Sensors and Materials, Vol. 37, No. 5 (2025) 2013–2026
MYU Tokyo

S & M 4038

Hybrid Convolutional-gated Recurrent Unit Neural Network Model for Prediction of Weather Indicators

Hongtao Zhang,[1] Tsungming Lo,[2*] Wanying Zhang,[3] Ho Sheng Chen,[4]
Jian Xiong,[1] Chihmin Yu,[5] and Tiansyung Lan[6]

[1]Department of Artificial Intelligence and Data Science, Guangzhou Xinhua University,
No. 721, Guangshan 1st Road, Tianhe District, Guangzhou City, Guangdong Province 510520, China
[2]Dongguan University of Technology, Guangdong-Taiwan College of Industrial Science & Technology
251 Xueyuan Road, Guancheng District, Dongguan City, Guangdong Province 523000, China
[3]School of Electrical and Computer Engineering, Nanfang College-Guangzhou,
No. 882, Wenquan Avenue, Conghua District, Guangzhou, Guangdong Province 510970, China
[4]School of Materials Science and Engineering, Guangdong University of Petrochemical Technology,
139 Guandu Road, Maonan District, Maoming, Guangdong Province 525000, China
[5]School of Materials Science and Engineering, Chung Yuan Christian University,
No. 200, Zhongbei Rd., Zhongli Dist., Taoyuan Province 320314, Taiwan (R.O.C.)
[6]Department of Information Management, Yu Da University of Science and Technology,
No. 168, Hsueh-fu Rd., Tanwen Village, Chaochiao Township, Miaoli County 361, Taiwan

(Received November 11, 2024; accepted May 9, 2025)

Keywords: weather prediction, convolutional neural network (CNN), gated recurrent unit neural network (GRU), long short-term memory (LSTM), feedforward neural network (FNN)

In this study, we developed a novel integrated model that combines a convolutional neural network (CNN) with a gated recurrent unit neural network (GRU) for the prediction of regional weather. The CNN–GRU model addresses the inherent complexity influenced by global environmental factors in capturing high-level data characteristics and reduces the error rate of current climate index prediction models. The model conducts time-series prediction gathering high-level data characteristics based on CNN and GRU's capacity. For an accurate prediction, the CNN–GRU model integrates data and retrieved features from it. In the experiment, data including four climatic indices of Beijing, China from 1901 to 2022 were used to construct two-dimensional time-series matrices. The model outperformed the other models including the long short-term memory (LSTM)–fusion neural network, bilateral LSTM, bilateral GRU, double-LSTM, double-GRU, and CNN–GRU (Double-Conv2d) models. The CNN–GRU model was more accurate than the other models.

1. Introduction

Temperature, precipitation, the number of rainy days, and the rate of cloud cover are the weather indicators crucial to constructing and evaluating a climate prediction model. To measure such indicators, a variety of sensors are employed in meteorological stations and satellites. For temperature measurement, a thermistor, platinum resistance thermometer, and thermocouples

*Corresponding author: e-mail: 2024028@dgut.edu.cn
https://doi.org/10.18494/SAM5469

ISSN 0914-4935
https://myukk.org/

are used.[1] Precipitation is generally measured using a tipping bucket rain gauge, a weighing rain gauge, optical precipitation sensors, and weather radars.[2] Radars are used to detect precipitation by measuring the reflection of radio waves off raindrops and other hydrometeors. Doppler radars, in particular, can also detect the motion of raindrops. Rainy days are observed using a weighing rain gauge or automated weather stations integrating rain gauges and other sensors.[3] Ceilometers, pyranometers, and satellite instruments such as a moderate-resolution imaging spectroradiometer and a visible infrared imaging radiometer suite are used to monitor cloud cover from land or space.[4] These sensors and devices are widely used to measure weather data. The selection of sensors depends on the application and the level of precision required in ground-based meteorological stations or remote satellite systems.

The spatiotemporal features of indicators and global environmental changes pose a challenge to predicting climate accurately. Recently, machine learning technology has been applied to predicting climate from previous records leveraging its capability to extract information on weather.[5–9] Machine learning algorithms for climate prediction have considerably advanced. Kadow *et al.* proposed a method based on long short-term memory (LSTM) using a double-layer network to predict temperature, precipitation, humidity, and sunshine duration hours.[10] This method showed smaller root mean squared error (*RMSE*), mean absolute percent error (*MAPE*), and validation loss than models based on recurrent neural networks (RNNs) and the support vector machine (SVM). However, the model cannot capture non-time-series characteristics in the data on weather. To predict PM2.5 concentration and meteorological data, Zhang *et al.* suggested an LSTM–fusion neural network (FNN) model.[11] In this model, the properties of the deep neural network (DNN) were integrated with the linked layer. Compared with the conventional LSTM model, the model reduced *RMSE* and *MAE* by 11.60 and 14.86%, respectively. However, the DNN layers could not capture the high-level properties of the data despite their capability to capture non-time-series characteristics.[11] Kuo *et al.* proposed an LSTM- and FNN-based forecasting model using Kunming City's climatic comfort index. The model combined a single-layer FNN with a four-layer LSTM.[12] On the basis of the climate data from 1980 to 2010, the model forecasted the temperature of the Kunming area and reduced *RMSE* and *MAE* by 0.12 and 0.09 ℃, respectively, which were improved forecasts compared with those obtained using the model constructed with a bidirectional recurrent neural network (BRNN) and a bidirectional long short-term memory neural network (Bi-LSTM). Nevertheless, this model failed to capture the features related to oscillations in the loss function (*RMSE*). Zhao suggested a convolutional neural network (CNN)–gated recurrent unit neural network (GRU)-based model to predict multipoint temperature and humidity for mushroom cultivation. The two-layer GRU and CNN networks were used with a maximum pooling layer.[12] A dataset including air temperature, relative humidity, substrate temperature, and light intensity was used for model training and predicting the spatial distribution of humidity over 20 min. When compared with backpropagation (BP), LSTM, and GRU models, the *RMSE* and *MAE* at different points inside the mushroom greenhouse were reduced by 2.731 and 1.713%, respectively. However, computational complexity resulted from the construction of double convolutional layers with a single pooling layer even though this method captured the high-level properties of the data.[12–14]

