

Article

Gated Recurrent Unit Network-Based Short-Term Photovoltaic Forecasting

Yusen Wang ¹, Wenlong Liao ^{2,*} and Yuqing Chang ²

¹ School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden; yusenw@kth.se

² Key Laboratory of Smart Grid of Ministry of Education, Tianjin University, Tianjin 300072, China; ciecezyb2014@163.com

* Correspondence: wenlongliao@cau.edu.cn

Received: 19 July 2018; Accepted: 15 August 2018; Published: 18 August 2018



Abstract: Photovoltaic power has great volatility and intermittency due to environmental factors. Forecasting photovoltaic power is of great significance to ensure the safe and economical operation of distribution network. This paper proposes a novel approach to forecast short-term photovoltaic power based on a gated recurrent unit (GRU) network. Firstly, the Pearson coefficient is used to extract the main features that affect photovoltaic power output at the next moment, and qualitatively analyze the relationship between the historical photovoltaic power and the future photovoltaic power output. Secondly, the K-means method is utilized to divide training sets into several groups based on the similarities of each feature, and then GRU network training is applied to each group. The output of each GRU network is averaged to obtain the photovoltaic power output at the next moment. The case study shows that the proposed approach can effectively consider the influence of features and historical photovoltaic power on the future photovoltaic power output, and has higher accuracy than the traditional methods.

Keywords: photovoltaic power forecasting; GRU network; Pearson coefficient; K-means

1. Introduction

With the reduction of fossil energy supply and increased environmental pollution, the utilization of renewable energy to achieve sustainable and healthy development of the economy has become the consensus of countries all around the world. For example, solar radiation as a natural energy source has the following advantages: no noise, no pollution, inexhaustibility and so on. By 2016, the total installed capacity of photovoltaic power stations in the world had exceeded 300 GW [1]. However, the intermittence and volatility of photovoltaic systems bring challenges to the security, stability and economic operation of the power system. Therefore, it is necessary to predict photovoltaic power output, and with the help of energy storage systems, improve the operating performance of the power system.

With respect to time horizons, photovoltaic forecasting can be divided into very short-term photovoltaic forecasting, short-term photovoltaic forecasting, medium-term photovoltaic forecasting, and long-term photovoltaic forecasting, for which the forecasting horizon cut-offs are minutes, hours, months and years, respectively. Many methods for forecasting short-term photovoltaic power have been proposed in the past few years. These methods mainly include autoregressive integrated moving average model (ARIMA), regression analysis, gray system model, support vector machine (SVM), back propagation neural network (BP) and so on. The ARIMA model forecasts the photovoltaic power output at the next moment based on the trend of historical time series, which requires high stability of the historical power output series. In addition, the influence of external features, such as

temperature, humidity and visibility of the photovoltaic power, is not taken into account which leads to a large prediction error [2,3]. The regression analysis models are relatively simple and clear, but they can't describe the relationship between the input features and the photovoltaic power output with accurate mathematical formulas [4]. The GM (1,1) method in the gray system is the most widely used method for prediction. It does not require a large number of samples and has low computational cost. Nevertheless, the GM (1,1) method also has the disadvantage of not considering the impact of external features on the results [5]. SVM is a supervised learning method, which is often used for classification and regression analysis. Compared to traditional methods, it can solve problems which involve non-linearity, convergence to local optima, and high dimensionality. However, if the amount of sample data is large, the SVM method will have slow training speed and low prediction accuracy [6,7]. BP neural networks have strong nonlinear mapping ability and flexible network structures. The number of hidden layers and cells in each layer can be set according to the specific photovoltaic power situation, but they have the shortcomings of slow learning speed and easily falling into local optima [8]. Moreover, the SVM and BP neural networks can only determine the value of the photovoltaic power at the next moment according to the main features, and they ignore the impact of historical trends on the future photovoltaic power output.

With the rapid development of deep learning, deep learning techniques have become some of the most popular research fields in academia and industry [9]. Compared with shallow learning, deep learning is a set of machine learning techniques with multi-layer deep artificial neural network (ANN) architectures that use stacked layers of transformation trainable from the beginning to the end. The common deep learning architectures for forecasting photovoltaic power output include Boltzmann machines, deep belief networks (DBN) and recurrent neural networks (RNN). Neo proposed DBN to determine the parameters that best fit the data to obtain the least prediction error. The study case showed that DBN has higher forecasting accuracy than traditional methods [10]. A Bayesian neural network model was proposed to forecast solar irradiation in [11]. The effectiveness of the proposed model is verified by comparing with the traditional algorithm. Cao combined an artificial neural network and wavelet analysis to improve the forecasting accuracy [12]. Although the above methods have achieved some positive results, the impact of previous information on the output is not considered, which is not suitable for forecasting photovoltaic power output. In order to solve this problem, some scholars have applied a long short time memory (LSTM) network to predict photovoltaic power output, and the study cases show that LSTM network presented better performance than traditional networks [13,14]. Although the accuracy of LSTM is higher than that of other algorithms, the training time of LSTM is much longer than that of other algorithms. How to reduce the training time under the premise of ensuring the high accuracy is still a challenge worth studying. The gated recurrent unit is a special case of LSTM. It has shorter training time than LSTM. At present, GRU networks are mainly used in classification, and are seldom applied in regression problems [15,16].

