Journal of Computer Science and Frontier Technologies
ISSN Print: 3104-4204
ISSN Online: 3104-4212
CD ISSN: 3104-4220
https://doi.org/10.63313/JCSFT.9050

Forecasting Chaotic Multivariate Time Series Based on DB-ICDNet

RuiXia Xie. LiLi Wang*
College of Physics and Electronic Engineering, Northwest Normal University, Lanzhou 730070, PR China
Email: wanglili@nwnu.edu.cn

How to cite this paper: Xie, R. X., & Wang, L. L. (2026). Forecasting Chaotic Multivariate Time Series Based on DB-ICDNet. *Journal of Computer Science and Frontier Technologies*, 2(3), 86-100. ISSN Print: 3104-4204; ISSN Online: 3104-4212.
https://doi.org/10.63313/JCSFT.9050
Published: 2026-03-10

Copyright © 2026 by author(s) and Erytis Publishing Limited.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
http://creativecommons.org/licenses/by/4.0/

Abstract

Chaotic time series widely exist in complex scenarios such as nonlinear dynamic systems, financial markets, and meteorological observations. Its strong nonlinearity, sensitivity to initial values, and accumulation of long-term prediction errors make it possible to achieve stable, high-precision multi-step forecasting under limited observation conditions. Prediction is extremely challenging. Traditional linear models are difficult to approach nonlinear dynamics, while conventional deep learning models are prone to phase drift and error amplification during long-step extrapolation. Therefore, this study focuses on the problem of multivariate multi-step forecasting, and proposes a lightweight in-depth forecasting framework DB-ICDNet. In the representation learning stage, the framework extracts long-range dependencies and local perturbations through LSTM and causally inflated TCN, and realizes controlled fusion of information in the near-term window based on the residual gated fusion mechanism. In the prediction stage, it designs a step-size conditional dual-memory cooperative decoder, adaptively retrieves long-term and short-term contexts according to different prediction steps, and gradually generates future trajectories with end observations as anchors through incremental cumulative regression heads, supplemented by consistency constraints to suppress error diffusion. Experiments on Lorenz and Chen theoretical data sets and HS300 real data show that this method has higher accuracy and stability in multi-step prediction, especially in long-step extrapolation, it can effectively maintain the trajectory shape and alleviate error accumulation, which verifies its effectiveness in meeting the challenge of chaotic sequence prediction.

Keywords

Chaotic Time Series; Multi-Step Prediction; Attention Mechanism; Two-Branch Coding; Cooperative Decoding

1. Introduction

Chaos is a ubiquitous phenomenon in nature and human society, and widely appears in the fields of physics, meteorology, biological neural network and financial market.

With the passage of time, the dynamic state of these systems often presents chaotic characteristics, and the generated time series are apparently highly nonlinear, highly sensitive to initial values, and limited in long-term predictability. Due to the complexity of chaotic dynamics and the uncertainty of real acquisition conditions, high-precision prediction of chaotic time series has always been a challenging problem.

Around chaotic time series prediction, many kinds of methods have been proposed. According to the seasonal pattern of tuberculosis incidence in China, Mao proposed the SARIMA model as a predictive tool for surveillance of epidemics [1]. AngelCujia uses intervention analyses to estimate shortages by outliers and SARIMA models to obtain best estimates [2]. Although the traditional time series prediction method has been improved, the preconditions for prediction are still relatively strict, and the common time series in life mostly follow nonlinear behavior. With the rise of data mining processes, there is an increasing interest in using Machine Learning (ML) [3] to analyze time-dependent sequences. Matheus built a short-term rainfall time series prediction model through BN, which is more practical than the long-term rainfall series prediction model [4]，and Luis used the GP time series model to simulate long-term wind speed monitoring data and achieved good results[5]. In the late 1990s, Vapnik V proposed a Support Vector Machine (Support Vector Machine) model with high generalization performance[6]. Kumar proposed a financial market trend prediction model based on Nearest Neighbor Support Vector Machine (Projection-SVM, PSVM), which obtains feature information through quantitative information available in the market, and builds a prediction model. Experimental results verify the effectiveness of the model [7]. Furkan proposed a fuzzy regression function (FRF-SVM) method based on support vector machines. By using FRF-SVM to estimate global daily tie solar radiation, this method has good robustness for outlier observation and overfitting problems [8]. Celestino proposed a prediction method for the remaining service life of aircraft engines that combines ARIMA and SVM models [9]，The ARIMA model is used to estimate the values of predictive variables in advance, and then the results are used as the input of SVM for model construction. Compared with traditional prediction, the results of this method are better. With the in-depth study of neural network related methods, domestic and foreign scholars have also done more in-depth exploration and research on time series modeling and application. Taking the Narmada River in India as an example, Nibedita used the Back Propagation Neural Network (BPNN) training algorithm to predict the daily sand discharge, and performed better in the prediction simulation [10]. RNN is prone to phenomena such as gradient disappearance, gradient explosion or data redundancy [11]. Hochreiter and Schmidhuber proposed the LSTM network based on RNN [12]，which is widely used by people, such as air pollution prediction [13]，clinical data-based diagnosis [14] and solar irradiance prediction [15]. by leveraging memory cells and gating methods to capture sequence

long-term dependencies. Zahra proposed transduction LSTM (T-LSTM), which completes the positioning of the objective function by considering the weighted quadratic cost function, performs weather prediction, and achieves good results [16]. Song et al. propose a model based on long-term short-term memory (LSTM) neural networks to infer production of fractured horizontal wells in volcanic reservoirs, addressing the limitations of traditional methods and showing accurate predictions. The particle swarm optimization algorithm is used to optimize the basic configuration of the LSTM model, and the experimental results show that the proposed LSTM model is superior to other methods [17].