Zhang used a Bi-LSTM and an auto-encoder (AE) in a prediction model for the concentration of the particulate matter of 2.5 µm (PM2.5). To use data for adaptive learning, an AE layer was added before the Bi-LSTM. The auto-encoded input was transmitted to the Bi-LSTM for time-series prediction. The model outperformed the RNN, LSTM, AE-RNN, and AE-LSTM models in terms of *RMSE*. However, the model could not predict other weather indicators because of their complicated characteristics.(15) A deep learning model for the prediction of evapotranspiration (ET0) one week in advance was proposed by Ahmed *et al*.(16) For daily ET0 prediction, a model composed of CNN, GRU, and ant colony optimization (ACO) algorithms was used for multistage learning using ET0 data from the previous week. The model presented a lower *MAE* and a higher efficiency than multivariate adaptive regression spline (MARS), LSTM, and GRU models. However, its accuracy in predicting other types of weather was not validated.(16–19)

Considering the benefits and drawbacks of the previous models, we developed a weather prediction model by integrating CNN and GRU in this study. The CNN–GRU model outperformed other models in terms of *RMSE* and *MAE* on the test and training datasets. Its loss function also converged better. The model predicted temperature, precipitation, the number of rainy days, and the rate of cloud cover with high accuracy.

2. Dataset and Models

The climatic research unit time-series (CRU-TS) dataset was used in this study as it is the most widely used. The dataset is provided by the United Kingdom's National Centre for Atmospheric Science (NCAS). It is constructed using monthly observational data and angular distance weighting (ADW) interpolation. It comprises the data measured daily or sub-daily. The network common data form (NetCDF) was also used for dimensionality reduction. The data on the land surface area near Beijing, China at the longitude and latitude of 39.75 and 116.75° were processed. Daily mean temperature (℃), precipitation (mm), the number of rainy days (days), and the rate of cloud cover (%) were chosen to evaluate the models. Figure 1 shows the daily mean temperatures in the Beijing area from 1901 to 2022.

2.1 CNN

CNN is an FNN using convolutional computations. CNN consists of convolutional, pooling, and fully connected layers (Fig. 2). Convolutional layers are responsible for capturing higher-level characteristics of the data and analyzing intrinsic relationships of features. In addition to reducing computational complexity and the number of parameters, pooling layers are used to improve the efficiency of training.(21,22)

2.2 GRU

GRU is an RNN that memorizes key events in sequential data and predicts the following events on the basis of time-series data (Fig. 3).(23) Its architecture is similar to that of LSTM.

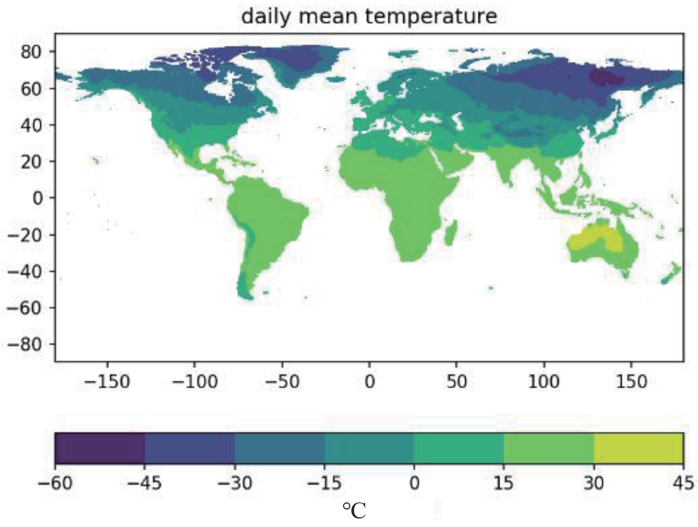

Fig. 1. (Color online) Daily mean temperature in Beijing area from 1901 to 2022.[20]

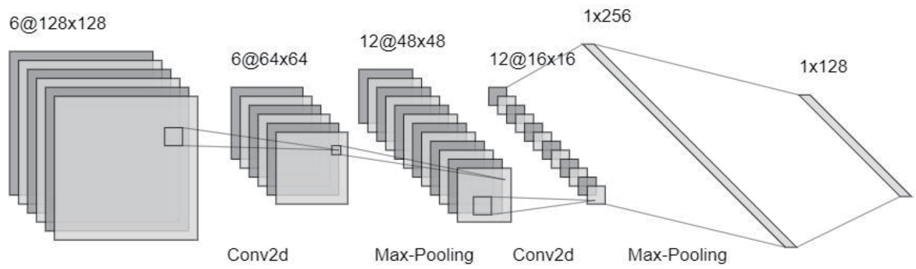

Fig. 2. Structure of CNN.

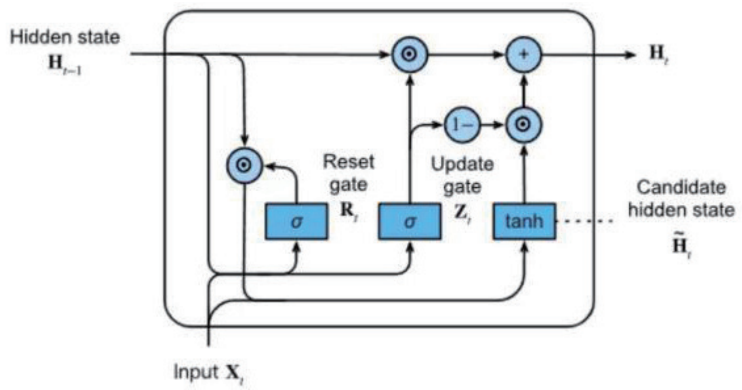

Fig. 3. (Color online) Structure of GRU.