In this paper, a GRU neural network is designed to forecast photovoltaic power output, so as to reduce the training time and improve the accuracy. The key contributions of this paper are as follows:

- (1) The Pearson coefficient method is used to extract the main features that affect the photovoltaic power and analyze the relationship between historical photovoltaic power and the future photovoltaic power output.
- (2) The K-means method is utilized to divide the data into several groups based on the similarity of the features, so as to improve the accuracy of prediction.
- (3) The GRU network that can simultaneously consider the influence of features and historical photovoltaic power output trend on the future photovoltaic power output is designed for forecasting short-term photovoltaic power. It not only inherits the advantages of LSTM network, but also shortens training time.

The rest of the paper is organized as follows: Section 2 briefly introduces the framework of the proposed approaches. Section 3 explains how to use Pearson coefficient method to extract main

features and classify data according to the similarity of features. Section 4 explains the basic principle of GRU network and designs a GRU network to predict photovoltaic power. Section 5 discusses the simulations and results. The conclusion is described in Section 6.

2. The Forecasting Framework of Proposed Approaches

Figure 1 shows the framework of proposed approaches. As can be seen from the figure, the approach proposed in this paper includes three key steps. Firstly, the min-max normalization is used to normalize the original data, so as to eliminate any influence of dimensionality. The Pearson coefficient is used to measure the correlation between the variables and the photovoltaic power, and the main features affecting the photovoltaic power are extracted according to the size of the Pearson coefficients. Secondly, the K-means method is utilized to divide data into several groups based on the similarity of features. Several GRU networks are trained in each group. Finally, the distance between the current object and the center of each group is calculated, and the group that is closest to the current object is selected. The next moment's photovoltaic power output is obtained by averaging the outputs of multiple GRU networks that belong to the selected group.

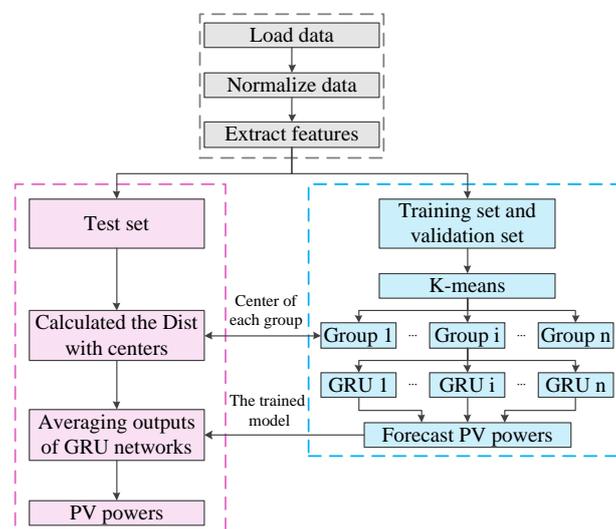


Figure 1. The framework of the proposed approach.

3. Feature Extraction and Cluster Analysis

3.1. Introduction of the Dataset

The data set is from the Global Energy Forecasting Competition 2014 [17], including the hourly photovoltaic power from 1 April 2012 to 1 July 2014. In addition, 12 weather variables are obtained from the European Centre for Medium-range Weather Forecasts. These variables are shown in Table 1.

3.2. Feature Extraction

There are many factors that affect the next moment's photovoltaic power output, such as pressure, humidity, light intensity, cloud, and so on. Some existing methods, such as the ARIMA and GM (1,1) rely on the trend of historical time series to predict photovoltaic power without considering the influence of various external features. These methods have the obvious defect that it is difficult to adapt to the mutability of the environment, especially for the inflection point. Considering the influence of various features on the future photovoltaic power output is beneficial to improve the prediction accuracy. However, if the features that are not related to photovoltaic power are put into the model, it will not only increase the complexity of the algorithm, but also interfere with the model parameters. Therefore, it is necessary to extract features before building models.

Table 1. Data variables and description.