Aiming at the problems of complex variable coupling and serious error accumulation in multi-step prediction of multivariate chaotic time series, this study proposes a lightweight deep learning model DB-ICDNet. The model takes the raw multivariate sequence as input and is modeled end-to-end under strict causal constraints. Representation learning adopts two-branch temporal coding, LSTM captures long-range dependence and continuous evolution, and TCN extracts multi-scale local features through causal inflation convolution. The residual gated fusion mechanism adaptively modulates the two-branch information to enhance cross-variable interaction modeling. In the prediction stage, step-size conditional decoding is introduced to retrieve context cooperatively from global and near-term memory, and the trend and local dual predictors combined with adaptive mixing coefficients are used to dynamically balance long-term evolution and short-term disturbances. The predictor adopts KAN nonlinear mapping to improve the fitting ability. Multi-step trajectories are generated step by step with the end observations as the anchor point, and the error diffusion is suppressed through joint calibration. Auxiliary items such as smoothing constraints, regularization, and consistency constraints are added to the training to improve prediction accuracy and long-term stability while ensuring efficiency.

2. Methods

2.1. Proposed model

Aiming at the problems of complex variable coupling, frequent local disturbances and spikes, and easy accumulation of long-term errors in multi-step prediction tasks of chaotic multivariate time series, traditional schemes based on a single recurrent network or a single convolution network are often difficult to take into account continuous state evolution modeling and local multi-scale dynamic capture at the same time. Therefore, this study proposes a multi-step prediction model DB-ICDNet (Dual-Branch Incremental Co-Decoder Network), which completes end-to-end modeling in the original multivariable observation domain, and improves the accuracy of multi-step prediction through the collaborative design of structural constraints and training objectives and stability. The overall structure of the model is shown in Figure 1.

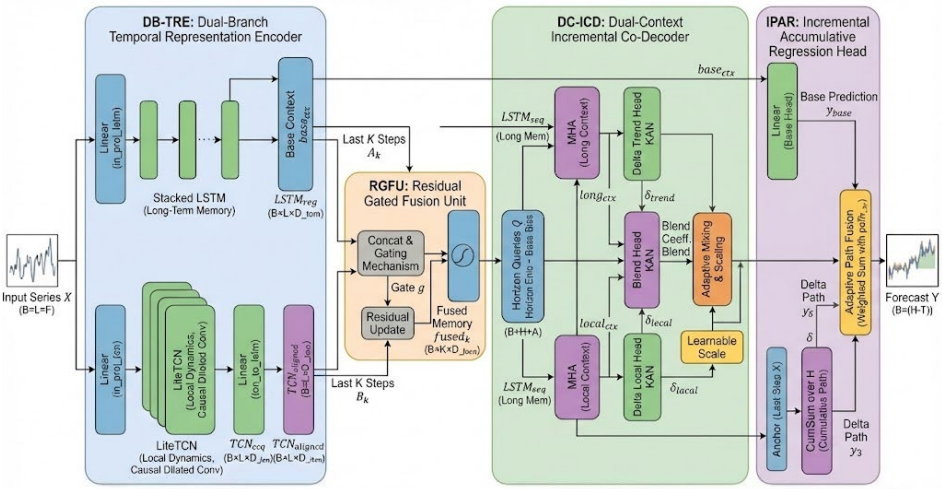


Figure 1. Overall architecture of DB-ICDNet for multivariate chaotic multi-step forecasting

DB-ICDNet consists of input correction and feature projection, two-branch multi-scale timing encoder, progressive adaptive gating fusion, step-size conditional stepwise decoder, KAN nonlinear projection head and enhanced loss constraint in training phase. The overall process is as follows: the original multivariable sequence is first mapped to the unified hidden space by reversible instance normalization correction and linear projection, and then enters the continuous recursive encoder and causal multi-scale dilated convolution encoder respectively to extract complementary dynamic representations; The output of the two branches is fused adaptively through the dimension-by-dimension gating mechanism to obtain a unified global context representation; On the prediction side, the model adopts step-size conditional step-by-step decoding, introduces step-size embedding into different prediction steps to achieve differentiated mapping, and uses KAN as the main nonlinear output head to generate future multi-step prediction results; In order to restrain the long-term error diffusion, the model introduces a progressive refining correction branch in the long-term prediction interval to carry out structural compensation for long-span prediction; In the training stage, the enhancement loss for multi-step time series prediction is introduced, and the dynamic consistency and long-term weighted constraints are superimposed on the basis of point value fitting to improve the stability of the prediction trajectory and the accuracy of long-step prediction.