GRU adopts gating units to address the inability of the standard RNN to maintain long-term memory and gradient problems in backpropagation.[24] It has a simpler internal architecture than LSTM, which requires less computational requirements.[25]

When the input for each time step includes the hidden state H_{t-1} determined from the previous time step and the input x_t from the current time step, Z_t is the output of the update gate, which is used to control the degree of update of the current state. When R_t is the output of the reset gate, the impact of past states on the current state is determined. $\widetilde{H_t}$ is a candidate hidden state by the superposition of the current input and past states. H_t updates the hidden state at the current time by updating the weighted average of the gate and past states, as well as the weighted average of the candidate's hidden state $\widetilde{H_t}$ (1–4).

$$Z_t = \sigma\left[W_z \cdot \left(H_{t-1}, x_t\right)\right] \tag{1}$$

$$R_t = \sigma\left[W_R \cdot \left(H_{t-1}, x_t\right)\right] \tag{2}$$

$$\widetilde{H_t} = \tanh\left[W \cdot \left(R_t \times H_{t-1}, x_t\right)\right] \tag{3}$$

$$H_t = \left(1 - Z_t\right) \times H_{t-1} + Z_t \times \widetilde{H_t} \tag{4}$$

3. Model Construction

3.1 Data processing

The original data were processed for reducing dimensionality, converging to the coordinates of the Beijing area, and obtaining the time-series data on temperature, precipitation, the number of rainy days, and the rate of cloud cover. The processed data were normalized to obtain the time-series matrix (Fig. 4).

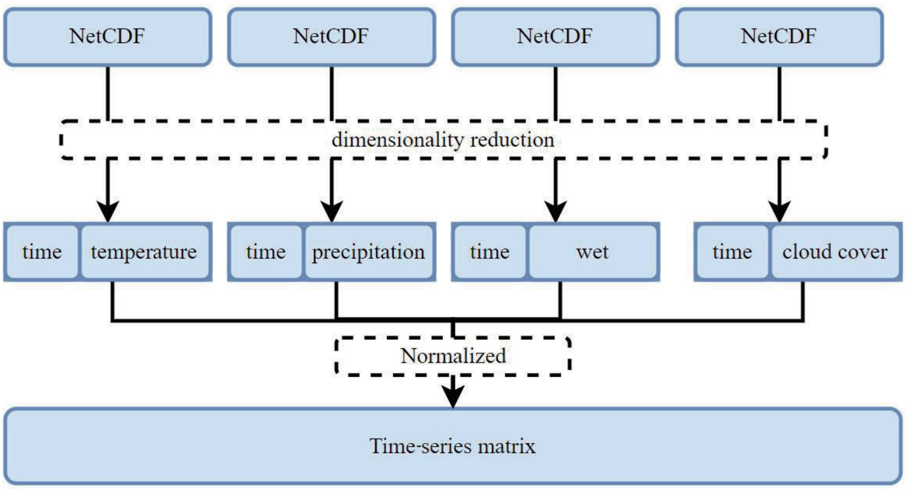

Fig. 4. (Color online) Flow of data processing.

3.2 CNN–GRU model

The effects of environmental factors on weather indicators are diverse and complex. Unexpected events such as natural disasters can alter their distributions with subsequent effects on temperature and precipitation patterns.(26) Human activities also affect the weather indicators. Such factors introduce high-level characteristics into the climate data and affect the accuracy of traditional time-series prediction models as they cannot capture the nuanced and interrelated effects of the factors.(27) To address this challenge, we incorporated a GRU model into a CNN model. This CNN–GRU model leverages the strength of CNN in extracting features and patterns, which is critical to understanding the complex interactions among the indicators.(28) The integration of CNN layers enables the model to detect intricate patterns and inherent high-level characteristics in the data, which are not determined by traditional time-series models.(29)

The extracted features obtained by the CNN–GRU model were input into a two-layer GRU structure as GRU effectively processes and predicts time-series data and captures the temporal dynamics of weather indicators. GRU is also known for its efficiency in modeling temporal dependencies and enables nuanced time-series predictions.(30) To the model, a fully connected layer of FNN was added to integrate the characteristics extracted by the previous layer. To enhance the robustness and generalization capabilities of the model, a dropout regularization technique was applied to prevent overfitting and ensure its effectiveness and accuracy.(31)

Figure 5 shows the structure of the CNN–GRU model, illustrating the integration of the CNN and GRU layers and their respective roles in the model. The model was designed to capture and analyze complex patterns of the data and effectively predict future trends and changes in the spatial and temporal dimensions of the data. In the model, climate dynamics caused by natural and anthropogenic factors were considered to enhance prediction accuracy.

