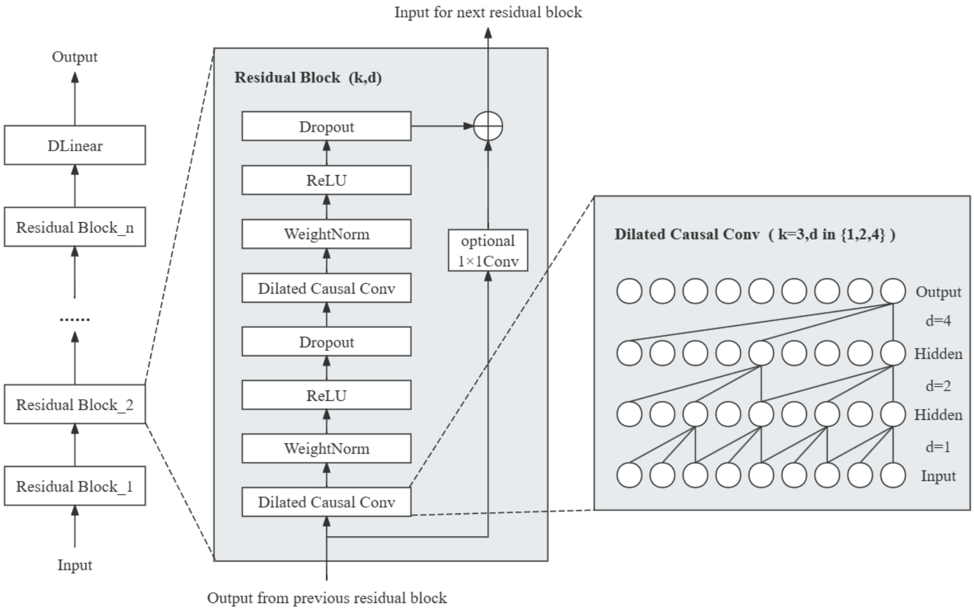


Figure 5. Structure of TCN-Linear.

This hybrid architecture is designed to address the complexities of time series data that exhibit both long-term dependencies and seasonal patterns, making the model versatile across different time series forecasting tasks. Furthermore, TCNs offer efficient parallel computation, significantly reducing both the number of training parameters and the training time compared to traditional RNN- and Transformer-based solutions. This efficiency, when combined with the direct computational characteristics of LSTF-Linear networks, renders the TCN-Linear model particularly suitable for large-scale time series datasets.

In the model, we employed the mean squared error (MSE) as the loss function—a commonly used metric in regression problems that calculates the mean squared error between the predicted values and the actual values. The Adaptive Moment Estimation (ADAM) optimizer [38] was utilized for network training due to its efficiency, robustness, and ease of configuration, making it one of the preferred optimizers in deep learning applications. Its role is to adjust the network parameters to minimize the loss function. Additionally, we adopted early stopping to prevent overfitting. This technique terminates the training process prematurely when the validation error stops decreasing after a certain number of epochs, thereby ensuring the model's generalization capability.

3. Experimental Evaluation

In this section, we evaluate the predictive capability, training cost, and applicability to real financial data of the proposed model using three classical chaotic systems (the Lorenz system, the Mackey–Glass system, and the Rossler system) and real-life stock data.

3.1. Dataset

3.1.1. Lorenz

The Lorenz equations were introduced in 1963 by Edward N. Lorenz [39] during his research on atmospheric convection, marking the inception of chaotic research. The Lorenz model is a dynamic system comprising three ordinary differential equations, representing the three-dimensional state of convective rolls.

$$\begin{cases} \dfrac{dx}{dt} = -\sigma(x - y) \\ \dfrac{dy}{dt} = -xz + rx - y \\ \dfrac{dz}{dt} = xy - bz \end{cases} \tag{1}$$

when the parameters are set to σ = 10, b = 8/3, r = 28, the system behaves in a chaotic state. In this state, with initial values set to $x(0) = 1$, $y(0) = 0$, and $z(0) = 1$, we generated time series for the system's three variables by employing the ODE45 integration method at a sampling frequency of 200 Hz within the time interval (t in [0, 55]), which had 11,000 points. We removed the first 3000 transient values and divided the remaining 8000 data points into training, validation, and test sets in a 6:2:2 ratio.

3.1.2. Mackey–Glass

The Mackey–Glass system, introduced in 1977 by Michael C. Mackey and Leon Glass [40], is a delay differential equation frequently utilized as a benchmark in chaotic time series analysis.

$$\dot{x}(t) = -bx(t) + \frac{ax(t-\tau)}{1 + x^c(t-\tau)} \tag{2}$$

when the parameters are set to a = 0.2, b = 0.1, c = 10, τ = 17, the system behaves in a chaotic state. In this state, we generated a time series for the system by employing the ODE45 integration method at a sampling frequency of 10 Hz within the time interval (t in [0, 1100]), which had 11,000 points. We removed the first 3000 transient values and divided the remaining 8000 data points into training, validation, and test sets in a 6:2:2 ratio.