2.2. Two-branch multiscale timing encoder

The encoding stage of DB-ICDNet takes "two-branch parallel modeling" as the main line, and the goal is to obtain complementary representations sensitive to long-term evolution and short-term disturbances at the same time without greatly increasing computational overhead. Chaotic multivariate sequences often exhibit multi-scale coupling characteristics, including smooth evolution dominated by dynamic systems,

as well as spikes and irregular fluctuations caused by noise, emergencies, or local phase changes. Recursive encoding of a single path is generally better at accumulating long-term dependencies, but the response to local high-frequency perturbations may be smoothed; Convolutional coding can extract local patterns and form multi-scale receptive fields in parallel, but if it is used directly on the scale of the whole sequence, it is easy to spread high-frequency fluctuations into long-term memory and affect stability. Based on the above analysis, this section constructs a two-branch multi-scale timing encoder (DB-TRE), which characterizes long-range dependence and state continuous evolution through LSTM full sequence branches, extracts local multi-scale perturbations through causally inflated TCN branches, and completes Dimensional alignment provides a unified representation space for subsequent dimension-by-dimension gated fusion and dual-context decoding.

2.3. Residual gated fusion

There is a difference between the long-term and local multi-scale representation of the output of the two-branch encoder in terms of information focus: the LSTM branch emphasizes continuous evolution and long-range dependence, and the TCN branch is more sensitive to near-term disturbances and spike changes. If the fusion is directly carried out by splicing or fixed weight addition, two types of problems are prone to occur. On the one hand, the fusion weight lacks adaptability to the heterogeneity of samples and variables, resulting in local disturbances being over-amplified or over-suppressed in some scenarios; On the other hand, splicing will bring dimensional expansion and increase subsequent module parameters, which is not conducive to maintaining the overall lightweight design. Based on the above considerations, this section designs a residual gated fusion module (DIRG), which adaptively modulates the two-branch representation in a dimension-by-dimension gated manner within the near-term window, and adopts the differential injected residual form to achieve stable fusion, so that local multi-scale information is supplemented into the long-term representation in a controlled manner. A schematic of the module is shown in Figure 2.

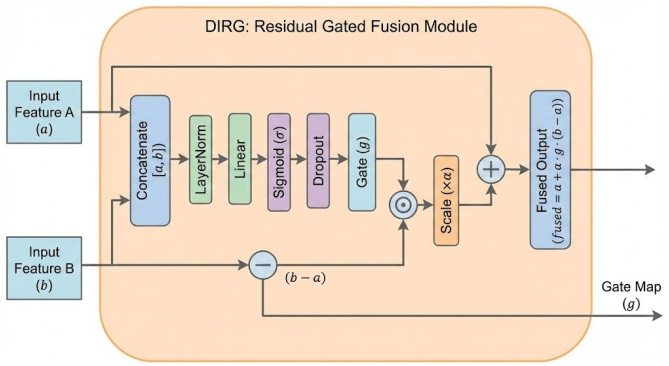


Figure 2. Architecture of the Differential Injection Residual Gate (DIRG)

2.4. Dual-memory step collaborative decoder

The core difficulty of multi-step forecasting is that different forecasting steps rely inconsistently on historical information: near-end forecasting tends to rely more on recent disturbances and local phase information, while far-end forecasting relies more on long-term evolution laws and global trend structure. If a uniform context representation or a fixed fusion strategy is used at the decoding end, the model is prone to error diffusion in the long-term range, or the response to local catastrophe is insufficient in the near-end range. Based on this observation, this section proposes a dual-memory step-size conditional cooperative decoder, which simultaneously maintains two types of context sources: full-sequence long-term memory and near-term fusion memory in the decoding stage, and provides different prediction steps through step-size embedding. Differentiation conditions enable the model to perform step-size adaptive collaborative modeling between trend information and local disturbance information.

2.5. Incremental cumulative regression head

The dual-memory step-size conditional cooperative decoder outputs two types of context-driven step-size representations, and on this basis generates incremental predictions for future H-steps. In chaotic multi-step forecasting, if the target value of each forecasting step is directly regressed independently, the model will easily produce inconsistent jumps in the step size dimension, and the long-term error will be more likely to spread. In order to maintain the time consistency of the multi-step output and improve the approximation ability to complex nonlinear dynamics, this section constructs an incremental cumulative regression head (IPAR). The regression head is composed of an "incremental generation path" and a "global calibration path", where KAN is used for the incremental generation path and for finer nonlinear fitting of the incremental map; The global calibration path uses lightweight linear mapping to provide overall calibration capability for multi-step outputs.

3. Dataset and experimental setup

3.1. Experimental data

In order to ensure the reliability and reproducibility of the experimental conclusions, this study sets theoretical chaos and financial market data sets to carry out multi-variable multi-step prediction experiments. The first class is the theoretical chaotic dataset, and two typical continuous chaotic systems, Chen and Lorenz, are selected.

The state equation of Chen chaotic system is defined as: $a = 35, b = 3, c = 28$,

$$\begin{aligned} \dot{x} &= a(y - x), \\ \dot{y} &= (c - a)x - xz + cy \\ \dot{z} &= xy - bz. \end{aligned} \tag{1}$$

Set the initial value (x_0, y_0, z_0), The fourth-order Runge-Kutta iterative output is performed at a fixed step size Δt and the data schematics are shown in Figure 3 and 4 below.

