Stochastic Weather Simulation Based on Gate Recurrent Unit and Generative Adversarial Networks

Lingling Han  
China Agricultural University

Xueqian Fu (fuxueqian@cau.edu.cn)  
China Agricultural University

Chunyu Zhang  
China Agricultural University

Olusola BAMISILE  
Imperial College London

Shirong Shang  
China Agricultural University

Research Article

Keywords:

Posted Date: November 29th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3663582/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License

Additional Declarations: No competing interests reported.
Stochastic Weather Simulation Based on Gate Recurrent Unit and Generative Adversarial Networks

Lingling Han\textsuperscript{a}, Xueqian Fu\textsuperscript{a,b,*}, Chunyu Zhang\textsuperscript{a,b}, Olusola BAMISILE\textsuperscript{c}, Shirong Shang\textsuperscript{d}

\textsuperscript{a}College of Information and Electrical Engineering, China Agricultural University, Beijing, China
\textsuperscript{b}National Innovation Center for Digital Fishery, China Agricultural University, Beijing, China
\textsuperscript{c}Faculty of Natural Sciences, Imperial College London
\textsuperscript{d}College of Science, China Agricultural University, Beijing, China

* Corresponding author. Tel.: +86-10-18813103877
Email address: fuxueqian@cau.edu.cn.

Postal address: CIEE 633, 17 Qinghua Donglu, Beijing 100083, People’s Republic of China

Abstract

The weather has a significant impact on power load and power system planning. Stochastic weather simulation is important in the field of power systems. However, due to factors such as long recording years, observation technology, and so on, the historical weather data often have the problem of missing or insufficient. Meteorological data are characterized by changeable, rapid change, and high dimensions. Therefore, it is a challenging task to accurately grasp the law of weather data. This paper presents a random weather simulation model based on gate recurrent unit (GRU) and generative adversarial networks (GAN). GRU selectively learns or forgets what was in the previous moment during training, it can learn the previous and current data of the time series data. When combined with the GAN, it will produce data with the same distribution as the original weather data. We evaluate the proposed method on a real weather dataset, and the results show that the proposed method outperforms the other contrast algorithms.

Nomenclature

\[ C^t = \text{Memory unit of the GRU} \]
\[ h^t = \text{The hidden state of the GRU} \]
\[ y^t = \text{The output state of the GRU} \]
\[ \theta_d = \text{Networks parameters for discriminator D} \]
\[ \theta_g = \text{Networks parameters for generator G} \]
Climate change has a great effect on power systems [1]. Electricity sources include hydropower, solar power, and other forms, but dry weather can lead to water shortage, which can lead to a decline in hydropower capacity, and changes in sunlight intensity can also affect the efficiency of solar power. Photovoltaic and wind power to new energy development rapidly, especially in rural areas, the grid is more sensitive to the weather [2][3][4]. Therefore, weather conditions have an important impact on the production and supply of electricity. California, USA, has been experiencing power supply shortages due to hot summer weather for many years. If the power sector can predict the high temperature and dry weather in advance, each department can prepare in advance for extreme weather. For example, large and medium-sized reservoirs can store or divert water in advance, to avoid the impact of extreme weather on power supply stability. It can be seen that electricity production and weather changes are inextricably linked. The development of distributed power sources such as wind power and photovoltaics has exacerbated the impact of weather on energy system optimization [5]. Therefore, in order to cope with the adverse effects caused by extreme weather in the future, the power system should fully consider extreme weather conditions, extract and predict weather changes, and carry out source-grid-load-storage collaborative planning to ensure sufficient capacity. Special consideration should also be given to the availability of flexible adjustment capabilities under operating conditions under extreme weather conditions to ensure that source-grid-load-storage integration can still provide feasible operation solutions in extreme conditions to avoid power cuts. In the capacity adequacy planning process, in order to predict weather changes, we need a large amount of historical data for modeling.

However, due to the long recording period, observation technology, and other factors, historical weather data is often missing or insufficient. At the same time, meteorological data has the characteristics of changeable, rapid change, and high dimension, and accurately grasping the laws of weather data is a challenging topic. Therefore, building an accurate and reliable random weather simulation model in electrical systems is particularly important.