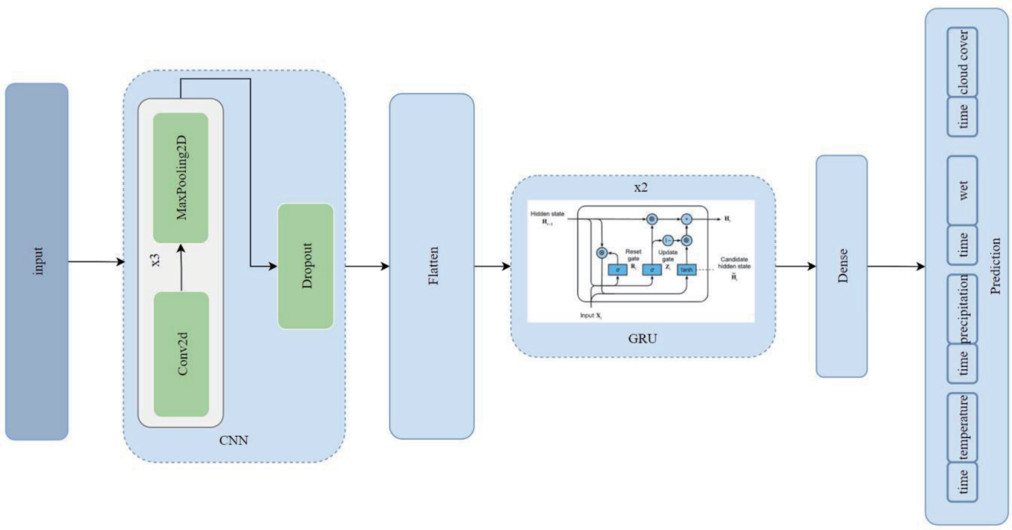

Fig. 5. (Color online) CNN–GRU model structure.

3.2.1 Input layer

The input layer was responsible for the dimension reduction of the data to establish a two-dimensional time-series matrix. The data were normalized to ensure consistency in scale and distribution and input into the model. For data normalization, the Min–max normalization method was used to make the data range from 0 to 1 [Eq. (5)]. This method was used to increase the comparability and stability of the model.[32]

$$Z = \frac{\left(x - min\right)}{\left(max - min\right)}, \tag{5}$$

where x represents the original data, and min and max respectively represent the minimum and maximum values of the original data.

3.2.2 CNN layers

CNN layers were used to extract semantic and spatiotemporal characteristics (high-level characteristics). Three convolutional layers (Conv2D) paired with three pooling layers (MaxPooling2D) were used as the CNN layers. Each of the convolutional layers had 32, 64, and 128 filters with a kernel size of 3 × 3 and a stride of 1. The rectified linear unit (ReLU) function was used as the activation function. The data were processed in a pooling layer with a pool size of 2 to reduce the number of dimensions of high-dimensional characteristics and computational complexity. The dropout regularization was applied to mitigate overfitting caused by weight decay. Lastly, the high-level characteristics captured by the CNN layers were fed into the GRU layer to capture temporal characteristics.

3.2.3 GRU layers

GRU layers were used to recurrently process the high-level characteristics and identify significant events at different times to capture the temporal characteristics. The two-layer GRU network structure was used with 64 and 128 neurons in each layer. The ReLU function was applied as the activation function. After capturing the temporal characteristics, a single-layer feed-forward neural network was used to construct a fully connected output layer.[32]

3.3 Prediction

The CNN–GRU model was trained using the data from 1901 to 2017 to predict weather indicators from 2018 to 2022. The results were compared with the measured data (Fig. 6).

The process of prediction consisted of the following three steps (Fig. 7):

Step 1. Process data to reduce the dimension of data, help the model converge fast, and improve performance.
Step 2. Train model to predict indicators.
Step 3. Predict four weather indicators for the next 60 months.

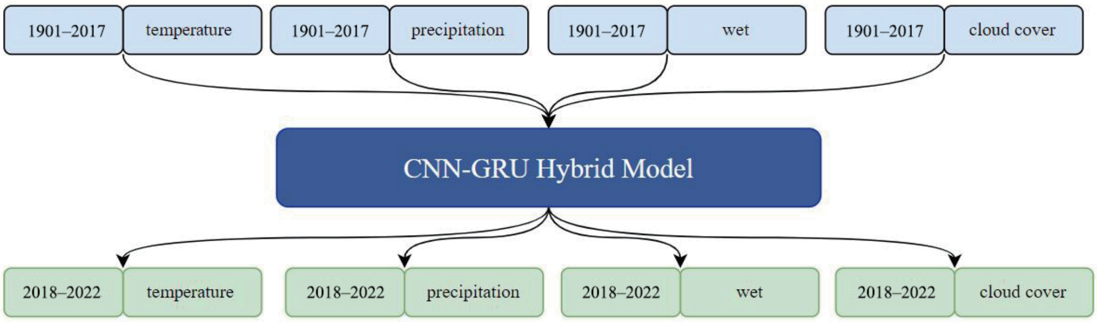

Fig. 6. (Color online) Training CNN–GRU model and its prediction.

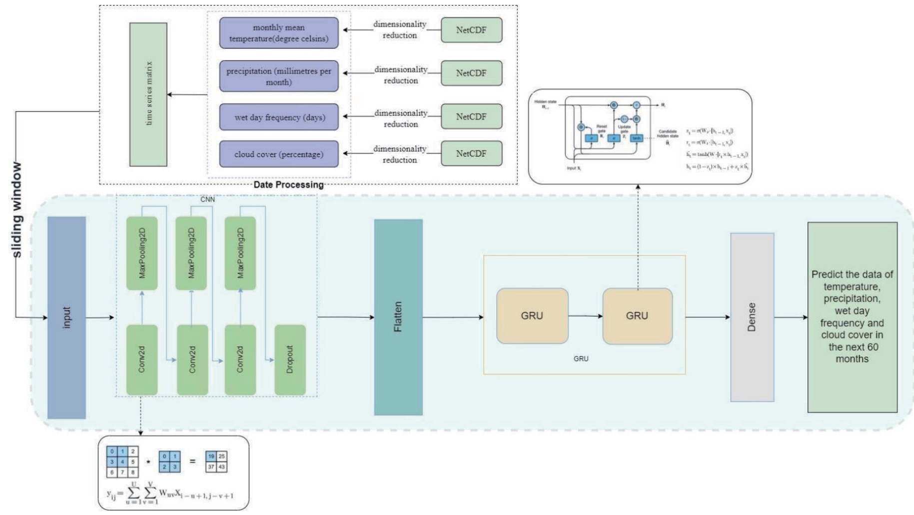

Fig. 7. (Color online) System structure of CNN–GRU model.