3.1.3. Rossler

The Rossler model, introduced in 1976 by the German biophysicist Otto E. Rössler [41], is a chaotic system. Compared to the Lorenz model, the equations of the Rossler model are simpler, and its phase diagram exhibits a clear spiral structure. This demonstrates that complex chaotic behavior can be observed even in exceedingly simple systems.

$$\begin{cases} \dfrac{dx}{dt} = -(y+z) \\ \dfrac{dy}{dt} = -x + ay \\ \dfrac{dz}{dt} = b + z(x-c) \end{cases} \quad (3)$$

when the parameters are set to a = 0.2, b = 0.2, c = 5.9, the system behaves in a chaotic state. In this state, with the initial values set to $x(0) = 0$, $y(0) = 0$, and $z(0) = 0$, we generated time series for the system's three variables by employing the ODE45 integration method at a sampling frequency of 50 Hz within the time interval (t in [0, 220]), which had 11,000 points. We removed the first 3000 transient values and divided the remaining 8000 data points into training, validation, and test sets in a 6:2:2 ratio.

3.1.4. Google Stock Price

We collected stock trading data of Google Inc. (San Francisco, CA, USA) from 2014 to 2024 from public databases. This dataset encompasses key financial indicators over a decade, including the opening price, closing price, highest price, lowest price of the day, and trading volume of Google's stock. As illustrated in Figure 6, we utilized the pairplot method from the seaborn library to create a diagonal chart that showcases the relationships between multiple variables, and selected the variables strongly correlated with the closing price as features. Subsequently, we divided the 4858 data points into training, validation, and test sets in a 6:2:2 ratio.

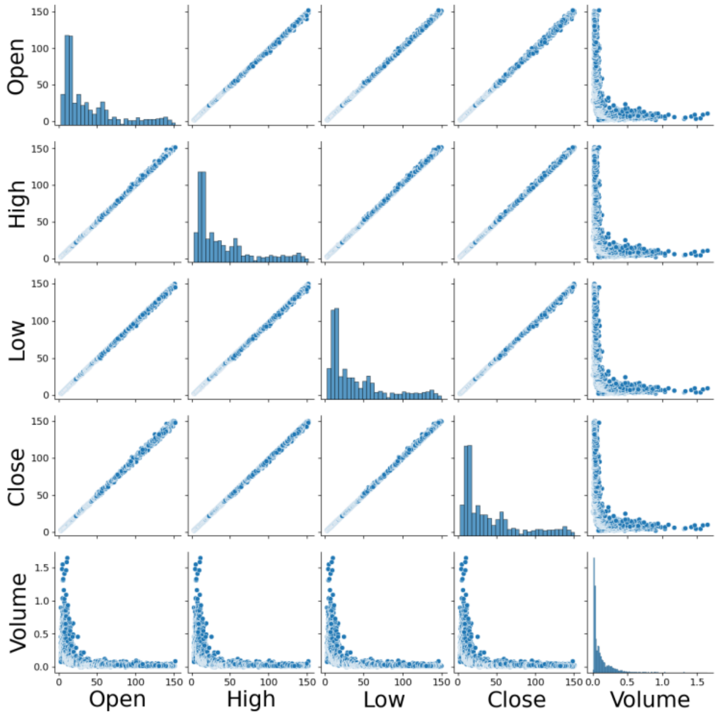


Figure 6. Mutual variables relationship diagram.

3.2. Experiment Settings

3.2.1. Experimental Configuration

The main hardware environments of the experiments were as follows: AMD R5-5600X CPU, NVIDIA RTX 3080Ti 16 GB GPU, and 16 GB of RAM, and the operating system of the computer was Windows 10. The software configuration used for simulation experiments in this study included CUDA Version 12.3, GPU Driver Version 546.29, torch 1.9.0+cu111, and python 3.9.18, and the parameter settings are shown in Table 1.

Table 1. Hyperparameter settings.

Parameters	Value
seed	42
batch_size	32
in_seq_len	24
out_seq_len	12
num_epochs	500
learning_rate	0.001
es_patience	15
lr_patience	5
kernel_size	25
num_layers	1
hidden_size	64
train_ratio	0.6
teaching_forcing_prob	0.75
dropout	0.2
criterion	nn.MSELoss
optimizer	optim.Adam

3.2.2. Prediction Evaluation Index

To evaluate the performance of each model, we set the MAE, MSE, RMSE, MAPE, and R^2, and their definitions are as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|(y_i - \hat{y}_i)| \tag{4}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \tag{5}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{6}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100 \tag{7}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \tag{8}$$

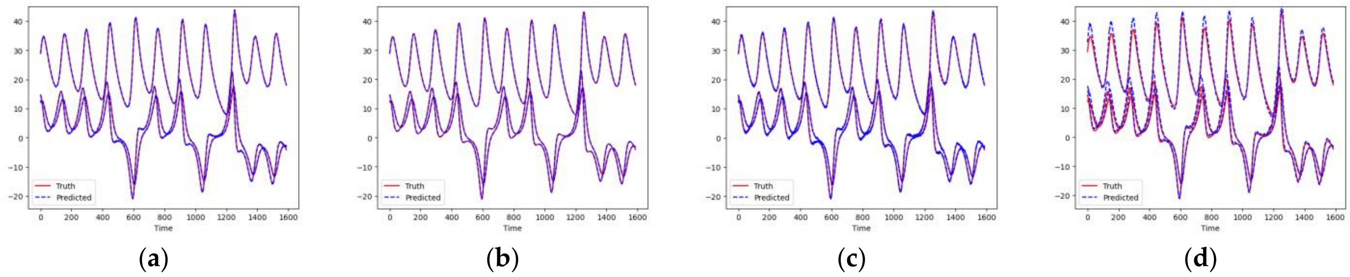
where n is the length of the predicted series, y_i is the true value, $\hat{y}_i$ is the predicted value, and $\overline{y}$ represents the mean of the true value of the sequence, respectively.