Variable Name	Unit
Total column liquid water (TCLW)	kg·m ⁻²
Total column ice water (TCIW)	kg·m ⁻²
Surface pressure (SP)	Pa
Relative humidity at 1000 mbar (R)	%
Total cloud cover (TCC)	0–1
10-m U wind component (10 U)	m·s ⁻¹
10-m V wind component (10 V)	m·s ⁻¹
2-m temperature (2 T)	K
Surface solar rad down (SSRD)	J·m ⁻²
Surface thermal rad down (STRD)	J·m ⁻²
Top net solar rad (TSR)	J·m ⁻²
Total precipitation (TP)	m

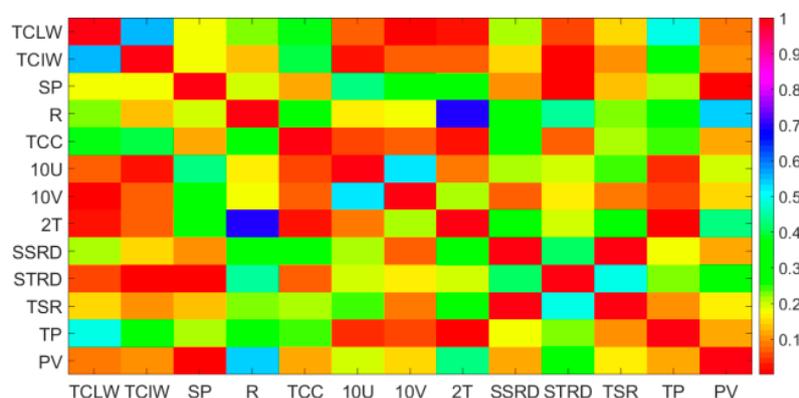
Extracting features is an important step in data mining. The existing methods for forecasting photovoltaic power often rely on experience to select the main features. These methods are subjective, and the features that affect photovoltaic power in different time and space are not the same. It is necessary to use a quantitative method to find the key features affecting photovoltaic power.

At present, there are many ways to evaluate the correlation between the two indicators, such as Pearson coefficient, chi-square test, mutual information, grey relational analysis and so on. Pearson coefficient is a simple and effective method to reflect the linear correlation of two indexes, which is widely used to extract features in the field of forecasting load and forecasting wind power. In this paper, Pearson coefficient is used to measure the correlation between each feature and photovoltaic power. The mathematical formula of Pearson coefficient can be described as follows:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (1)$$

where r_{xy} is the Pearson coefficient between x and y . \bar{x} is the mean of x , and \bar{y} is the mean of y . Pearson coefficient ranges from -1 to $+1$, where -1 is total negative linear correlation, 0 is no linear correlation, and $+1$ is total positive linear correlation.

Taking the above 12 variables as the original features, the Pearson coefficient is used to evaluate the correlation between each feature and the photovoltaic power. The absolute values of Pearson coefficient are shown in Figure 2.

**Figure 2.** The Pearson coefficient between each feature and photovoltaic power.

It can be seen from Figure 2 that the Pearson coefficients between total column liquid water, total column ice water, surface pressure and photovoltaic power output are small, so these three features are excluded and the remaining nine features are retained.

In order to qualitatively analyze the influence of the historical photovoltaic power on the next photovoltaic power, the photovoltaic power in the past n times is set as $P = (P_{t-1}, P_{t-2}, \dots, P_{t-n})$, and the absolute values of Pearson coefficient are calculated and are shown in Figure 3, where a large correlation between next moment's photovoltaic power output and historical photovoltaic power output can be seen. The absolute values of the Pearson coefficient decrease first and then increase from $t - 1$ to $t - 12$, and reach a minimum value at $t - 5$. This shows that the next photovoltaic power is mainly related to the historical photovoltaic power from $t - 1$ to $t - 4$. According to the above analysis, we should consider both the features of weather and historical photovoltaic power output, so that the accuracy can be improved.

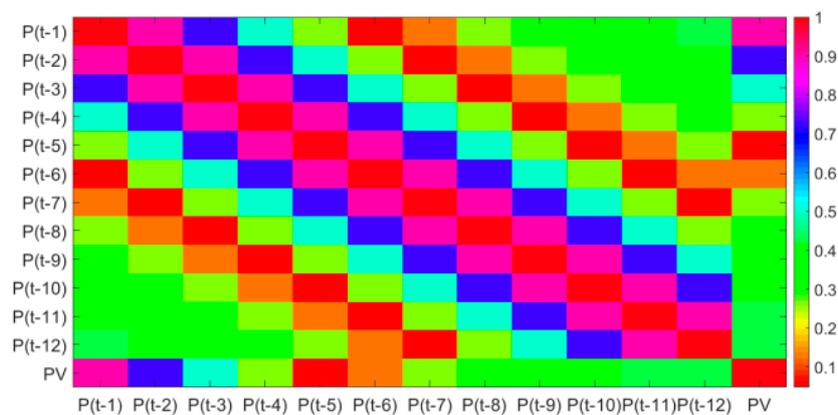


Figure 3. The Pearson coefficient relationship between the historical photovoltaic power and the next photovoltaic power.

3.3. Cluster Analysis

When the value of each feature is different, so their impact on the output is different. Clustering the data according to the similarity of the features is helpful to improve the accuracy in the field of forecasting load and forecasting wind power [18,19]. In this paper, we try to divide training sets into several groups based on the similarities of each feature, and analyze whether grouping the data helps improve the accuracy in the field of photovoltaic power forecasting.