Traditional numerical weather scenario generation techniques can be divided into fixed-date method, shifted-date method, and bootstrap method [6]. The first two methods are based on historical contours, but the second method's contours can be shifted. The bootstrap method involves resampling the data. A majority of the aforementioned methods are founded on probabilistic methods, which have limited model capacity [7]. Traditional methods mostly use the Markov model to determine whether the rainfall will occur [8], and then select a distribution to simulate the historical data, such as, gamma distribution [9] and the bivariate normal distribution.
The random weather simulation models (random weather generators) used in meteorological research are mostly statistical models to produce random numbers similar to the original data. In short, most traditional methods use Markov chain and a distribution to simulate weather data, which have a strong dependence on historical data, and the obtained results can not perfectly learn all the valuable features.

In recent years, deep learning has received a lot of attention. Different from the traditional methods, data-based deep learning has a powerful self-directed learning ability, which can effectively simulate high-dimensional, complex weather data. In the temporal data simulation, Long Short-term Memory Networks (LSTM) works well and is widely used. Considering the huge amount of data modeling that does not obey the normal distribution, scientists proposed the method of Generalized Linear Models (GLMS), and introduced Bayesian hierarchical framework, which provides an ensemble of weather series sampled from posterior distributions. The birth of generative adversarial networks (GAN) has enabled a new approach to data simulation, and the simulated weather has better results. GAN has a wide range of applications in image generation, face recognition, and computer vision. The GAN network performed well in the unsupervised image dataset, at the same time can also be applied to the image transformation problem. However, in terms of meteorological data generation, GAN is still less used. In terms of studies of cyclic units, a new model GRU is proposed, which can achieve the effect of LSTM with fewer parameters. Moreover, when the binding of GRU to CNN, it shows very good results.

In summary, in the field of weather generators, most of them are statistical models, using mean and variance as the starting point to model weather variables separately. This does not take into account the correlation between weather variables, only the characteristics of the original data except the mean and variance are considered. In order to learn the relationship between weather variables and learn the most complete features of the original data, we use the GAN. However, since classical GAN does not take into account the temporal characteristics of data, we introduce GRU into GAN. Therefore, a model combining GRU and GAN is proposed to simulate random weather data to model the intensity and temperature of sunshine. GRU is a kind of time series network, which can selectively learn current and past data, combined with the powerful data simulation capabilities of GAN, the proposed method has advantages in meteorological data simulation. The stochastic weather data simulation model constructed in this paper can be applied to power systems to assist in the collaborative planning of source-grid-load-storage to ensure adequate power capacity. In the case of sufficient capacity, the power system has a more flexible operation capacity, thus improving the reliability and stability of the power system.

The main contributions of this paper are as follows: (1) The law of annual weather is analyzed, and a random simulation method of annual light intensity and temperature based on GAN is proposed and named GRU-GAN. GRU is innovatively used as a generator and discriminator for GAN and applied to the stochastic simulation of weather data. GRU learns the data of the current moment and the previous moment of weather data and plays the function of generating data and discrimination in GAN. (2) An evaluation standard system for stochastic simulation of annual light intensity and
temperature is proposed. This paper evaluates the diversity, authenticity, and relevance of the simulated weather. These three aspects can more comprehensively evaluate the quality of the simulated weather relative to the original data.

The rest of this article is organized below. Section 2 details the framework of the GRU-GAN model. Section 3 introduces the data generated by the model in this paper under the actual dataset and the effect of the simulated weather under the evaluation index. Finally, section 4 summarizes the work of this paper.

2 | Methodology

The key to simulating meteorological data is to capture the temporal and variable characteristics of meteorological data. Different from other data, meteorological data are closely related before and after. GRU was initially applied to the field of natural language processing to extract the relationship between words before and afterwords and has a memory function, which is very suitable for extracting the timing of meteorological data. GAN is mostly used for image generation. It is a network with large parameters and strong autonomous learning ability. Compared with traditional methods, GAN can learn more meteorological data features, which is more conducive to learning variable features.