3.3. Results

3.3.1. Lorenz

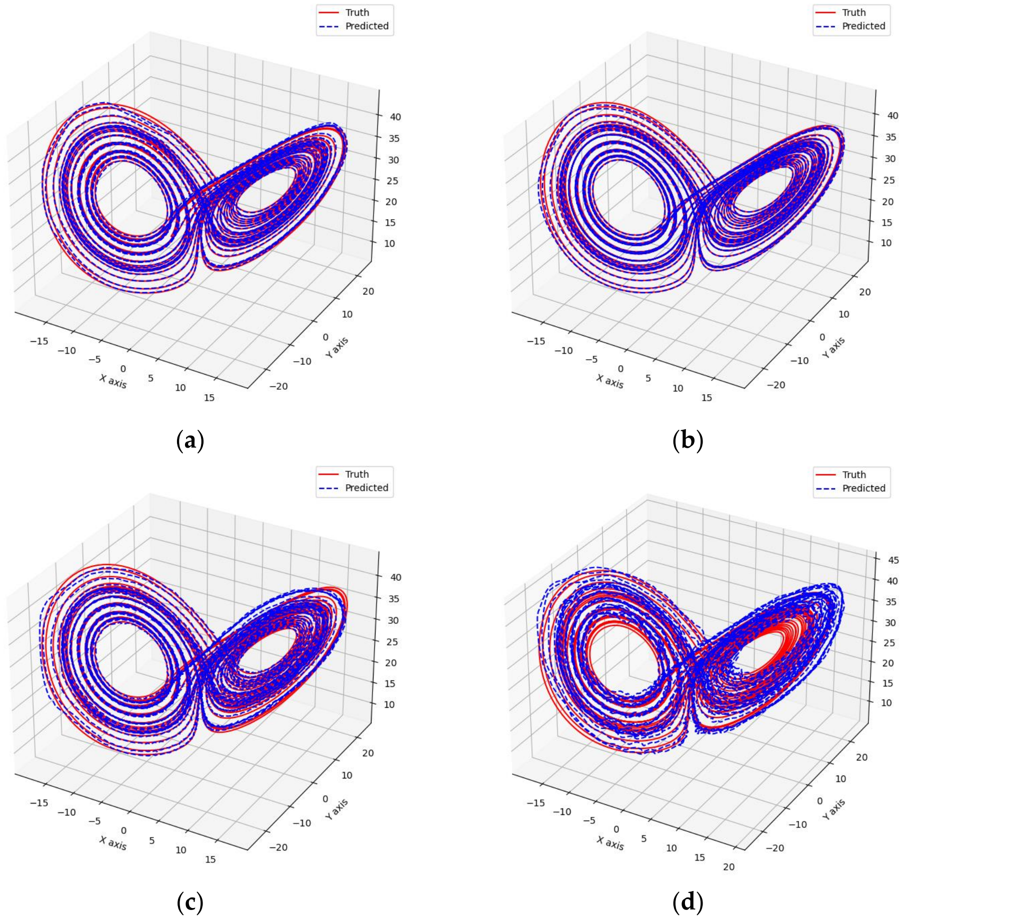
As shown in Figures 7 and 8, these models can capture the dynamic changes of the Lorenz system effectively. Their blue prediction curves align well with the red actual value curves, except for the Transformer network, which exhibits significant noise and fluctuation in its predictions. The results presented in Figure 9 and Table 2 further suggest that the

TCN-Linear network, with the smallest number of training parameters, outperforms the other models to varying degrees across various indexes.



(**a**) (**b**) (**c**) (**d**)

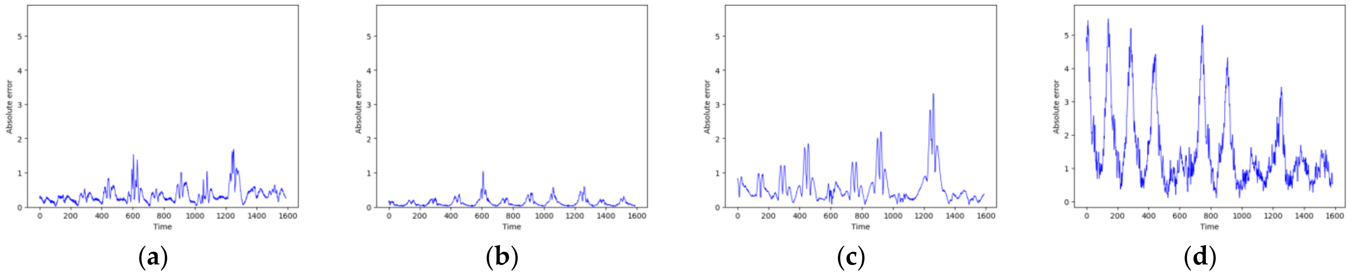
Figure 7. Predicted time series of Lorenz. (**a**) RC; (**b**) TCN-Linear; (**c**) LSTM; (**d**) Transformer.



(**a**) (**b**)

(**c**) (**d**)

Figure 8. Predicted phase diagrams of Lorenz. (**a**) RC; (**b**) TCN-Linear; (**c**) LSTM; (**d**) Transformer.