There are many clustering methods in the existing research. The clustering effect of each method depends on the application background and data set. The K-means method is a classical clustering algorithm with good robustness [20]. It has been widely used in the stage of data processing. In this paper, the K-means method will be used to group the training set. The goal is to make the features of the same group similar, and the features of different groups are quite different. The specific steps for K-means clustering are as follows:

- (1) Initializing the K centers of K groups: To eliminate the influence of dimension, the min-max normalization is used to standardize each feature. K samples are randomly selected as the initial centers of each group.
- (2) Assigning each sample to each group: The Euclidean distance between each sample and the center of each group is calculated, and each sample is allocated to the nearest group.
- (3) Recalculating the center of each group: The center of each group is recalculated based on the sample data of each group, and the results will be output if all the centers are not changed. Otherwise, return to step (2).

4. The Forecasting Framework Based on GRU Network

4.1. The GRU Network

Unlike the traditional BP neural network and the convolution neural network, the input of the hidden layer of RNN contains not only the output of the upper layer, but also the output between the hidden layer nodes of the same layer at the last time. Therefore, the RNN can effectively improve the accuracy of prediction by using the historical photovoltaic powers. Theoretically, the length of the historical photovoltaic series can be arbitrary. A large number of experiments show that traditional RNN will encounter many problems when learning long-range dependencies. Among them, the most common problem is gradient vanishing or exploding [21].

To solve this problem, Reiter and Schmidhuber proposed a LSTM network that reads and modifies memory cells by controlling input gates, forgetting gates, and output gates, and then uses different functions to update the state of hidden layer [22]. Up to now, LSTM is one of the most popular RNN architectures, and has achieved great success in the fields of wind power forecasting, load forecasting and photovoltaic power forecasting. Although the accuracy of LSTM is higher than that of other algorithms, the training time of LSTM is longer than that of other algorithms. How to reduce the training time is still a problem worth studying. The gated recurrent unit is a special case of LSTM proposed by Cho in 2014 [23]. Its performance in speech signal modeling was found to be similar to that of long short-term memory. In addition, it has shorter training time than traditional LSTM and has fewer parameters than LSTM, as they lack an output gate. Figure 4 shows the structure of gated recurrent unit network.

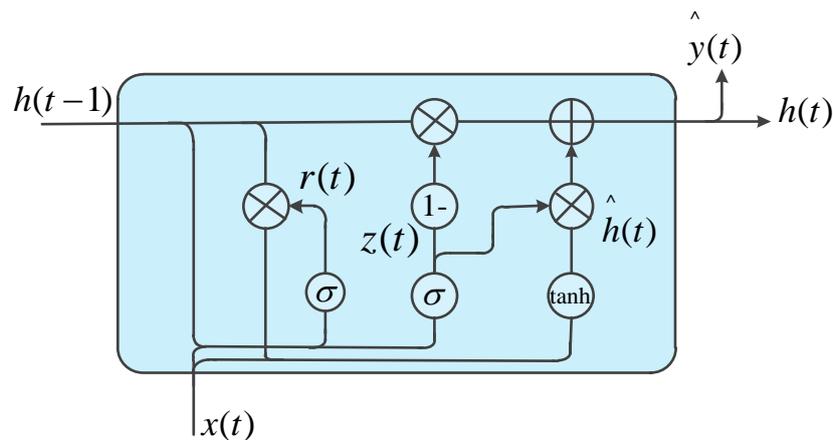


Figure 4. The structure of gated recurrent unit (GRU) network.

There are two input features at each time, which include the input vector $x(t)$ and previous output vector $h(t-1)$. The output of each gate can be obtained through logical operation and nonlinear transformation of input. The relationship between input and output can be described as follows:

$$r(t) = \sigma_g(W_r x(t) + U_r h(t-1) + b_r) \quad (2)$$

$$z(t) = \sigma_g(W_z x(t) + U_z h(t-1) + b_z) \quad (3)$$

$$h(t) = (1 - z(t)) \circ h(t-1) + z(t) \circ \hat{h}(t) \quad (4)$$

$$\hat{h}(t) = \sigma_h(W_h x(t) + U_h (r(t) \circ h(t-1)) + b_h) \quad (5)$$

where $z(t)$ is the update gate vector, $r(t)$ is the reset gate vector, W , U and are parameter matrices and vector. σ_g is a sigmoid function and σ_h is a hyperbolic tangent.

After building the structure of the GRU network, the training method for the GRU network needs to be determined. At present, for recurrent neural networks such as GRU, the popular training methods include back propagation through time (BPTT) and real time recurrent learning (RTRL). The BPTT has high computational efficiency and its computation time is shorter than RTRL [24]. Therefore, the BPTT is selected to train GRU network. In addition, previous studies show that the Adam optimizer has better performance than other algorithms which include stochastic gradient descent (SGD), RMSProp, Adadelta and Adagrad [25]. Thus the Adam algorithm is used for training the proposed forecasting framework.