After the generator and discriminator of GAN play against each other, we can finally get the simulated weather with the same distribution as the original data. In the simulation of weather data, we need to simulate the distribution of raw data, and GAN can just meet this requirement. At the same time, GRU is a circular network with a memory mechanism, weather data is a kind of time series data, and the data of the next moment is intrinsically related to the data of the previous moment, and the network of memory mechanisms can learn this intrinsic connection. Therefore, this paper has the advantage of using GRU-GAN for random weather data simulation.

Section 2.1 introduces the architecture of the model proposed in this paper. Section 2.2 introduces the simulated weather evaluation system proposed in this paper, and the generated weather data is evaluated in three aspects: diversity, authenticity, and relevance.

2.1 | Model architecture

Generative Adversarial Network (GAN) includes generator (G) networks and discriminator (D) networks [22]. We borrowed the principle of GAN network and use the game between generator and discriminator to simulate real meteorological data. The function of generator G is to generate false samples, and the function of discriminator D is to distinguish the true and false input samples. Generator and discriminator are two networks that play against each other, learn together, and progress together.

The model architecture is shown in Figure 1.
In the model presented in this paper, the objective functions of discriminator $D$ and generator $G$ are:

$$V_d = \frac{1}{m} \sum_{i=1}^{m} \log D(x^i) + \frac{1}{m} \sum_{i=1}^{m} \log(1 - D(x'^i))$$

(1)

$$V_g = \frac{1}{m} \sum_{i=1}^{m} \log D(G(z'^i))$$

(2)

where, $D(x^i)$ is the degree value of the real data recognized as true by the discriminator, $D(x'^i)$ is the degree value of the simulated weather recognized as true by the discriminator. From the perspective of discriminator $D$, we want $D(x^i)$ as close as possible to 1, $D(x'^i)$ as close as possible to 0; However, From the perspective of the generator, we wish the resulting sample $G(z'^i)$ can be rated high. Thus, the effect of mutual game is achieved, as shown in the Figure 2.

We update the parameters of the network by using the results of the discriminant.

$$\theta_d \leftarrow \theta_d + \eta \nabla V_d(\theta_d)$$

(3)

$$\theta_g \leftarrow \theta_g + \eta \nabla V_g(\theta_g)$$

(4)

where, $\eta$ is the learning rate, we set a learning rate of 0.001. Adjusting the parameter $\theta_g$ is in adjusting the distribution of simulated weather data. $\theta_g$ is the parameter that
the model needs to learn. By updating the parameters, the model also improves its
ability to extract the features of the input meteorological data.

Figure 2. The overall architecture of a classic

In this paper, we optimize the above network to replace the GRU network with
memory function with the original full connection layer, which can capture the timing
characteristics more effectively in the simulated weather data. In our model, the
schematic representation of the discriminator and the generator is shown in Figure 3.

The two layers of GRU are connected to each other, and $h^t$ at time $t$ serves as input at
moment $t + 1$.

Figure 3. Structure diagram of a two-layer GRU network

The parameters of the input memory cell of the Generator and the Discriminator
are only $Z$ and $R$, which control the update gate and reset gate respectively [23]. In
this experiment, $w^i, w', w', w$ orthogonal initialization is used. $b^i, b', b'$ are initialized
with 0.

$$Z = \sigma(w^i \times [x^t; h^{t-1}] + b^i)$$ (5)
\[ R = \sigma(w' \times [x'; h^{i-1}] + b') \] (6)

In the model of this article, we stitch a 24-point sunshine intensity data with 24-point temperature data into one data, for a total of 1271 pieces of data. \( X \in \mathbb{R}^{w \times c} \). Enter weather data for the model, \( w \) is the total number of days in a month in 41 years, \( c \) is the number of hours.

First, the reset gate "resets" the input data and gets \( h' \). The function \( h' \) is to record the state at the current moment [24].

\[ h' = \tanh(w' \times [x'; h^{(t-1)}] + b') \] (7)

Next, update the hidden status \( h' \). The output \( h' \) of the hidden layer contains the information of all meteorological data before time \( t \), which can well preserve the characteristics of meteorological data timing.

\[ h' = (1-Z) \times h^{i-1} + Z \times h' \] (8)

The first half selectively forgets the original hidden state, and the second half selectively remembers the \( h' \) of the current node. Only one gate is used to remember and forget at the same time, reducing the number of parameters, simplifying the calculation process. In this experiment, the first half selectively forgets the meteorological data of the previous moment, and the second half selectively remembers the meteorological data of the current moment.