(**a**) (**b**) (**c**) (**d**)

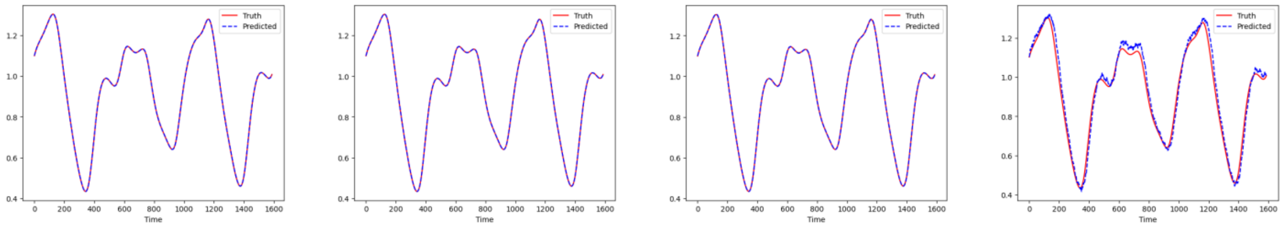
Figure 9. Predicted absolute errors of Lorenz. (**a**) RC; (**b**) TCN-Linear; (**c**) LSTM; (**d**) Transformer.

Table 2. Evaluation scores of Lorenz.

Index	TCN-Linear	Transformer	LSTM	RC
RMSE	0.02595169	1.38334787	0.29731486	0.10683146
MAE	0.01839166	1.03468537	0.19289005	0.07635882
MSE	0.00067349	1.91365147	0.08839613	0.06396773
R^2	0.99999064	0.97296977	0.99877666	0.99997096
parameter	2853	234,435	17,859	13,689
epoch	240	46	112	101

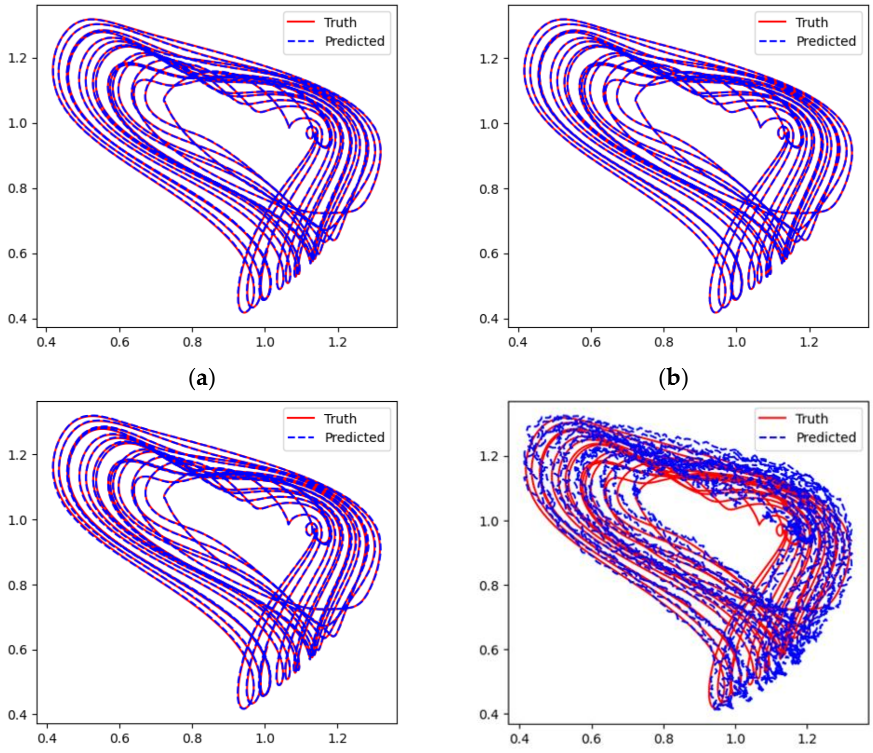
3.3.2. Mackey–Glass

Due to the Mackey–Glass system's time series containing only one dimension, the networks can fit its curves more smoothly, as demonstrated in Figure 10. However, in the phase space reconstruction shown in Figure 11, the Transformer network still exhibits more noise compared to the other models. According to the results in Figure 12 and Table 3, the error values of all models are quite low, with TCN-Linear continuing to exhibit the best performance among them.



(**a**) (**b**) (**c**) (**d**)

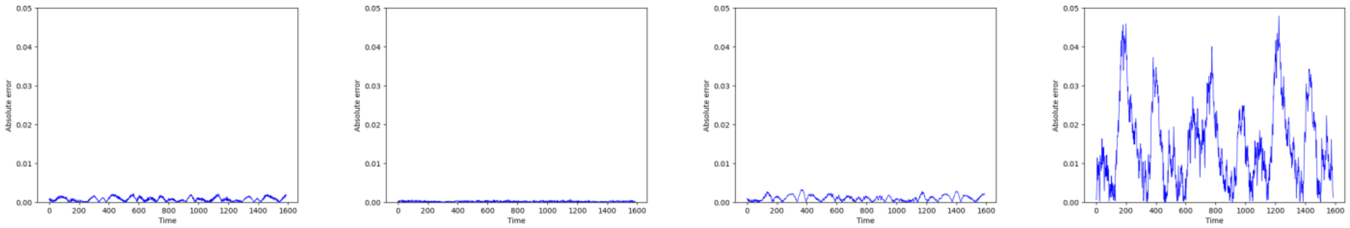
Figure 10. Predicted time series of Mackey–Glass. (**a**) RC; (**b**) TCN-Linear; (**c**) LSTM; (**d**) Transformer.