4.2. The Process for Forecasting Photovoltaic Power Based on GRU Network

The process for forecasting photovoltaic power based on GRU network is shown in Figure 5. The main steps are as follows:

- (1) Set a historical photovoltaic power series $P = (P_{t-1}, P_{t-2}, \dots, P_{t-n})$ that will be used to predict the next photovoltaic power. The matrix X consists of historical photovoltaic power and 9 features that include R, TCC, 10U, 10V, 2T, SSRD, STRD, TSR and TP.
- (2) Every row of the matrix X is the scaled features and the time step is fed to corresponding GRU block in the GRU layer. Since the sequential nature of the output of a GRU layer, the number of GRU layers that are stacked to form a recurrent neural network can be arbitrary.
- (3) The output of the top GRU layer are fed to a feed forward neural network that maps the output of GRU layer to photovoltaic power.

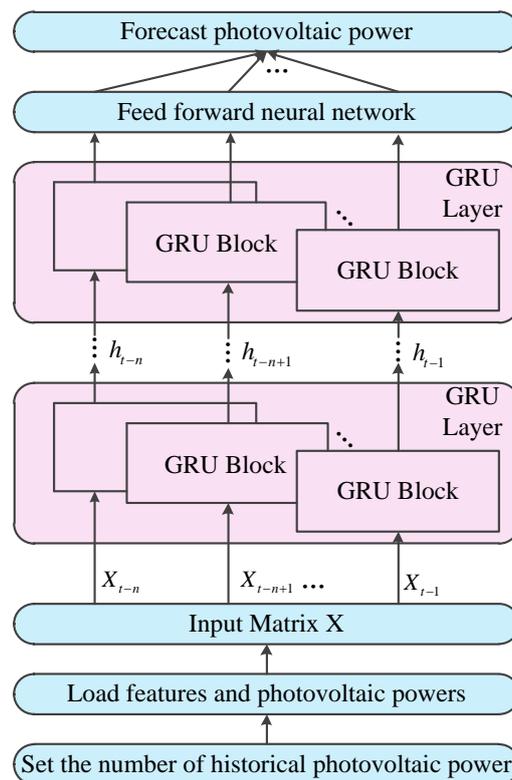


Figure 5. The process forecasting photovoltaic power based on GRU network.

4.3. Program Implementation

The program of GRU network is designed with multiple stages: (1) Define a network. (2) Compile the network. (3) Fit the network (4) Forecasting photovoltaic power. The partial codes for building the GRU network are shown in Table 2.

Table 2. GRU architecture based on Keras.

Program: Codes for Building the GRU Network
<pre> #1. Define Network from keras.models import Sequential from keras.layers import Dense from keras.layers.recurrent import GRU model = Sequential() model.add(GRU(units=10,input_shape=(trainX.shape [1], trainX.shape[2]),return_sequences=True)) model.add(GRU(units=10,return_sequences=True)) model.add(GRU(units=10)) model.add(Dense(units=1, kernel_initializer='normal',activation='sigmoid')) #2. compile the network model.compile(loss='mae', optimizer='adam') #3. fit the network history=model.fit(trainX,trainY,epochs=100,batch_size=10,validation_data=(validX,validY),verbose=2,shuffle=False) #4. forecasting photovoltaic power PV = model.predict(testX) </pre>

4.4. Indicators for Evaluating the Results

The popular indicators for evaluating the prediction results include root mean square error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE). MAPE is not suitable for evaluating the prediction results in this case, because the photovoltaic power output is equal to 0 sometimes, and MAPE will become infinite, so in this study, MAE and RMSE will be adopted to evaluate the accuracy of prediction. In addition, the MAE will be used as a loss function of GRU network. The mathematical formulas for MAE and RMSE are as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (6)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (7)$$

where n is the number of test sets, y_i is the real photovoltaic power and \hat{y}_i is the forecasted photovoltaic power. It should be noted that MAE and RMSE are equal when n is equal to 1.

5. Case Study

In practical applications, since the training time of the neural network is very long, each network is trained in advance using historical data. In this paper, the inputs of the GRU network include the environmental factors of the previous moment, the time and the photovoltaic powers of the past few hours, and the output is the photovoltaic power of the next hour.

The dataset from the Global Energy Forecasting Competition 2014 is used to verify the proposed approaches. The photovoltaic powers and corresponding features from 1 April 2012 to 9 April 2014 are used for training set and the data from 10 April 2014 to 20 May 2014 are used for validation set. The data from 21 May 2014 to 1 July 2014 are regarded as test data. All the approaches will be conducted using Keras on a laptop equipped with an Intel(R) Core(TM) i3-3110M 2.40-GHz processor and 6 GB of RAM.

In order to verify the effectiveness of the proposed approaches, we compared the proposed approaches with three benchmarking methods which include SVM, BP network and ARIMA. The parameters of these algorithms are set as follows:

- (1) The number of neurons in the input layer of the GRU is equal to the sum of the number of features plus the number of historical photovoltaic power. The output layer with sigmoid activation function has one neuron. After several experiments, the best choice is to use one GRU layer,

and the number of neurons is 15. The epochs are set to 100. In addition, LSTM uses the same parameters as GRU.