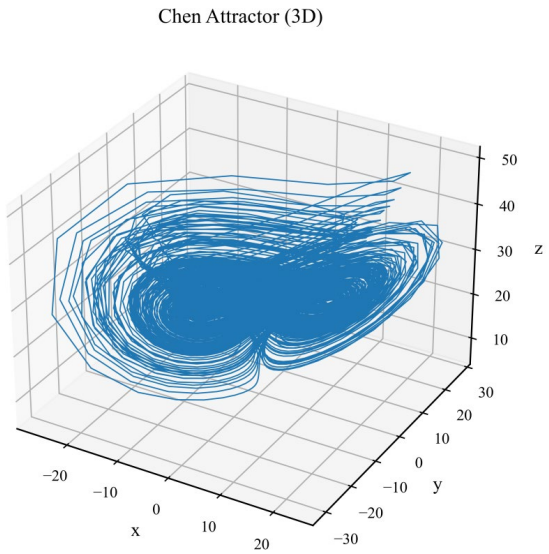


Figure 3. Three-dimensional diagram of Chen system

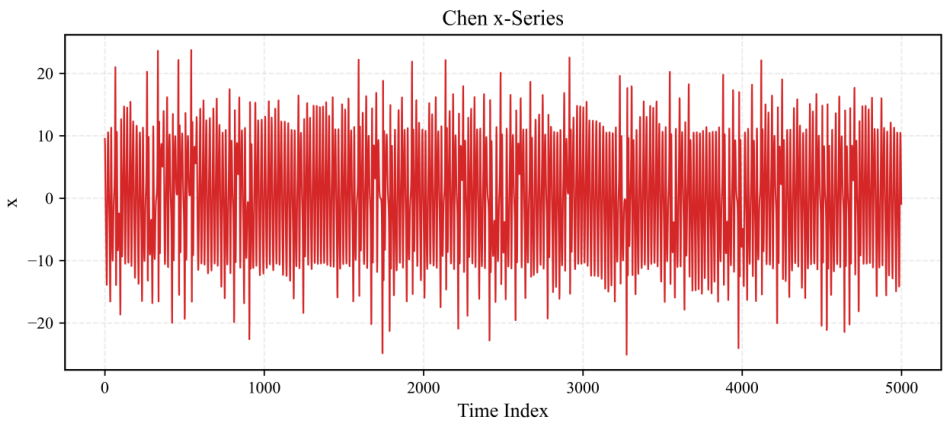


Figure 4. Chen system x diagram

The Lorenz-63 system takes the classical parameters $\sigma = 10, \rho = 28, \beta = 8/3$, and its equation of state is,

$$\begin{aligned} \dot{x} &= \sigma(y - x), \\ \dot{y} &= x(\rho - z) - y \\ \dot{z} &= xy - \beta z. \end{aligned} \tag{2}$$

When constructing the data, set the initial value (x_0, y_0, z_0) and the integration termination time T, construct an equidistant sampling sequence $t_k = k\Delta t$ with step size Δt, and use an ordinary differential equation numerical solver to integrate over

[0,T] to obtain the trajectory. Finally, sample at t_k to obtain N state points.

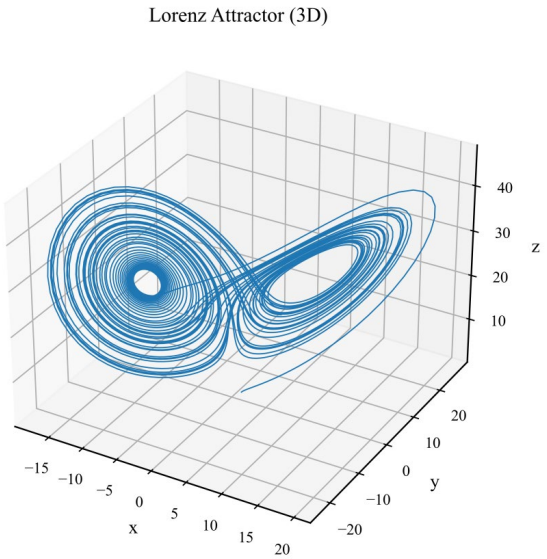


Figure 5. Three-dimensional diagram of Lorenz system

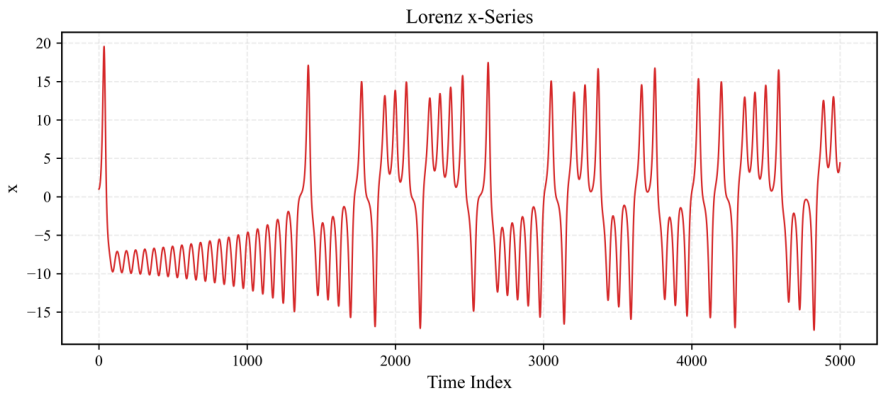


Figure 6. Lorenz system x graph

The second category is the financial market data set, which selects the daily closing price series of the Shanghai and Shenzhen 300 Index as the research object. The data source is the historical market download of Netease Finance and Economics, of which the CSI 300 covers July 2005 to June 2021, and the Shanghai Composite Index and Shenzhen Component Index cover October 2005 to April 2022. This study uses daily frequency OHLCV information As a multi-variable input, that is, the opening price (open), the highest price (high), the lowest price (low), the closing price (close) and the trading volume (volume), to describe the joint impact of price range fluctuations and trading activity on future trends, the forecast target is set as the closing price sequence (close) of multiple steps in the future.