The output \( y' \) is weather data with the same distribution as the input weather data. Output \( y' \) is:

\[ y' = \sigma(w \times h') \] (9)

In a word, the generator network consists of three layers of primordial GRU and a fully connected layer, and the activation function of each layer of the GRU is tanh and sigmoid. The initial input to the generator network is a randomly generated noise vector \( \{z^1, z^2, \ldots, z^m\} \) from the Gaussian distribution, then enter into the GRU, after updating and resetting, the output produced by the three-layer memory unit is finally obtained. The output results are passed through a fully connected layer to obtain a simulated weather sample that conforms to the original data distribution, after denormalization, the final result is obtained.

The discriminator \( D \) also consists of three layers of primitive GRU and a fully connected layer, the activation function of each layer of the GRU is tanh and sigmoid. The initial input to the discriminator network is raw data. Since the original data contained sunlight intensity and temperature, normalization was carried out to unify the dimension. After input into the GRU, gradient descent and Adam optimizers are used to update the model parameters to better distinguish between generated and raw data.
Then repeat the above steps until the two networks converge, that is, the final simulated weather is obtained.

### 2.2 | Model parameters and evaluation metrics

#### 2.2.1 | Model parameters

This paper takes gradient descent and Adam optimizer to train the model. The GRU-GAN model has a total of 5 hyperparameters. The time step is set to 31, corresponding to 31 days of data per month, representing 31 within the GRU memory units; The batch size is 8, which represents the number of samples captured by each training; The dimension of the hidden state is 4; Training iterations are 35000; Input size is 48, including 24 o'clock of insolation intensity data and 24 o'clock of temperature data.

As mentioned above, in the model construction, we chose a network connected by a three-layer GRU. To ensure that sufficient timing is learned, without overfitting caused by too many parameters, we chose a three-layer network based on experience. It should be noted that the number of layers will have an impact on the effect of the model and the final result: too many layers, the parameters increase, resulting in a waste of computing power, and more importantly, too many layers may lead to the problem of overfitting, so that the model works well on the training set, but the extensiveness is very poor, and the effect on the test data will not be ideal; If the number of layers is too small, the model is not complex enough, and the learning of the temporal features of the meteorological data will be poor, which will cause the model to not learn all the features of the meteorological data, resulting in a poor model effect. In summary, the number of layers of GRU will affect the effect of the model, and based on experience, we chose a three-layer network, which proved to work well in subsequent experiments.

#### 2.2.2 | Evaluation metrics

To verify the quality of the generated weather data, we selected three evaluation metrics for evaluation. (1) Principal Component Analysis (PCA), this evaluation metric reflects the diversity of the data generated. (2) Wasserstein distance, this metric reflects the distance between the simulated weather and the distribution of the raw data, verifying the authenticity of the simulated weather. (3) Mutual information is to verify the internal correlation of the simulated weather data. Since the raw data is spliced by the intensity of sunlight and temperature, there is a certain relationship within the data, and this indicator can reflect the internal relationship between whether the simulated weather extracts the original data.

**1) Principal component analysis**

The principal component analysis is a dimensionality reduction algorithm, it performs linear dimensionality reduction while preserving the maximum variance between samples [25]. The dimensionality reduction data can be represented graphically, and the distribution of simulated weather and raw data can be seen more intuitively. This reflects the diversity of the simulated weather. In this paper, the
sunshine intensity data and weather data are separated, and the distribution map after dimensionality reduction of the simulated weather and the original data is drawn, and the comparison between the two reflects the diversity of the simulated weather relative to the original data.

(2) Wasserstein distance

Wasserstein distance can characterize the degree of similarity between two distributions [26]. Think of both distributions as mounds, and the price of adjusting one mound to the other is the Wasserstein distance. Specifically, for two continuous distributions, the Wasserstein distance is calculated as:

\[ W(P, Q) = \inf_{\gamma \in \Pi(P, Q)} \mathbb{E}_{(x, y) \sim \gamma} [|| x - y ||] \]

\[ E_{(x, y)}[|| x - y ||] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \gamma(x, y) || x - y || dx dy \]

where, \( \Pi(P, Q) \) is the set of all the joint distributions between \( P \) and \( Q \), in the evaluation system of this paper, \( P \) is the generated weather data and \( Q \) is the original weather data, \( \gamma \) is one of these joint distributions. Wasserstein distance is the least expensive to find one \( \gamma \) so that two distributions become identical.