- (2) After several experiments, the best choice of BP network is to use two hidden layers, and the number of neurons in each layer is 15 and 5 respectively. The epochs are set to 100.
- (3) The radial basis function (RBF) is used as a kernel function for SVM. (4) After several experiments, the best parameters of ARIMA are set as follows: The number of autoregressive terms (p) is equal to 4. The degree of differencing (d) is equal to 2. The number of lagged forecast errors in the prediction equation (q) is set to 4.

5.1. The Optimal Number of Groups

In order to explore the impact of the number of groups on the forecasting results, the number of groups is changed from 1 to 10. Even if the number of groups is the same, the K-means result is different each time. Therefore, experiments are performed 50 times for each number. The RMSE and MAE are shown in Table 3.

It can be seen from Table 3 that grouping the training set is conducive to improving the prediction accuracy. The forecasting error is reduced as the number of groups increases. When the range of values of the features is closed, the relationship between the features and the photovoltaic power output is also similar. It is reasonable to train GRU networks using training set with similar features.

Table 3. Results in different number of groups.

Group No.	MAE/MW	RMSE/MW	Group No.	MAE/MW	RMSE/MW
1	0.0409	0.0725	6	0.0396	0.0701
2	0.0407	0.0717	7	0.0395	0.0701
3	0.0404	0.0725	8	0.0395	0.0698
4	0.0409	0.0718	9	0.0379	0.0683

5.2. The Optimal Step Size

The GRU network can predict the next photovoltaic power output by using the photovoltaic power features and historical time series. Theoretically, the length of the time series can be arbitrary. If the time series is too short, it will lead to a lack of historical information learning. On the contrary, if the time series is too long, it will increase the complexity of the algorithm and even make the prediction worse.

To determine how many objects in the historical photovoltaic power output series should be used to forecast short-term photovoltaic power, the number of step sizes was ranged from 0 to 12. The averages of MAE and RMSE are obtained by repeating the experiment 30 times. Figures 6 and 7 show the performance of MAE and RMSE under different step sizes.

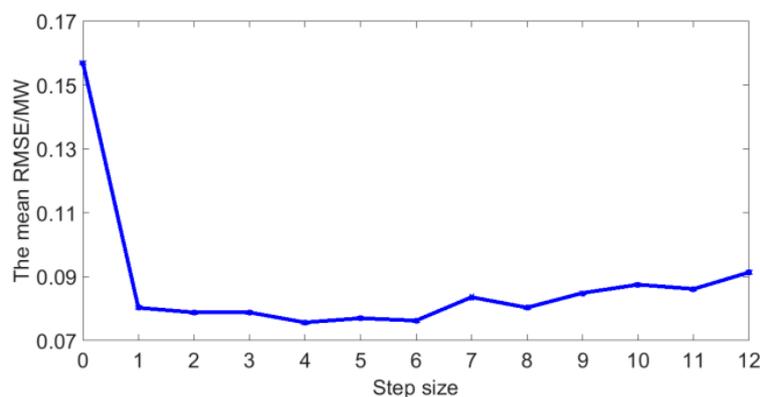


Figure 6. The root mean square error (RMSE) of photovoltaic power in different step size.

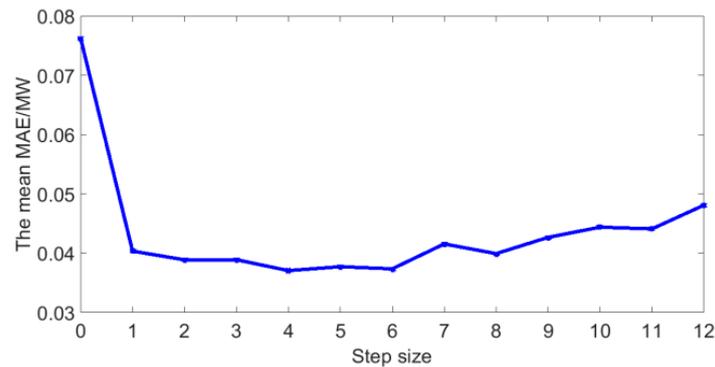


Figure 7. The mean absolute error (MAE) of photovoltaic power in different step size.

It can be seen from Figures 6 and 7 that considering historical time series of photovoltaic power can significantly reduce the forecasting error. Besides, with the increase of historical photovoltaic power output, the overall trend of MAE and RMSE decreases first and then increases. If the step size is too large, it will not only increase the complexity of the algorithm, but also lead to an excessive reliance on historical time series, which will lead to a drop in the accuracy of the prediction. When the step size is too small, the trend of time series is not comprehensively considered. In this dataset, the optimal step size of historical photovoltaic power is 4. The next moment's photovoltaic power has a strong relationship with the historical photovoltaic power output in the past 4 moments, which is consistent with the conclusions inferred from Pearson coefficient.