4. Methods

4.1 System configuration

The hardware used for the experiment included an AMD Ryzen 7 5800H processor, a graphic card of Radeon 3.20 GHz, and a RAM of 16.0 GB (13.9 GB available) in the Windows 64-bit operating system.

4.2 Evaluation metrics of model performance

We used *RMSE*[33] and *MAE*[34] as evaluation metrics for the model [Eqs. (6–8)]. Specifically, *RMSE* was used to represent the loss function value. In model training, *RMSE* and *MAE* were monitored on the training and test datasets to estimate the convergence rate. The dataset from 1901 to 2017 was split into ratios of 0.75 and 0.25 for training and testing the models, respectively.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{y}_i - y_i\right)^2 \tag{6}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\hat{y}_i - y_i\right)^2} \tag{7}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\hat{y}_i - y_i\right| \tag{8}$$

5. Results and Discussion

5.1 Performance evaluation in training and testing

The convergence rates of *RMSE* and *MAE* of the CNN–GRU model were compared with those of the LSTM–FNN, Bi-LSTM, Bi-GRU, Double-LSTM, Double-GRU, and CNN–GRU (Double-Conv2d) models. All models had the same number of epochs, input batch size, ReLU activation function, and Adam optimizer. The CNN–GRU model showed a higher convergence rate of *RMSE* with a more gradual decrease. Its *RMSE* and *MAE* were lower than those of the other methods. Figures 8 and 9 show the *RMSE* and *MAE* of the CNN–GRU model for the weather indicators over 100 epochs in training and testing.

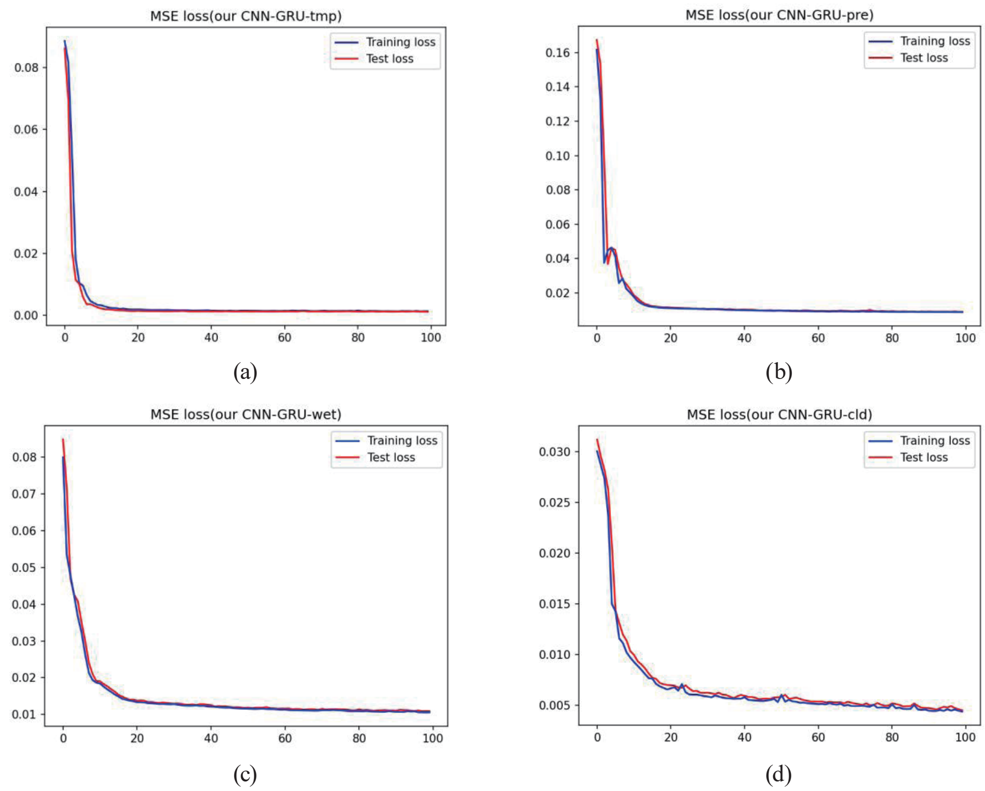

Fig. 8. (Color online) Loss functions of weather indicators. (a) Loss function of temperature. (b) Loss function of precipitation. (c) Loss function of number of rainy days. (d) Loss function of rate of cloud cover.