(3) Mutual information

The mutual information indicates the strength of the relationship between two variables \( X \) and \( Y \), \( X \) is the sunshine intensity and \( Y \) is the temperature. Here we measure the relationship between the two variables of sunshine intensity and temperature. In real life, the temperature is affected by the intensity of sunlight, and there should be a correlation between the two. To evaluate whether this relationship is preserved in the simulated weather, we introduce mutual information to evaluate. The mutual information formula is:

\[ I(X, Y) = \int_X \int_Y P(X, Y) \log \frac{P(X, Y)}{P(X)P(Y)} \]

As you can see, if \( X \) is independent of \( Y \), then \( I(X, Y) = 0 \) means that there is no correlation between \( X \) and \( Y \).

3 | EXPERIMENTS

3.1 | Datasets

The dataset chosen in this paper is MERRA-2 (The Second Modern-Era Retrospective analysis for Research and Applications version 2), MERRA-2 is a long-time series analysis dataset that includes various meteorological variables, such as sunshine intensity, temperature, relative humidity, etc. At the same time, MERRA-2 data covers the globe with a temporal resolution of 1 hour. The MERRA-2 dataset
emphasizes the use of satellite observation data, and the data are all on the same resolution scale, its spatial resolution is fixed at 0.625 degrees × 0.5 degrees (Note: longitude direction 5/8 degree, dimension direction 1/2 degree), and MERRA-2 uses aerosol information assimilation model to better simulate meteorological data changes. It uses real-time atmospheric temperature values obtained by the Microwave Limb Sensor (MLS) on the MODIS Aura satellite, and the improved calibration coefficients provided by the MLS data have achieved better results in atmospheric correction in the polar regions.

This paper selects data from Qingyuan City, Guangdong Province, covering the period from 1980 to 2020. In this paper, the data is divided into 12 groups in months, each set of data has 1271 pieces of data, and each data has 48 columns. 1271 records are all weather data for 41 years of the current month, and 48 columns include sunshine intensity and temperature data at 24 o'clock.

To prove the feasibility of the model, we selected a set of data from the above dataset in January 2021-2023 as test data to test the effect of the model.

For ease of processing, the number of days in each month is stretched to 31 days, and the specific operation is as follows: the data of the 30th day is the data of the 31st day, the data of the 29th day is the data of the 30th day, and so on. The data entering the model is preprocessed, so there are 1271 pieces of data every month.

3.2 | Experimental setup

This paper uses the weather data of Qingyuan City, Guangdong Province as input data, and uses 41 years of sunshine intensity and temperature data from 1980 to 2020. The output data is the same size as the input data, and the training output is 1271 pieces of 48 columns of weather data every month. The TensorFlow framework was used to build a random weather simulation model of GRU-GAN, and the training time of each GRU-GAN model on the CPU was 40 minutes.

In order to prove that the weather data generated in this paper can more sensitively capture the unusual changes of real weather data, and the superiority of the memory of this model in generating time series data, we compare the proposed method with the other two methods and use the above dataset to experiment and compare the effect of the simulated weather.

Firstly, we compare the methods of this paper with those of statistical learning. The statistical learning method uses the mean and variance as the starting point for the simulated weather with a similar distribution to the original data [1], and models temperature and sunshine intensity separately. Doing so may overlook small probability events where the data suddenly changes in a real-world situation, and separate modeling ignores the relationship between insolation intensity and temperature. Through this set of comparative experiments, the superiority of the proposed method in small probability event learning can be reflected, and the intrinsic relationship between sunlight intensity and temperature can be retained.