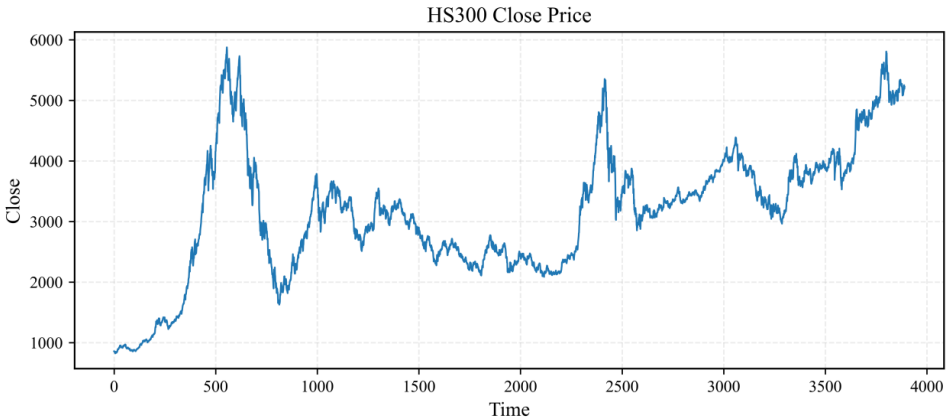


Figure 7. Closing price trend chart of CSI 300 Index

3.2. Experimental parameter setting

In order to verify the training stability and prediction performance of the proposed model on different types of chaotic sequences, comparative experiments are carried out on theoretical chaotic system data and real chaotic sequence data. For Chen and Lorenz chaotic systems, a sliding time window is used to construct the supervised samples. The length of the history window is 96, and the prediction step is 48. Data were divided chronologically into training versus test sets at a ratio of 8: 2. For real chaotic sequence data, the input window length is adjusted to 30, and the prediction step size is adjusted to 7. The rest of the data construction process remains consistent.

During the training process, samples are inputted into the network in batches and processed for forward calculation, and the model output and real label are calculated and lost to perform back propagation to update the parameters. The optimizer uses AdamW, the initial learning rate is set to 0.001, the batch size is 32, and the total number of training rounds is 150. The ReduceLROnPlateau learning rate scheduling policy is enabled during the training phase. To ensure that the results are reproducible, the random seed is set to 42, and the random states of fixed Python, NumPy, and PyTorch are synchronized.

3.3. Prediction evaluation index

To measure the accuracy and stability of model prediction, this study uses four indicators to evaluate whether the predicted value generated by the prediction method fits the real value[18]. Given by the following formula:

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{3}$$

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{4}$$

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}} \tag{5}$$

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

Where y_i is the true value, $\hat{y}_i$is the predicted value, n is the length of the predicted series, and $\bar{y}$represents the average of the true values of the series. The value range of R^2 is usually between [0,1], and the closer the value is to 1, the better the fitting effect of the model.

4. Experimental results and analysis

4.1. Lorenz

This section conducts a result error analysis for the Lorenz theoretical chaotic dataset. First, the comparison of the error indicators of different models is given in Table 1, and then the cumulative characteristics of the error with the increase of the step size are discussed in combination with the visualization curves of the three prediction steps of t 1, t 25 and t 48 in Figure 8.

Table 1. Comparison of multi-step prediction errors in the Lorenz dataset

Model	MAE	MSE	RMSE	R^2
LSTM	0.1623	0.0719	0.2454	0.9989
GRU	0.2596	0.1633	0.3695	0.9974
LSTM-CNN	0.6343	0.9509	0.8811	0.9848
LSTM-Attention	0.1375	0.0826	0.239	0.9987
TCN	1.6248	5.4802	2.0837	0.9125
DB-ICDNet	0.0276	0.0018	0.0398	0.9999

As can be seen from Table 1, the error differences of each baseline model on the Lorenz dataset are more obvious. LSTM-Attention performed relatively better in the traditional RNN series, with MAE and RMSE of 0.1375 and 0.2390 respectively, indicating that there are still obvious cumulative deviations in multi-step extrapolation. The error of TCN is large, and the MAE reaches 1.6248, which shows that TCN is not enough to describe the long-range evolution of Lorenz system. In contrast, this research method achieves the minimum value on all error indicators, with MAE 0.0276 and RMSE 0.0398, showing stronger multi-step prediction accuracy and stability.

Combined with the visualization results in Figure 8, most models can follow the overall trend under the condition of short step size t 1, but as the prediction step size increases, the baseline method gradually shows phase drift and extreme amplitude distortion, especially at the peak-valley turning point. deviation; At t + 25 and t + 48, this kind of deviation will accumulate further and appear as the dislocation or overshoot of the local fluctuation segment. This research method can maintain a good degree of curve coincidence under the three step sizes, and can still stably align the peak value, valley value and key turning position at the long step size t 48, reflecting the nonlinear dynamic structure of the Lorenz system. Stronger characterization ability also explains the reason for the error reduction in Table 1.