5.3. Comparison with Traditional Methods

In order to fully verify the effectiveness of the proposed approaches, we will compare the proposed approaches with LSTM network, BP network, SVM and ARIMA. Experiment for each method will be repeated 30 times independently to ensure objectivity of the results. The statistical results are shown in Figures 8–11 and Table 4.

The box plot depicted in Figure 8a shows the RMSE of each sample in the test set. The average RMSE of LSTM, GRU, BP, SVM and ARIMA are 0.037, 0.036, 0.105, 0.170 and 0.101, respectively. Red marks represent outliers, which are greater than the maximum or less than the minimum. To further analyze the outliers, we randomly selected a day that included multiple outliers, and predicted the photovoltaic powers of the day. The forecasted photovoltaic powers of each algorithm are shown in Figure 8b. Obviously, the forecasting accuracy of GRU is slightly better than that of other algorithms. The volatility of the photovoltaic output power on this day is very large, which leads to a large deviation between real photovoltaic power and forecasted photovoltaic power of each algorithm.

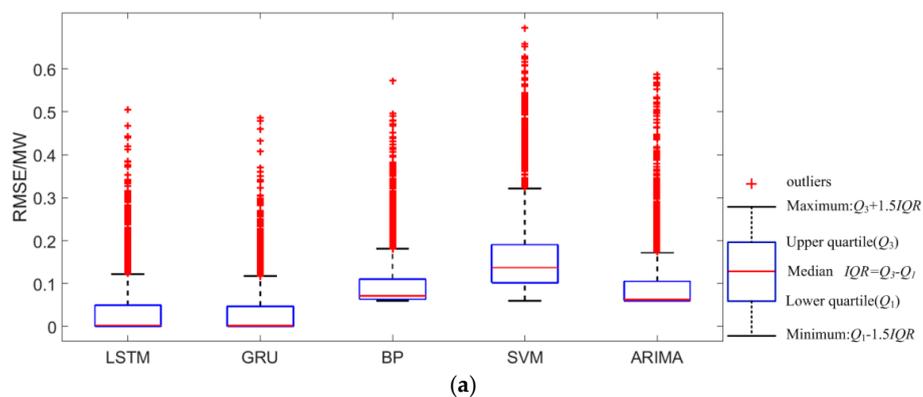


Figure 8. Cont.

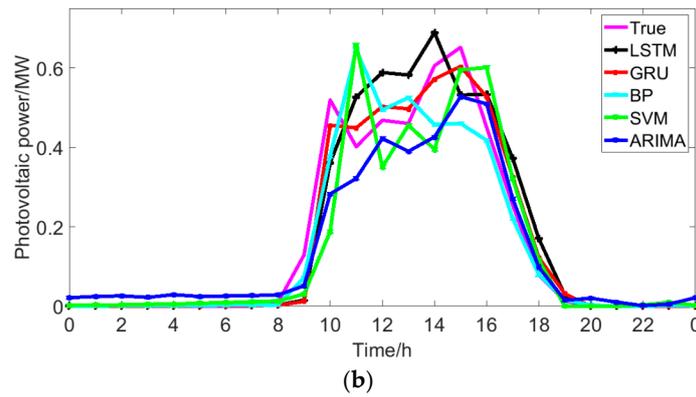


Figure 8. The performance of each method RMSE of each method. (a) The RMSE of each method; (b) Photovoltaic power of 28 June 2014.

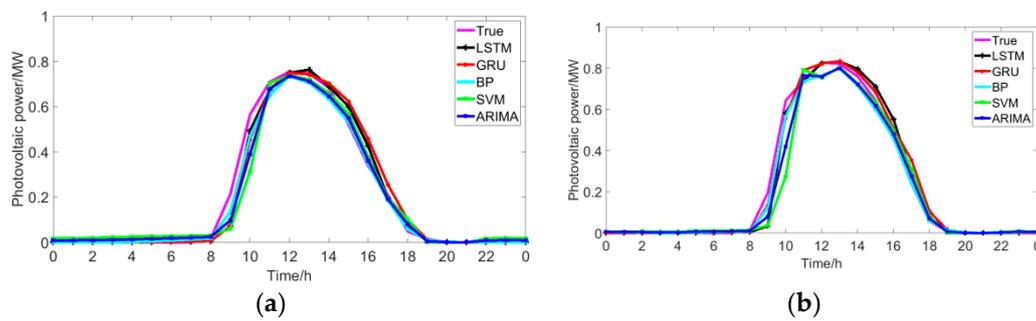


Figure 9. The real photovoltaic power and forecasted photovoltaic power selected from training set randomly. (a) Photovoltaic power of 14 May 2013; (b) Photovoltaic power of 5 April 2014.