Secondly, we compare the proposed method with that of classical GAN. This set of experiments can show the superiority of the proposed method in generating time series data due to the memory mechanism brought by the introduction of GRU.
3.3 | Experimental results

3.3.1 | Simulated weather

In this paper, a total of 12 GRU-GAN models are trained, the data is put into the model in monthly units, one model is trained every month, and finally, the data generated by the 12 models are stitched together to generate one year's data. Each month has 41 years of data, and 41 years of data are also generated. In this paper, one year's original data is randomly selected to compare the proposed method. Figure 4 shows the sunshine intensity data, take the result of year (2010) as an example where each time interval includes 1 hour. The upper side is the real sunshine intensity data, and the lower side is the generated sunshine intensity data. Figure 5 shows the temperature data, also take the result of year (2010) as an example where each time interval includes 1 hour. The upper side is the real temperature data, and the lower side is the generated temperature data. Through the comparison of the figure, we can see that the distribution of the proposed method fits the distribution of the real data well.

As listed in Figure 4, the sunshine intensity of the original data in June changes more sharply, because June is in summer, the weather is changeable, the influence of dark clouds and other factors makes the sunshine intensity data change sharply, and this law can still be seen in June when the data is generated, indicating that the model in this paper learned some drastic changes in the original data. In December, the sunshine intensity of the raw data remained at a high stable level, and the December period of the simulated weather still had this pattern, indicating that the model in this paper learned some stable changes in the original data.

As listed in Figure 5, the trend of the original data and the simulated weather is similar to the overall view, indicating that the model in this paper has learned the most
basic laws in the learning training. At the same time, we can see that there was an unusual and drastic change in the original temperature data in February, and in the simulated weather, there was also a drastic change in February, indicating that our model did learn unusual small probability events in learning training.

![Figure 5. Generating temperature data and raw temperature data](image)

### 3.3.2 Evaluation indicators

In this paper, PCA, Wasserstein distance, and mutual information are used as the three indicators of evaluation and simulated weather, representing diversity, authenticity, and similarity, respectively, and these three standards are of great significance for the subsequent development of source-grid-load-storage collaborative planning in energy meteorology.

#### (1) PCA

The raw data and simulated weather were visualized using PCA, and the distribution of sunlight intensity data was shown in Figure 6, and the distribution of temperature data was shown in Figure 7, where red was the distribution of raw data and blue was the distribution of simulated weather.
Figure 6. Visualization of the insolation intensity distribution

Figure 7. Distribution visualization of temperature data
It can be seen that in the 12-month PCA dimensionality reduction chart, most of the simulated weather contains the distribution of the original data, which indicates that the simulated weather has the characteristics of diversity. In Figure 6, in the subplot representing February and March, the blue dots representing simulated weather data and the red dots representing real weather data are very similar in the figure. In Figure 7, in the subplot representing January, February, March, August, December, this pattern is also shown. In other subplots of Figure 6 and Figure 7, the dense area of the blue point and the dense area of the red point is in the same location, which means that the simulation data and the real data are distributed similarly.

(2) Wasserstein distance

Table 1 shows the Wasserstein distance between the simulated weather sample and the original data sample trained by each model. The histogram charts in Figure 8 and Figure 9 provide a more visual representation of the numerical differences. Figure 8 represents the Wasserstein Distance for light intensity data, and Figure 9 represents the Wasserstein Distance for temperature data. It is clear from the histogram that the Wasserstein distance obtained by GRU-GAN models is smaller than the Wasserstein distance obtained by other models, which indicates that GRU-GAN model generates data that is closer to the original distribution. That is to say, the simulated weather has the characteristics of authenticity.

As listed in Table 1, "SI" stands for sunshine intensity, and "Temp" stands for Temperature.