(**a**) (**b**)

(**c**) (**d**)

Figure 11. Predicted phase diagrams of Mackey–Glass. (**a**) RC; (**b**) TCN-Linear; (**c**) LSTM; (**d**) Transformer.



(**a**) (**b**) (**c**) (**d**)

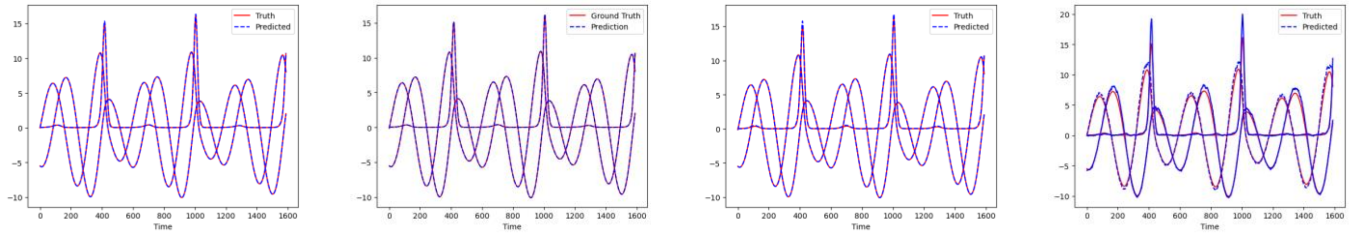
Figure 12. Predicted absolute errors of Mackey–Glass. (**a**) RC; (**b**) TCN-Linear; (**c**) LSTM; (**d**) Transformer.

Table 3. Evaluation scores of Mackey–Glass.

Index	TCN-Linear	Transformer	LSTM	RC
RMSE	0.00018653	0.03591761	0.00122464	0.00094763
MAE	0.00014705	0.02931397	0.00101639	0.00083345
MSE	0.00000003	0.00129008	0.00000150	0.00000105
R^2	0.99999941	0.97798753	0.99997440	0.99996435
parameter	951	234,049	17,217	8735
epoch	76	13	163	69

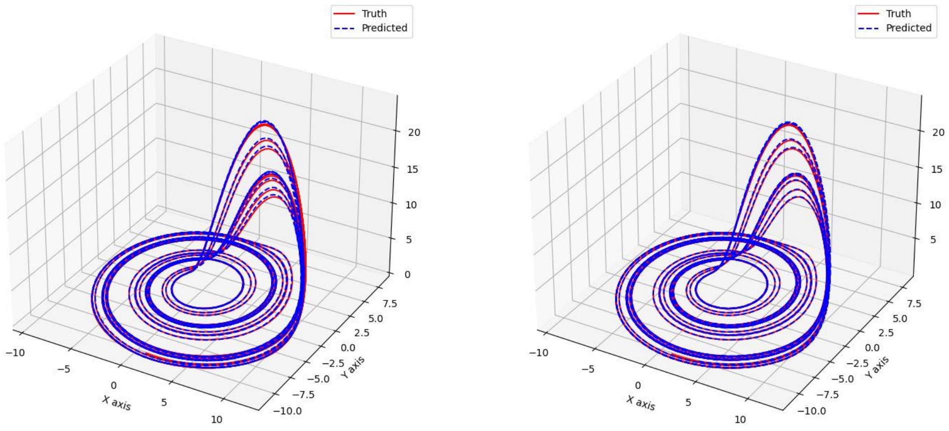
3.3.3. Rossler

Similar to the previous two experiments, the networks demonstrated excellent performance in fitting the time series of the Rossler system, as can be seen from Figures 13–15. In Table 4 the RMSE, MAE, and MSE values were all very low, and the R^2 values were very close to 1. Once again, TCN-Linear emerged as the top performer.



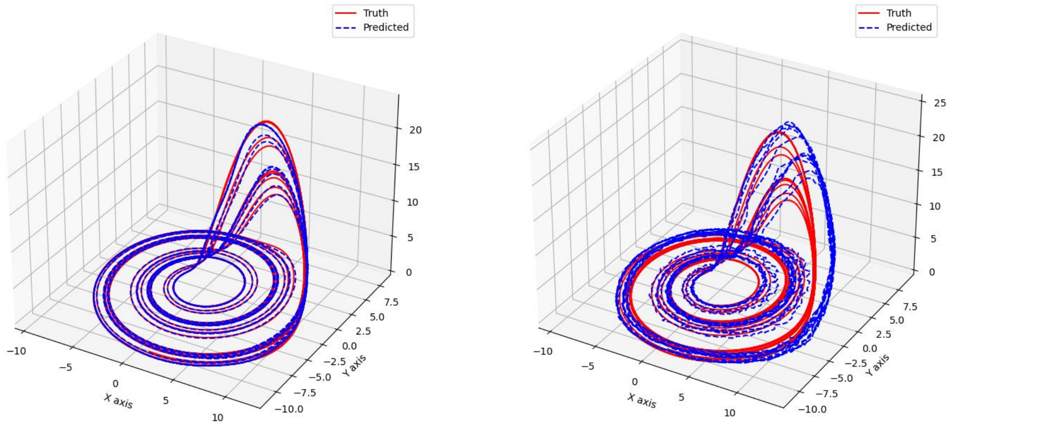
(**a**) (**b**) (**c**) (**d**)

Figure 13. Predicted time series of Rossler. (**a**) RC; (**b**) TCN-Linear; (**c**) LSTM; (**d**) Transformer.