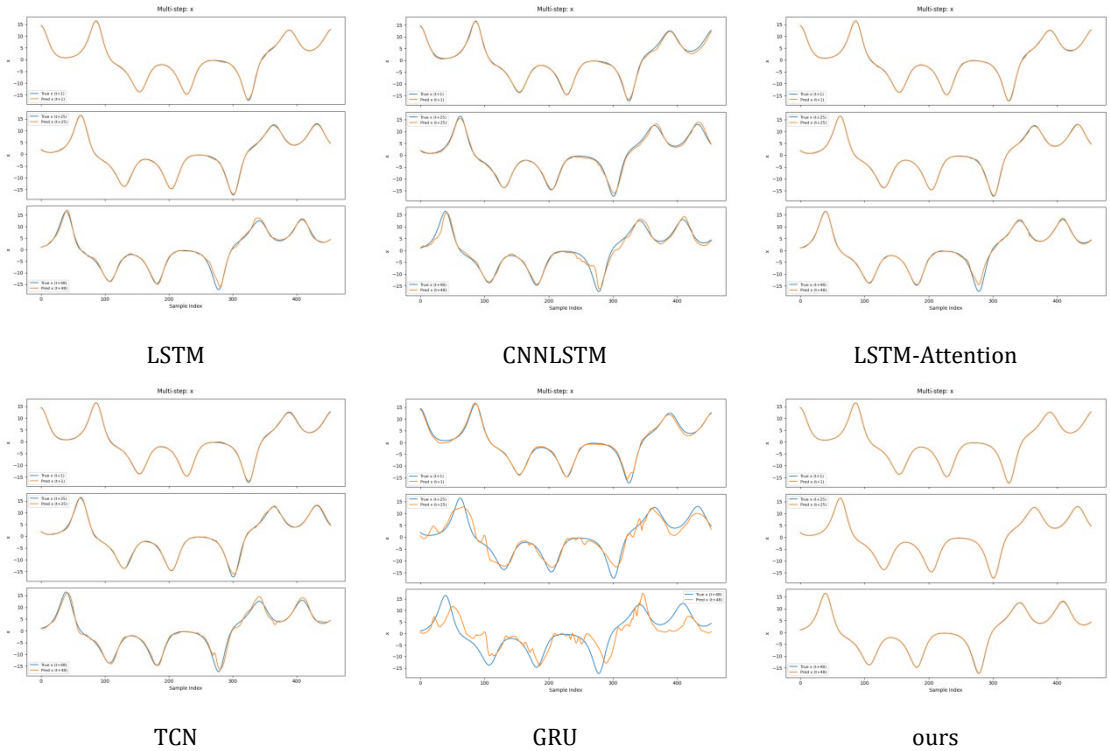


Figure 8. Comparison of Lorenz dataset results

4.2. Chen

This section further analyzes the error of the multi-step prediction results of the chaotic dataset of Chen theory. The quantitative error comparison of each model is given in Table 2, and then combined with the visualization curves of Chen's data set (corresponding to the three prediction steps of t 1, t 25, and t 48 respectively), the error accumulation of different methods when the step size increases way.

Table 2. Comparison of multi-step prediction errors in Chen dataset

Model	MAE	MSE	RMSE	R^2
LSTM	0.3871	0.8527	0.7226	0.988
GRU	0.5029	1.3663	0.9672	0.9808
LSTM-CNN	0.4823	0.8617	0.8031	0.9879
LSTM-Attention	0.2556	1.1327	0.6714	0.984
TCN	2.6997	16.9021	3.677	0.7623
DB-ICDNet	0.1099	0.2559	0.3516	0.9964

It can be seen from Table 2 that the overall error of each baseline model on the Chen dataset is higher than that of Lorenz, indicating that the system presents stronger nonlinear coupling and local instability under the current sampling and prediction settings, resulting in multi-step extrapolation more prone to Cumulative bias. LSTM-Attention is relatively better in the baseline, but its MAE is still 0.2556, and its RMSE is 0.6714, indicating that there are still obvious deviations in the prediction of medium and long steps. The error of TCN is magnified (MAE 2.6997), reflecting its lack of stability in modeling multi-scale oscillations and phase evolution of Chen

systems. In contrast, this research method achieves the minimum value on all error indicators, MAE is 0.1099, RMSE is 0.3516, indicating that the model not only reduces the average deviation, but also more effectively suppresses the error diffusion caused by the increase of step size.

Combined with the visual comparison of Chen's dataset, it can be seen that under the short step size t 1, most models can follow the main oscillation trend, but when the step size is extended to t 25 and t 48, some baseline methods begin to appear peak-valley position dislocation and local amplitude distortion, manifested as advance/lag of the turning segment and overshoot or contraction near the extreme value, so that the waveform of the middle and back segments gradually deviates from the real trajectory. This method maintains better curve coincidence under the three step sizes, especially at t + 48, it can still more stably align the main peaks and valleys and key turning intervals, reflecting its stronger fit to Chen chaotic systems.