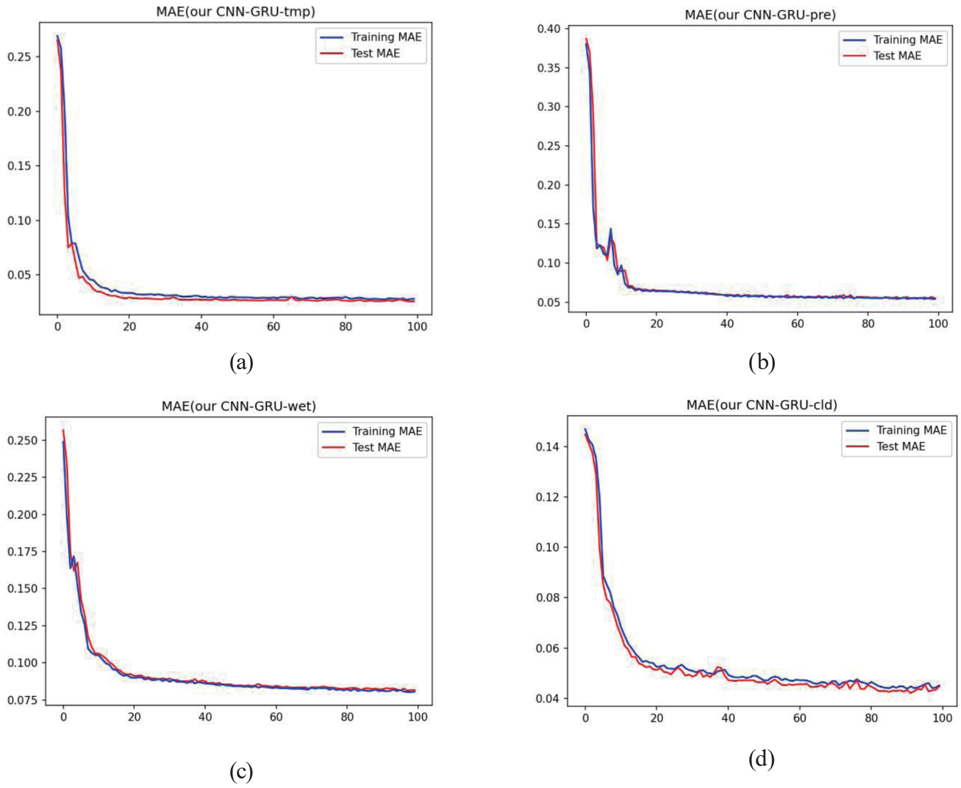

Fig. 9. (Color online) *MAE*s of weather indicators. (a) *MAE* of temperature. (b) *MAE* of precipitation. (c) *MAE* of number of rainy days. (d) *MAE* of rate of cloud cover.

The CNN–GRU model showed faster convergence and lower *MAE* for the weather indicators than the other models (Table 1). The *RMSE* and *MAE* values of the weather indicators were stabilized after 20 epochs and approached minimum by the 40th iteration. These findings proved the exceptional convergence rate of the CNN–GRU model. The developed CNN–GRU model showed *RMSE* values of 0.0331 ℃, 0.0938 mm, 0.1024 days, and 0.0663%, whereas the other models showed *RMSE* values of 0.0346–0.1442 ℃, 0.0932–0.136 mm, 0.0167–0.133 days, and 0.0768–0.0948% for temperature, precipitation, the number of rainy days, and the rate of cloud cover, respectively. The *MAE* values of the CNN–GRU model were 0.0256 ℃, 0.0540 mm, 0.0806 days, and 0.0425%, whereas those of the other models were 0.0268–0.0909 ℃, 0.0542–0.1092 mm, 0.0829–0.1085 days, and 0.0506–0.0684% for temperature, precipitation, the number of rainy days, and the rate of cloud cover, respectively. The *RMSE* and *MAE* values of the CNN–GRU model were lower than those of the other models, which indicates the better prediction results of the CNN–GRU model.

5.2 Performance evaluation in predicting

The data from 2018 to 2022 were fed into the model for prediction. A sliding window approach was employed to predict the data for 60 months with a window size of 12 months (Fig. 10).

Table 1
Loss function and *MAE* values of different models.

Model	Weather indicators	Loss function	*MAE*
LSTM–FNN	Temperature (°C)	0.1442	0.0909
	Precipitation (mm)	0.1360	0.1092
	Number of rainy days (days)	0.1330	0.1085
	Rate of cloud cover (%)	0.0948	0.0684
Double-LSTM	Temperature (°C)	0.0400	0.0314
	Precipitation (mm)	0.1009	0.0597
	Number of rainy days (days)	0.1131	0.0885
	Rate of cloud cover (%)	0.0774	0.0517
Double-GRU	Temperature (°C)	0.0374	0.0292
	Precipitation (mm)	0.0994	0.0582
	Number of rainy days (days)	0.1086	0.0847
	Rate of cloud cover (%)	0.0768	0.0518
Bi-LSTM	Temperature (°C)	0.0412	0.0320
	Precipitation (mm)	0.1019	0.0682
	Number of rainy days (days)	0.1135	0.0881
	Rate of cloud cover (%)	0.0787	0.0530
Bi-GRU	Temperature (°C)	0.0400	0.0319
	Precipitation (mm)	0.0984	0.0609
	Number of rainy days (days)	0.1118	0.0868
	Rate of cloud cover (%)	0.0774	0.0508
CNN–GRU (Double-Conv2d)	Temperature (°C)	0.0346	0.0268
	Precipitation (mm)	0.0932	0.0542
	Number of rainy days (days)	0.1067	0.0829
	Rate of cloud cover (%)	0.0774	0.0506
CNN–GRU (Developed in this study)	Temperature (°C)	0.0331	0.0256
	Precipitation (mm)	0.0938	0.0540
	Number of rainy days (days)	0.1024	0.0806
	Rate of cloud cover (%)	0.0663	0.0425

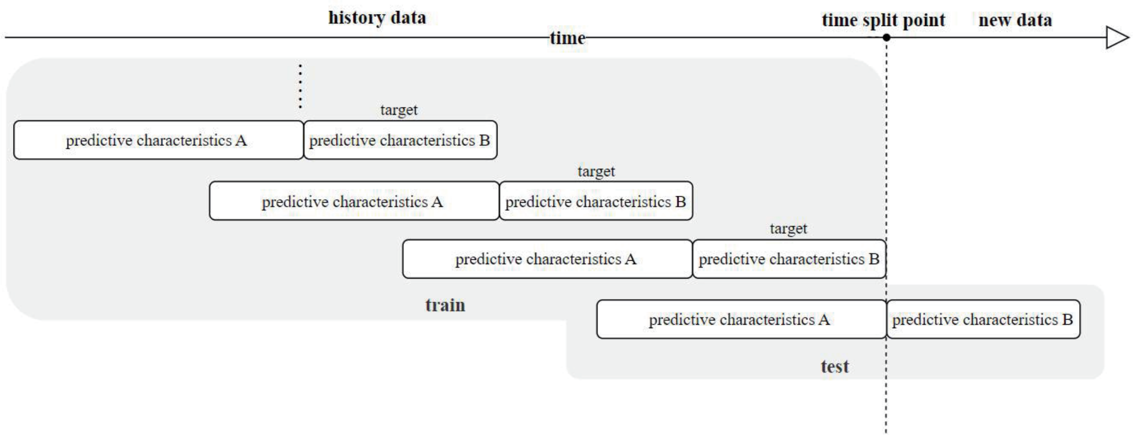

Fig. 10. Data windowing process.