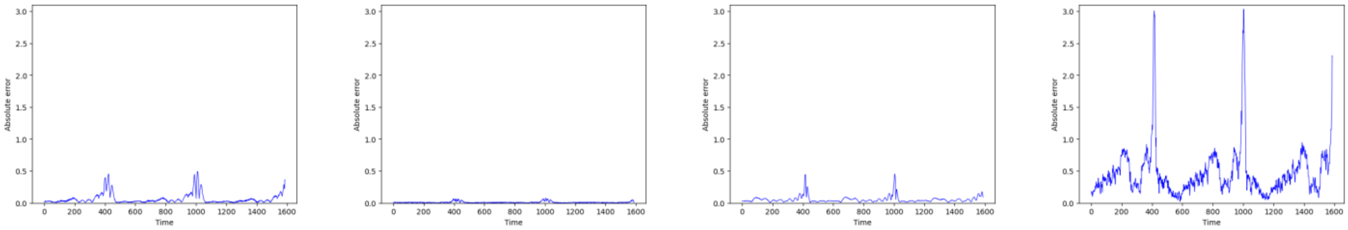
(**a**) (**b**)

Figure 14. *Cont*.



(**c**) (**d**)

Figure 14. Predicted phase diagrams of Rossler. (**a**) RC; (**b**) TCN-Linear; (**c**) LSTM; (**d**) Transformer.



(**a**) (**b**) (**c**) (**d**)

Figure 15. Predicted absolute errors of Rossler. (**a**) RC; (**b**) TCN-Linear; (**c**) LSTM; (**d**) Transformer.

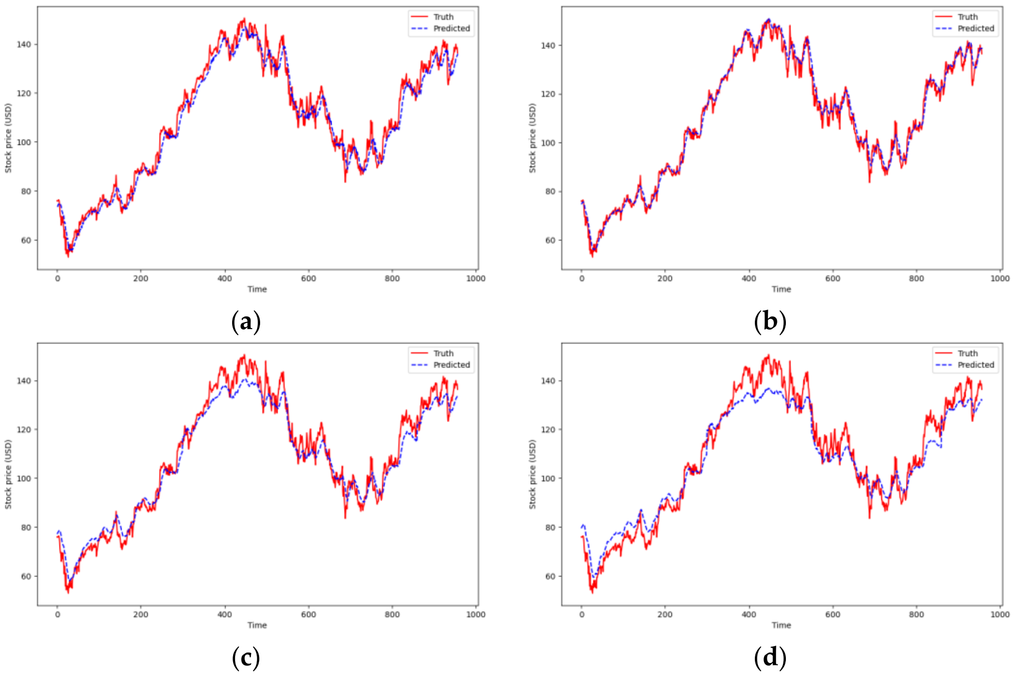
Table 4. Evaluation scores of Rossler.

Index	TCN-Linear	Transformer	LSTM	RC
RMSE	0.01509988	0.52885771	0.06465438	0.04373308
MAE	0.00645125	0.31585521	0.03655762	0.04769310
MSE	0.00022801	0.27969050	0.00418019	0.00374805
R^2	0.99996984	0.97996159	0.99948332	0.99984731
parameter	2853	234,435	17,859	10,176
epoch	93	27	108	147

3.3.4. Google Stock Price

To demonstrate the model's applicability to real financial sequences, we selected Google's stock data from the past decade for analysis and compared TCN-Linear's performance with that of hybrid models CNN-GRU [42], Seq2Seq [43], and Bi-LSTM [44], which have shown promising results in this domain.

Due to the high frequency of fluctuations in real stock data, it is challenging for models to fully learn and fit the actual values without overfitting the data. As depicted in Figure 16, the prediction curves roughly outline the general trend of the stock price. The indexes in Table 5 show a noticeable deterioration compared to the previous chaotic systems; however, the TCN-Linear model still demonstrates excellent predictive capabilities.



(**a**) (**b**) (**c**) (**d**)

Figure 16. Predicted stock price of Google. (**a**) Bi-LSTM; (**b**) TCN-Linear; (**c**) Seq2Seq; (**d**) CNN-GRU.

Table 5. Evaluation scores of the Google stock price.