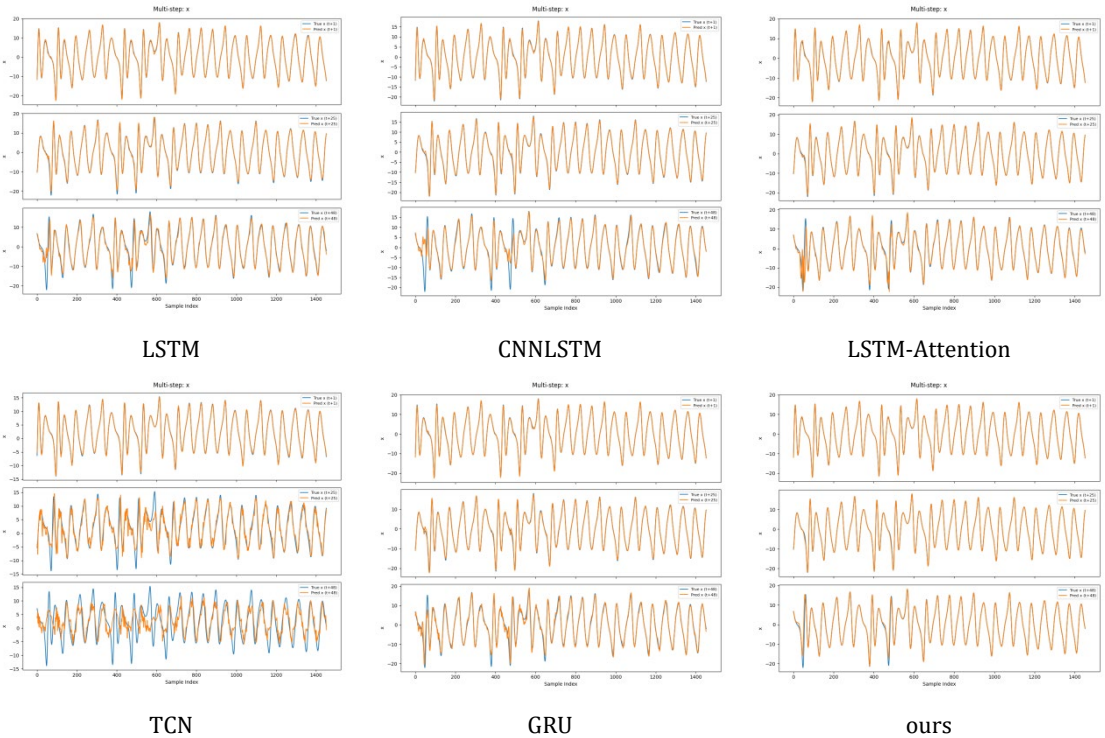


Figure 9. Comparison of chen dataset results

4.3. HS300 share price

Furthermore, the multi-step prediction error analysis on the real financial market data set (HS300 stock price) is given. In order to ensure the objectivity of the comparison, all models are trained and tested under the same data division and evaluation process, and OHLCV (open, high, low, close, volume) is used as the input feature, and the prediction target is the future multi-step closing price sequence. Table 3 summarizes the error indicators of different models on the test set. In

addition, the last sample is extracted from the test set for multi-step visual comparison, and the results are shown in Figure 10.

Table 3. Comparison of multi-step prediction errors in HS300 stock price data set

Model	MAE	MSE	RMSE	R^2
LSTM	102.78	21017.88	138.421	0.9357
GRU	101.1247	19690.90	134.773	0.9398
LSTM-CNN	110.3862	22718.08	148.516	0.9306
LSTM-Attention	104.461	20717.65	139.106	0.9366
TCN	97.4725	18356.89	130.888	0.9439
DB-ICDNet	96.6155	16672.83	129.123	0.9452

It can be seen from Table 3 that on the HS300 dataset, the convolution structure and gating structure are generally better than traditional LSTM and its attention variants, among which TCN has been able to reduce the MAE to 97.4725, indicating that local pattern extraction has a great impact on the short-term fluctuations of financial sequences. There is a certain adaptability. On this basis, this research method further achieves the minimum error, MAE is 96.6155, RMSE is 129.1233, which shows that the model not only reduces the absolute error, but also converges the relative error synchronously, reflecting better multi-step prediction accuracy.

Figure 10 shows the comparison curve of the last sample of the test set at different prediction steps. Where MAE is the evaluation index data of the last sample. It can be observed that the partial baseline method is more likely to shift the trend segment when the step size increases, which is manifested as inconsistent characterization of local inflection points and fluctuation amplitudes, resulting in visible cumulative deviations in the middle and back segments. In contrast, this research method has a higher degree of fit with the real curve in each prediction step, and the follow-up of key turning points and fluctuation amplitudes is more stable. The deviation of the curve is mainly concentrated in a small number of local intervals and the amplitude is small. The synchronous decline of MAE, RMSE and relative error indicators in Table 3 is consistent, indicating that the model has a better ability to describe the "trend-fluctuation" coupling structure in real financial scenarios.