<p>| Table 1: Wasserstein distances for the generated and raw data for the three methods |
|---------------------------------|-----|-----|-----|-----|-----|-----|-----|</p>
<table>
<thead>
<tr>
<th>Wasserstein distance</th>
<th>January</th>
<th>February</th>
<th>March</th>
<th>April</th>
</tr>
</thead>
<tbody>
<tr>
<td>SI</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.023</td>
<td>0.086</td>
<td>0.02</td>
<td>0.038</td>
</tr>
<tr>
<td>GAN</td>
<td>0.031</td>
<td>0.093</td>
<td>0.035</td>
<td>0.043</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.056</td>
<td>0.14</td>
<td>0.141</td>
<td>0.213</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Temp</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.017</td>
<td>0.103</td>
<td>0.103</td>
<td>0.078</td>
</tr>
<tr>
<td>GAN</td>
<td>0.028</td>
<td>0.07</td>
<td>0.086</td>
<td>0.073</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.058</td>
<td>0.14</td>
<td>0.148</td>
<td>0.08</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.028</td>
<td>0.373</td>
<td>0.035</td>
<td>0.086</td>
</tr>
<tr>
<td>GAN</td>
<td>0.042</td>
<td>0.413</td>
<td>0.051</td>
<td>0.093</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.154</td>
<td>0.398</td>
<td>0.1</td>
<td>0.101</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SI</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.056</td>
<td>0.061</td>
<td>0.046</td>
<td>0.083</td>
</tr>
<tr>
<td>GAN</td>
<td>0.061</td>
<td>0.073</td>
<td>0.052</td>
<td>0.093</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.132</td>
<td>0.091</td>
<td>0.201</td>
<td>0.111</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Temp</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.017</td>
<td>0.103</td>
<td>0.103</td>
<td>0.078</td>
</tr>
<tr>
<td>GAN</td>
<td>0.028</td>
<td>0.07</td>
<td>0.086</td>
<td>0.073</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.058</td>
<td>0.14</td>
<td>0.148</td>
<td>0.08</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>September</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.056</td>
<td>0.061</td>
<td>0.046</td>
<td>0.083</td>
</tr>
<tr>
<td>GAN</td>
<td>0.061</td>
<td>0.073</td>
<td>0.052</td>
<td>0.093</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.132</td>
<td>0.091</td>
<td>0.201</td>
<td>0.111</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>October</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.017</td>
<td>0.103</td>
<td>0.103</td>
<td>0.078</td>
</tr>
<tr>
<td>GAN</td>
<td>0.028</td>
<td>0.07</td>
<td>0.086</td>
<td>0.073</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.058</td>
<td>0.14</td>
<td>0.148</td>
<td>0.08</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>November</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.056</td>
<td>0.061</td>
<td>0.046</td>
<td>0.083</td>
</tr>
<tr>
<td>GAN</td>
<td>0.061</td>
<td>0.073</td>
<td>0.052</td>
<td>0.093</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.132</td>
<td>0.091</td>
<td>0.201</td>
<td>0.111</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>December</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.017</td>
<td>0.103</td>
<td>0.103</td>
<td>0.078</td>
</tr>
<tr>
<td>GAN</td>
<td>0.028</td>
<td>0.07</td>
<td>0.086</td>
<td>0.073</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.058</td>
<td>0.14</td>
<td>0.148</td>
<td>0.08</td>
</tr>
</tbody>
</table>
(3) Mutual information
Table 2 shows the mutual information obtained by the three methods. As can be seen from the table, Among the 12 months of generated data, the mutual information of the simulated weather data of the GRU-GAN model is the closest to the mutual information of the original weather data, which indicates that the data generated by the GRU-GAN models can better retain the implied relationship between sunshine intensity and temperature data.

<table>
<thead>
<tr>
<th>Month</th>
<th>January</th>
<th>February</th>
<th>March</th>
<th>April</th>
<th>May</th>
<th>June</th>
</tr>
</thead>
<tbody>
<tr>
<td>raw data</td>
<td>0.563</td>
<td>0.694</td>
<td>0.702</td>
<td>0.597</td>
<td>0.587</td>
<td>0.736</td>
</tr>
<tr>
<td>GRU-GAN</td>
<td>0.554</td>
<td>0.744</td>
<td>0.715</td>
<td>0.644</td>
<td>0.630</td>
<td>0.776</td>
</tr>
<tr>
<td>GAN</td>
<td>0.578</td>
<td>0.754</td>
<td>0.765</td>
<td>0.341</td>
<td>0.675</td>
<td>0.783</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.676</td>
<td>0.622</td>
<td>0.667</td>
<td>0.701</td>
<td>0.743</td>
<td>0.787</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Month</th>
<th>July</th>
<th>August</th>
<th>September</th>
<th>October</th>
<th>November</th>
<th>December</th>
</tr>
</thead>
<tbody>
<tr>
<td>raw data</td>
<td>0.674</td>
<td>0.726</td>
<td>0.694</td>
<td>0.636</td>
<td>0.700</td>
<td>0.696</td>
</tr>
</tbody>
</table>