Index	TCN-Linear	CNN-GRU	Seq2Seq	Bi-LSTM
RMSE	3.820	7.703	6.408	4.366
MAE	2.918	5.730	4.621	4.200
MSE	14.591	16.832	15.973	14.386
R^2	0.976	0.921	0.964	0.953
MAPE	2.793%	5.964%	4.389%	3.763%

4. Conclusions

In this paper, a novel hybrid TCN-Linear model for the prediction of chaotic time series is proposed. To enhance the capacity of the LSTF-Linear model, we integrated it with the Temporal Convolutional Network model, which has long-term memory and parallel computing capabilities, thereby circumventing the gradient issues associated with RNNs and the loss of temporal scale information due to the attention mechanisms in Transformers. Experiments conducted on time series generated by several classical chaotic systems and real stock sequences demonstrate that our model is capable of capturing the future trends of dynamic systems and making accurate predictions. It achieved the lowest error metrics compared to other models, with the R^2 value closest to 1. The novel structure of our model offers fresh insights into solving LTSF problems. However, there are still some limitations in our work. For instance, it is challenging to maintain low error rates in multi-step predictions of multi-dimensional or high-frequency variable data. Moreover, there is still a long way to go in terms of accurately restoring the dynamic behaviors and patterns of chaotic systems. We believe that Recurrent Neural Networks and Reservoir Computing hold promising potential in nonlinear dynamics analysis, so these are areas we aim to explore in future improvements. Our future research will focus on designing models that balance computational resources and prediction accuracy, and applying them to more complex real-world engineering applications such as weather systems, turbulent flow data, and industrial fault diagnosis.

Author Contributions: Conceptualization, M.W. and F.Q.; methodology, M.W. and F.Q.; software, F.Q.; validation, M.W. and F.Q.; formal analysis, M.W. and F.Q.; investigation, M.W.; writing—original draft preparation, F.Q.; writing—review and editing, M.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (Grant No. 62071411).

Data Availability Statement: The datasets generated and/or analyzed during the current study are available from the corresponding author upon reasonable request.

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

References

1. Ramadevi, B.; Bingi, K. Chaotic time series forecasting approaches using machine learning techniques: A review. *Symmetry* **2022**, *14*, 955. [CrossRef]
2. Bayani, A.; Rajagopal, K.; Khalaf, A.J.M.; Jafari, S.; Leutcho, G.D.; Kengne, J. Dynamical analysis of a new multistable chaotic system with hidden attractor: Antimonotonicity, coexisting multiple attractors, and offset boosting. *Phys. Lett. A* **2019**, *383*, 1450–1456. [CrossRef]
3. Khuntia, S.R.; Rueda, J.L.; Meijden, M.A. Forecasting the load of electrical power systems in mid-and long-term horizons: A review. *IET Gener. Transm. Distrib.* **2016**, *10*, 3971–3977. [CrossRef]
4. Tian, Z. Preliminary research of chaotic characteristics and prediction of short-term wind speed time series. *Int. J. Bifurc. Chaos* **2020**, *30*, 2050176. [CrossRef]
5. Pecora, L.M.; Carroll, T.L. Synchronization in chaotic systems. *Phys. Rev. Lett.* **1990**, *64*, 821. [CrossRef] [PubMed]
6. Petropoulos, F.; Apiletti, D.; Assimakopoulos, V.; Babai, M.Z.; Barrow, D.K.; Taieb, S.B.; Bergmeir, C.; Bessa, R.J.; Bijak, J.; Boylan, J.E.; et al. Forecasting: Theory and practice. *Int. J. Forecast.* **2022**, *38*, 705–871.
7. Mosavi, A.; Salimi, M.; Faizollahzadeh Ardabili, S.; Rabczuk, T.; Shamshirband, S.; Varkonyi-Koczy, A.R. State of the art of machine learning models in energy systems, a systematic review. *Energies* **2019**, *12*, 1301. [CrossRef]
8. Bzdok, D.; Nichols, T.E.; Smith, S.M. Towards algorithmic analytics for large-scale datasets. *Nat. Mach. Intell.* **2019**, *1*, 296–306. [CrossRef] [PubMed]
9. Cheng, C.; Sa-Ngasoongsong, A.; Beyca, O.; Le, T.; Yang, H.; Kong, Z.; Bukkapatnam, S.T. Time series forecasting for nonlinear and non-stationary processes: A review and comparative study. *Iie Trans.* **2015**, *47*, 1053–1071. [CrossRef]
10. Amaranto, A.; Mazzoleni, M. B-AMA: A Python-coded protocol to enhance the application of data-driven models in hydrology. *Environ. Model. Softw.* **2023**, *160*, 105609. [CrossRef]
11. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph.* **2019**, *38*, 1–12. [CrossRef]
12. Zhang, J.; Man, K.F. Time series prediction using RNN in multi-dimension embedding phase space. In *SMC'98 Conference Proceedings, Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 98CH36218), San Diego, CA, USA, 14 October 1998*; IEEE: Piscataway, NJ, USA, 1998; Volume 2, pp. 1868–1873.
13. 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]
14. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
15. Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Are transformers effective for time series forecasting? In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 11121–11128. [CrossRef]
16. Lim, B.; Zohren, S. Time-series forecasting with deep learning: A survey. *Philos. Trans. R. Soc. A* **2021**, *379*, 20200209. [CrossRef] [PubMed]
17. Sun, Y.; Zhang, L.; Yao, M. Chaotic time series prediction of nonlinear systems based on various neural network models. *Chaos Solitons Fractals* **2023**, *175*, 113971. [CrossRef]
18. Uribarri, G.; Mindlin, G.B. Dynamical time series embeddings in recurrent neural networks. *Chaos Solitons Fractals* **2022**, *154*, 111612. [CrossRef]
19. Sangiorgio, M.; Dercole, F.; Guariso, G. *Deep Learning in Multi-Step Prediction of Chaotic Dynamics: From Deterministic Models to Real-World Systems*; Springer Nature: Berlin, Germany, 2022.
20. Sangiorgio, M.; Dercole, F. Robustness of LSTM neural networks for multi-step forecasting of chaotic time series. *Chaos Solitons Fractals* **2020**, *139*, 110045. [CrossRef]
21. Sangiorgio, M.; Dercole, F.; Guariso, G. Forecasting of noisy chaotic systems with deep neural networks. *Chaos Solitons Fractals* **2021**, *153*, 111570. [CrossRef]
22. Pathak, J.; Hunt, B.; Girvan, M.; Lu, Z.; Ott, E. Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Phys. Rev. Lett.* **2018**, *120*, 024102. [CrossRef]