After predicting the weather indicators from 2018 to 2022, the results were compared with the measured data. Figure 11 shows the predicted and measured data of the weather indicators

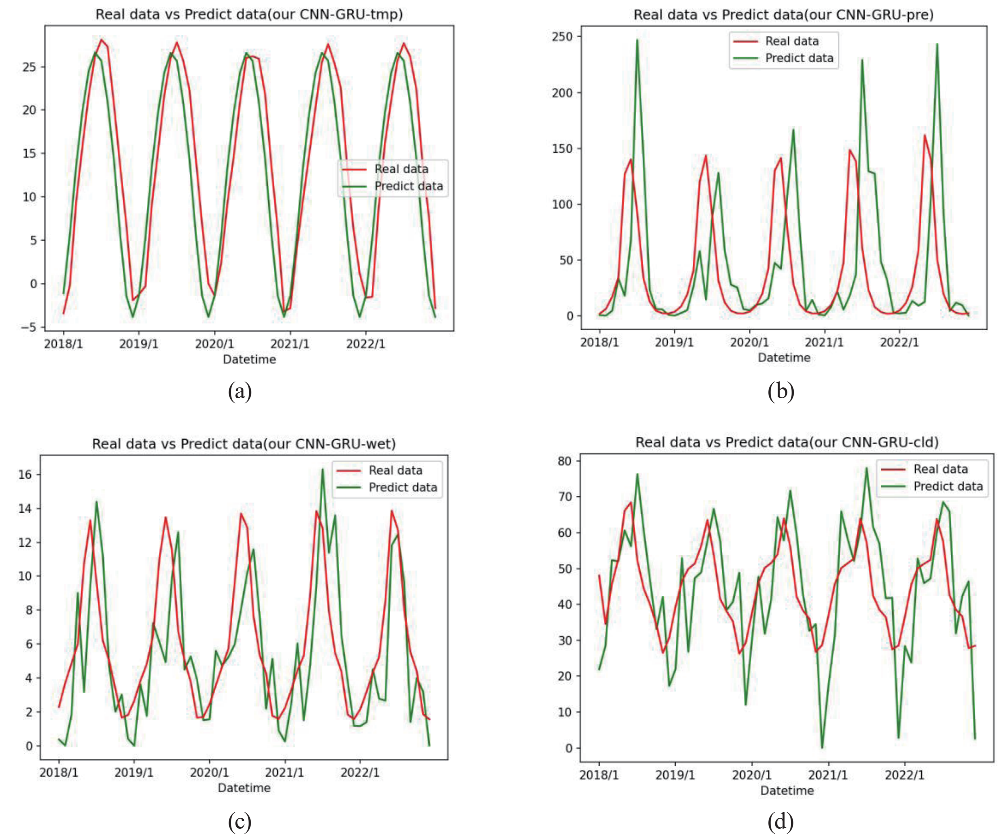

Fig. 11. (Color online) Predicted and measured data of weather indicators from 2018 to 2022. (a) Temperature, (b) precipitation, (c) number of rainy days, and (d) rate of cloud cover.

predicted by the CNN–GRU model. The CNN–GRU model accurately predicted temperature, precipitation, the number of rainy days, and the rate of cloud cover. The model also captured the temporal characteristics of the four indicators, reflecting the variability of the weather indicators for the next 60 months. The predicted temperature data showed the smallest difference, indicating the model's accuracy in predicting temperature.

5. Conclusions

We developed a CNN–GRU model to predict weather indicators and compared its performance in training, testing, and prediction with other models including LSTM–FNN, Bi-LSTM, Bi-GRU, Double-LSTM, Double-GRU, and CNN–GRU (Double-Conv2d) models. To evaluate its performance, its *RMSE*, convergence rate, and *MAE* were compared with those of the other models. The CNN–GRU model showed lower loss function and *MAE* than the other methods, indicating that the model presented a higher accuracy. The CNN–GRU model predicted the weather indicators accurately with negligible differences from those of the measured data from 2018 to 2022. Such results proved the superior prediction capability of the CNN–GRU model to the other models. The developed CNN–GRU model in this study can be used for predicting certain weather indicators in the long term for up to five years, but it needs an improvement to be used for the prediction of other indicators. In addition, the model prediction results can be considered to develop relevant weather sensors to provide more accurate data.

Acknowledgments

This research was funded mainly by "Guangdong Province Key Discipline Construction Scientific Research Capacity", grant number 2022ZDJS152, titled "Research on the Application of Artificial Intelligence Based on Medical Imaging Big Data", and partly by Xinhua University-Guangzhou, grant number 2024KYZDZK04, titled "Research on Image Encryption Algorithms Based on Coupled Piecewise Chaotic Mapping of Linear Diophantine Equations".