Figure 8: Wasserstein distance between the lighting data and the raw data

Figure 9. Wasserstein distance between the temperature data and the raw data
<table>
<thead>
<tr>
<th>Method</th>
<th>Diversity</th>
<th>Authenticity</th>
<th>Relevance</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRU-GAN</td>
<td>0.731</td>
<td>0.792</td>
<td>0.426</td>
</tr>
<tr>
<td>GAN</td>
<td>0.752</td>
<td>0.742</td>
<td>0.323</td>
</tr>
<tr>
<td>Statistical learning</td>
<td>0.755</td>
<td>0.686</td>
<td>0.424</td>
</tr>
</tbody>
</table>

The above three indicators comprehensively evaluate the quality of the simulated weather in terms of diversity, authenticity, and relevance. PCA shows the diversity of generated data, and the distribution of simulated weather almost all contains the distribution of original data, and the dense location of distribution ensures the diversity and authenticity of simulated weather; Wasserstein distance reflects the difference in distribution between the simulated weather and the original data, and of the three methods, the proposed method performs best and the Wasserstein distance is the smallest; The mutual information reflects whether the relationship between sunshine intensity and the temperature has been preserved, and the mutual information of the proposed method is closest to the original data, indicating that the intrinsic relationship between the two variables is best preserved.

### 3.3.3 Test data results

To verify the feasibility of the model, a set of data from January 2021 to 2023 was also selected, including 24-hour sunshine intensity data and temperature data, with a total of 93 pieces of data. Then the data of this period is entered into the model, and the simulated meteorological data is obtained. The PCA of these two data are shown in Figure 10.

On the sunlight intensity variable: the Wasserstein distance between generating data and real data is 0.47, and on the temperature variable, the Wasserstein distance between generating data and real data is 0.13, indicating that the distribution of the two is very close, which means that our model is extensive and can be generalized to untrained.

Among the test data, the cross-correlation between sunshine intensity and temperature variables is 0.65, and the cross-correlation of meteorological data obtained by the model is 0.76, which is very close, indicating that our model retains the correlation between the two variables well.

![Figure 10. the results of the test data](image-url)
In this paper, a data generation model of GRU-GAN is proposed and applied to the stochastic simulation of weather data. In the power system, climate change will affect the stability of the power supply of the power system, which is very important for weather-grid operation analysis. The simulation results are diverse, realistic, and relevant. Specifically, in the PCA diagram of the simulation results, the distribution of the simulated weather is almost the same as the distribution of the original data, and the range is larger than that of the original data, reflecting the characteristics of the diversity of the generated data. The small Wasserstein distance between the generated data and the original data means that the distribution of the simulated weather is closest to the original data distribution and has authenticity; The mutual information between the insolation intensity variable and the temperature variable in the simulated weather is closest to the original data, meaning that the correlation between the two variables simulating the weather is preserved.

In this paper, GRU is innovatively used as a generator and discriminator of GAN, and the memory mechanism is introduced into the stochastic simulation model of weather data. Combined with the powerful data generation capability of GAN, a set of weather data in line with the original data distribution is obtained. The evaluation system is composed of three evaluation indicators: PCA, Wasserstein distance, and mutual information, which can comprehensively measure the diversity, authenticity, and relevance of the simulated weather.

In recent years, generative artificial intelligence has developed rapidly, and models based on GAN and VAE have been increasingly applied to the field of simulation. In March 2023, Adobe launched its own generative AI tool, Firefly, which allows users to generate a variety of creative images using natural language. This paper is a simulation of real weather scenarios, but in power system planning, the impact of extreme weather is very large, and more often, we hope to accurately simulate extreme weather scenarios. Implementing accurate simulations of extreme weather is a challenging task. In the future, based on the model of this paper, we will build a model that accurately simulates extreme weather, to better plan the power system, predict extreme weather in advance, and avoid power resource shortages.

**Conflict of Interest**

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

**Author Contributions**

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

**Funding**

This study is supported by the National Natural Science Foundation of China under Grant 52007193 and The 2115 Talent Development Program of China Agricultural University.
REFERENCES