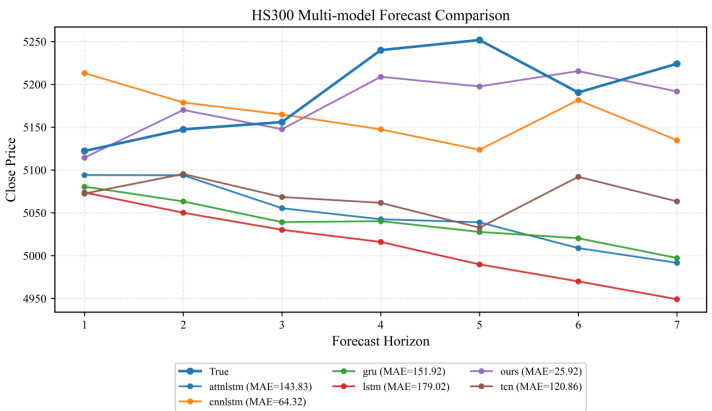


Figure 10. Comparison of HS300 dataset results

5. Conclusion

This research focuses on the multi-step prediction task of chaotic multivariate time series. Aiming at the problems of easy accumulation of errors in long-step extrapolation and insufficient characterization of nonlinear dynamics, a prediction framework centered on two-branch temporal series representation coding and incremental cooperative decoding is constructed., and the corresponding training and reasoning process is given. The experimental part is systematically evaluated on the chaotic dataset of Lorenz and Chen theory and the financial market dataset. Through the comparison of error indicators and the visualization results under different prediction steps, it shows that the proposed method has consistent advantages in multi-step prediction accuracy and stability. It can better maintain the alignment of trajectory shape and key turning points under medium and long step conditions, and provides a reproducible technical basis for subsequent time series modeling and application verification for complex real scenes. The proposed method can maintain more stable phase alignment and trajectory morphology consistency under different prediction steps, and the clear contribution of each key module to accuracy and stability is reflected in comparative experiments and ablation analysis. The paper has formed a complete technical chain from "chaos discrimination-univariate phase space prediction-multivariate multi-step end-to-end modeling".

Although the performance of this research method is consistent with that of the real data, there are still some problems worthy of further investigation in multivariate chaotic multi-step prediction. The long-term extrapolation ability of the model is still limited by the dynamic interval and noise structure covered by the training data. When the data distribution drifts or there are unseen extreme disturbances, the stability of the long-term prediction may still decline. Subsequent research can introduce stricter distributed robust training strategies and uncertainty modeling mechanisms under the premise of maintaining causal constraints, so that the model can provide credible intervals while giving point predictions, and achieve more robust results when the distribution changes. decision.

Acknowledgments

This study was funded by Gansu Province Higher Education Innovation Fund of China (2024A-005).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Mao Q, Zhang K, Yan W, et al. Forecasting the incidence of tuberculosis in China using the seasonal auto-regressive integrated moving average (SA-RIMA) model[J]. Journal of infection and public health, 2018, 11(5): 707-712.

[2] Cujia A, Agudelo D, Pacheco-Bustos C, et al. Forecast of PM10 timeseries data: A study case in Caribbean cities[J]. Atmospheric Pollution Research, 2019, 10(6): 2053-2062.

[3] Ahmed N K, Atiya A F, Gayar N E, et al. An empirical comparison of machine learning models for time series forecasting[J]. Econometric Reviews, 2010, 29(5-6): 594-621.

[4] Geraldi M S, Ghisi E. Short-term instead of long-term rainfall time series in rainwater harvesting simulation in houses: An assessment using Bayesian Network[J]. Resources, Conservation and Recycling, 2019, 144: 1-12.

[5] Avendao-Valencia L D, Chatzi E N. Modelling long-term vibration monitor-ing data with Gaussian Process time-series models[J]. IFAC-PapersOnLine, 2019, 52(28): 26-31.

[6] Cortes C, Vapnik V. Support-vector networks[J]. Machine learning, 1995, 20(3): 273-297.

[7] Kumar D, Meghwani S S, Thakur M. Proximal support vector machine ba-sed hybrid prediction models for trend forecasting in financial markets[J]. Journal of Computational Science, 2016, 17: 1-13.

[8] Baser F, Demirhan H. A fuzzy regression with support vector machine ap-proach to the estimation of horizontal global solar radiation[J].Energy, 2017, 123: 229-240.

[9] Ordóez C, Lasheras F S, Roca-Pardias J, et al. A hybrid ARIMA – SVM model for the study of the remaining useful life of aircraft engines[J]. Jou-rnal of Computational and Applied Mathematics, 2019, 346: 184-191.

[10] Bisoyi N, Gupta H, Padhy N P, et al. Prediction of daily sediment discha-rge using a back propagation neural network training algorithm: A case st-udy of the Narmada River, India[J]. International journal of sediment rese-arch, 2019, 34(2): 125-135.

[11] Sherstinsky A. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network[J]. arXiv preprint arXiv:1808.03314, 2018.

[12] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computat-ion, 1997, 9(8): 1735-1780.

[13] Freeman B S, Taylor G, Gharabaghi B, et al. Forecasting air quality time series using deep learning[J]. Journal of the Air & Waste Management As-sociation, 2018, 68(8): 866-886.

[14] Lipton Z C, Kale D C, Elkan C, et al. Learning to diagnose with LSTM recurrent neural networks[J]. arXiv preprint arXiv:1511.03677, 2015.

[15] Qing X, Niu Y. Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM[J]. Energy, 2018, 148: 461-468.

[16] Karevan Z, Suykens J K. Transductive LSTM for time-series prediction: An application to weather forecasting[J]. Neural Networks, 2020.

[17] Song X, Liu Y, Xue L, et al. Time-series well performance prediction bas-ed on Long Short Term Memory (LSTM) neural network model[J]. Journal of Petroleum Science and Engineering, 2020, 186: 106682.

[18] Tian, Z. D. "Preliminary Research of Chaotic Characteristics and Prediction of Short-Term Wind Speed Time Series," International Journal of Bifurcat-ion and Chaos, 2020, 30,2050176.