References

1 PennState College of Earth and Mineral Scinece: https://www.e-education.psu.edu/meteo3/l3_p10.html (accessed Octiber 2024).
2 M. Putra, M. S. Rosid, and D. Handoko: Signals **5** (2024) 542. https://doi.org/10.3390/signals5030030
3 W. Wan and Y. Zhou: Atmos. Res. **310** (2024) 107600. https://doi.org/10.1016/j.atmosres.2024.107600
4 X. Dong, B. Xi, and P. Minnis: J. Clim. **19** (2006) 1765. https://doi.org/10.1175/JCLI3710.1
5 B. Zhang, Z. Guo, L. Zhang, T. Zhou, and T. Hayasaka: Earth Space Sci. **7** (2020) e2019EA001027. https://doi.org/10.1029/2019EA001027
6 P. B. Gibson, W. E. Chapman, A. Altinok, L. Delle Monache, M. J. DeFlorio, and D. E. Waliser: Commun. Earth Environ. **2** (2021). https://doi.org/10.1038/s43247-021-00225-4
7 S. He, H. Wang, H. Li, and J. Zhao: J. Atmos. Intell **4** (2021) 13. https://doi.org/10.32629/jai.v4i1.483
8 Z. Zhou, W. Tang, M. Li, W. Cao, and Z. Yuan: Remote Sens. **15** (2023) 1951. https://doi.org/10.3390/rs15071951
9 S. Liu, D. Lu, S. L. Painter, N. A. Griffiths, and E. M. Pierce: FRWA **5** (2023). https://www.frontiersin.org/journals/water/articles/10.3389/frwa.2023.1150126/full
10 C. Kadow, D. D. Hall, and U. Ulbrich: Nat. Geosci. **13** (2020) 408. https://www.nature.com/articles/s41561-020-0582-5
11 L. Zhang, X. Zhao, G. Zhu, J. He, J. Chen. Z. Chen, S. Traore, J. Liu, and V. P. Singh: Agric. Water. Manag. **289** (2023) 108498. https://doi.org/10.1016/j.agwat.2023.108498
12 P.-C. Kuo, Y.-T. Chou, K.-Y. Li, W.-T. Chang, Y.-N. Huang, and C.-S. Chen: Eng. Struct. **306** (2024) 117733. https://doi.org/10.1016/j.engstruct.2024.117733
13 S. Huang, Q. Liu, Y. Wu, M. Chen, and H. Yin: Agronomy **14** (2024) 473. https://doi.org/10.3390/agronomy14030473
14 X. Wang, I. Wu, C. Liu, and H. Yang: J. Beijing Univ. Aeronaut. Astronaut. **44** (2018). https://doi.org/10.13700/j.bh.1001-5965.2017.0285
15 B. Zhang, H. Zhang, G. Zhao, and J. Lian: Environ. Model. Softw. **124** (2020) 104600. https://doi.org/10.1016/j.envsoft.2019.104600
16 A. M. Ahmed, R. Deo, Q. Feng, and A. Gharamani: Stoch. Env. Res. Risk. Assess. **36** (2021) 1. https://link.springer.com/article/10.1007%2Fs00477-021-02078-x
17 Z. Zhao, H. Wang, C. Wang, W. Li, H. Chen, and C. Deng: Agric. Water. Manag. **231** (2020). 105958. https://doi.org/10.1016/j.agwat.2019.105958
18 Z. Yin, Q. Feng, L. Yang, R. C. Deo, X. Wen, J. Si, and S. Xiao: Water **9** (2017) 880. https://doi.org/10.3390/w9110880
19 M. Zou, S. Kang, J. Niu, and H. Lu: J. Hydrol. **585** (2020) 124323. https://doi.org/10.1016/j.jhydrol.2019.124323
20 S. I. Peng, J. Guo, Y. Zhao, J. Wang, L. Cao, M. Wang, Q. Qang, and J. Feng: Acta Meteorol. Sin. **80** (2022) 136. http://doi.org/10.11676/qxxb2022.008
21 A. Krizhevsky, I. Sutskever, and G. E. Hinton: Commun. ACM **25** (2012). https://doi.org/10.1145/3065386
22 L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M.

Al-Amidie, and L. Farhan: J. Big Data **8** (2021). https://doi.org/10.1186/s40537-021-00444-8
23 A. Faraji, S. A. Sadrossadat, W. Na, F. Feng, and Q.-J. Zhang: IEEE Trans. Circuits Syst. I: Regul. Pap. **70** (2023) 2904. https://doi.org/10.1109/TCSI.2023.3264616
24 N. Jia, X. Tian, W. Gao, and L. Jiao: Remote Sens. **15** (2023) 3171. https://doi.org/10.3390/rs15123172
25 T. T. Mueller, J. C. Paetzold, C. Prabhakar, D. Usynin, D. Rueckert, and G. Kaissis: IEEE Trans. Pattern Anal. Mach. Intell. **45** (2022) 7308.
26 M. Shiraiwa: ACS Environ. Au. **3** (2023) 121. https://doi.org/10.1021/acsenvironau.3c00014
27 A. J. Felton, R. K. Shriver, M. Stemkovski, J. B. Bradford, K. N. Suding, and P. B. Adler: Ecol. Lett. **25** (2022) 2688. https://doi.org/10.1111/ele.14132
28 P. Das, S. Sharma, and H.K. Kapoor: IEEE Trans. Comput. **72** (2022) 693. https://doi.org/10.1109/TC.2022.3174591
29 T. Xia, B. Zhao, J. Ma, G, Fu, W. Zhao, and N. Zheng IEEE Trans. Circuits Syst. I: Regul. Pap. **70** (2023) 1242. https://doi.org/10.1109/TCSI.2022.3227608
30 A. J. Gámez, C. S. Zhou, A. Timmermann, and J. Kurths: Nonlinear Proc. Geoph. **11** (2004) 393. https://doi.org/10.5194/npg-11-393-2004
31 D. Singh and B. Singh: Appl. Soft Comput. **97** (2020) 105524. https://doi.org/10.1016/j.asoc.2019.105524
32 P. Rajendra, K. V. N. Murthy, A. Subbarao, and R. Boadh: Model. Earth Syst. Environ. **5** (2019) 1051. https://doi.org/10.1007/s40808-019-00590-2
33 C. J. Willmott and K. Matsuura: Clim. Res. **30** (2005) 79. https://doi.org/10.3354/cr030079
34 D. M. Allen: Technometrics **13** (1971) 469. https://doi.org/10.2307/1267161