23. Vlachas, P.R.; Pathak, J.; Hunt, B.R.; Sapsis, T.P.; Girvan, M.; Ott, E.; Koumoutsakos, P. Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics. *Neural Netw.* **2020**, *126*, 191–217. [CrossRef]
24. Patel, D.; Canaday, D.; Girvan, M.; Pomerance, A.; Ott, E. Using machine learning to predict statistical properties of non-stationary dynamical processes: System climate, regime transitions, and the effect of stochasticity. *Chaos Interdiscip. J. Nonlinear Sci.* **2021**, *31*, 033149. [CrossRef] [PubMed]
25. Xia, M.; Shao, H.; Ma, X.; de Silva, C.W. A stacked GRU-RNN-based approach for predicting renewable energy and electricity load for smart grid operation. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7050–7059. [CrossRef]
26. Lazcano, A.; Herrera, P.J.; Monge, M. A combined model based on recurrent neural networks and graph convolutional networks for financial time series forecasting. *Mathematics* **2023**, *11*, 224. [CrossRef]
27. Cao, J.; Li, Z.; Li, J. Financial time series forecasting model based on CEEMDAN and LSTM. *Phys. A Stat. Mech. Its Appl.* **2019**, *519*, 127–139. [CrossRef]
28. Fu, K.; Li, H.; Deng, P. Chaotic time series prediction using DTIGNet based on improved temporal-inception and GRU. *Chaos Solitons Fractals* **2022**, *159*, 112183. [CrossRef]
29. Qing, X.; Niu, Y. Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. *Energy* **2018**, *148*, 461–468. [CrossRef]
30. Xia, M.; Shao, H.; Ma, X.; de Silva, C.W. Wind speed forecasting based on the hybrid ensemble empirical mode decomposition and GA-BP neural network method. *Renew. Energy* **2016**, *94*, 629–636.
31. Livieris, I.E.; Pintelas, E.; Pintelas, P. A CNN–LSTM model for gold price time-series forecasting. *Neural Comput. Appl.* **2020**, *32*, 17351–17360. [CrossRef]
32. Fischer, T.; Krauss, C. Deep learning with long short-term memory networks for financial market predictions. *Eur. J. Oper. Res.* **2018**, *270*, 654–669. [CrossRef]
33. Lv, S.X.; Wang, L. Deep learning combined wind speed forecasting with hybrid time series decomposition and multi-objective parameter optimization. *Appl. Energy* **2022**, *311*, 118674. [CrossRef]
34. Zhang, W.; Lin, Z.; Liu, X. Short-term offshore wind power forecasting-A hybrid model based on Discrete Wavelet Transform (DWT), Seasonal Autoregressive Integrated Moving Average (SARIMA), and deep-learning-based Long Short-Term Memory (LSTM). *Renew. Energy* **2022**, *185*, 611–628. [CrossRef]
35. Cheng, Y.; Xing, W.; Pedrycz, W.; Xian, S.; Liu, W. NFIG-X: Non-linear fuzzy information granule series for long-term traffic flow time series forecasting. *IEEE Trans. Fuzzy Syst.* **2023**, *31*, 3582–3597. [CrossRef]
36. Lv, Y.; Duan, Y.; Kang, W.; Li, Z.; Wang, F.-Y. Traffic flow prediction with big data: A deep learning approach. *IEEE Trans. Intell. Transp. Syst.* **2014**, *16*, 865–873. [CrossRef]
37. Bai, S.; Kolter, J.Z.; Koltun, V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv* **2018**, arXiv:1803.01271.
38. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
39. Lorenz, E.N. Deterministic nonperiodic flow. *J. Atmos. Sci.* **1963**, *20*, 130–141. [CrossRef]
40. Mackey, M.C.; Glass, L. Oscillation and chaos in physiological control systems. *Science* **1977**, *197*, 287–289. [CrossRef] [PubMed]
41. Rössler, O.E. An equation for continuous chaos. *Phys. Lett. A* **1976**, *57*, 397–398. [CrossRef]
42. Sajjad, M.; Khan, Z.A.; Ullah, A.; Hussain, T.; Ullah, W.; Lee, M.Y.; Baik, S.W. A novel CNN-GRU-based hybrid approach for short-term residential load forecasting. *IEEE Access* **2020**, *8*, 143759–143768. [CrossRef]
43. Schuster, M.; Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [CrossRef]
44. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. In Proceedings of the Advances in Neural Information Processing Systems 27 (NIPS 2014), Montreal, QC, Canada, 8–13 December 2014.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.