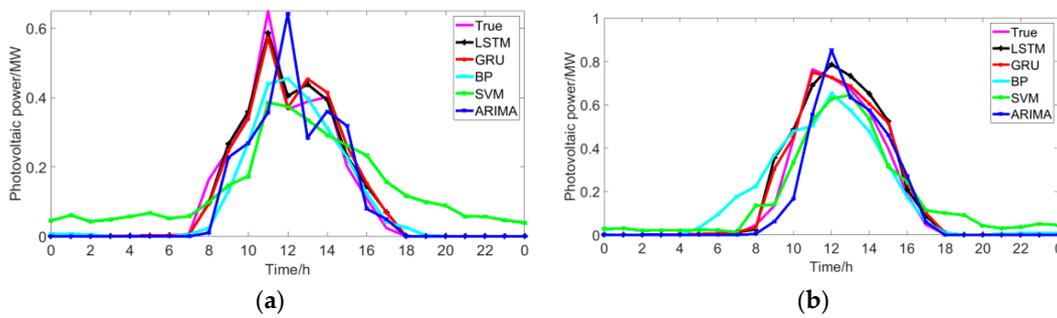


Figure 10. The real photovoltaic power and forecasted photovoltaic power selected from validation set randomly. (a) Photovoltaic power of 21 April 2014; (b) Photovoltaic power of 17 May 2014.

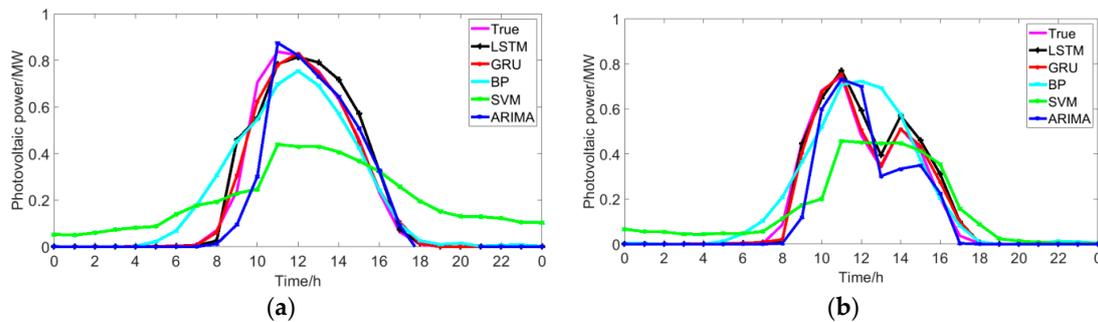


Figure 11. The real photovoltaic power and forecasted photovoltaic power selected from test set randomly. (a) Photovoltaic power of 17 June 2014; (b) Photovoltaic power of 23 June 2014.

Table 4. The training time of each method.

Method	The Best Case/s	The Worst Case/s	The Average Case/s	Standard Deviation/s
LSTM	393.01	400.57	396.27	1.80
GRU	354.92	379.57	365.40	7.25
BP	7.87	14.55	10.62	1.58
SVM	30.03	32.05	30.65	0.35
ARIMA	2.64	5.99	3.66	0.67

In addition, Figure 8a shows the maximum RMSE of the GRU is smaller than that of other algorithms. In general, the ARIMA predicts the next moment's photovoltaic power based on the trends of historical time series, and does not consider the influence of environmental factors, so the prediction error is large. Although BP and SVM can take the environmental factors into account, they cannot give consideration to the impact of historical time series on next moment's photovoltaic power, so the error is larger than that of GRU and LSTM. The forecasting accuracy of GRU is higher than that of the other four algorithms and the predictive results of LSTM and GRU are very close. This is because LSTM and GRU can not only consider the influence of features on photovoltaic power output at the next moment, but also improve the accuracy of prediction by using historical time series.

In order to further illustrate the advantages of GRU, each algorithm is tested with a training set, validation set and test set, respectively. Figure 9 shows that each algorithm has a good prediction effect when the training set is selected as the test data, because the parameters of each algorithm are determined by the training set. As shown in Figures 10 and 11, if the validation set and test set are used to test the algorithm, the prediction errors of BP, SVM and ARIMA are very large, which indicates that these algorithms are overly dependent on the training set, leading to poor generalization ability. Compared with other algorithms, the prediction errors of LSTM and GRU are relatively low, which indicates their adaptability to fresh samples.

Table 4 shows that the longest training time of GRU is shorter than the best training time of the LSTM, which fully demonstrates that GRU is better than LSTM in term of training time. The GRU network does not have to use a memory unit to control the flow of information like the LSTM network and it can directly make use of all the hidden states without any control. Therefore, the training time of GRU is slightly shorter than that of LSTM. In addition, it is obvious that the average training time of LSTM and GRU is much longer than that of the other three traditional algorithms. Since the models of these algorithms are trained offline before taken into use, the training time does not have a decisive influence on which algorithm is selected to forecast photovoltaic power. Therefore, it is feasible to predict short-term photovoltaic power using trained GRU network. Considering training time and forecasting accuracy, the GRU network is the best choice for forecasting short-time photovoltaic power.

6. Conclusions

The paper tries to improve the accuracy for forecasting short-time photovoltaic power output using a GRU network. Firstly, the Pearson coefficient is used to extract the main features that affect photovoltaic power output at the next moment, and qualitatively analyze the relationship between the historical photovoltaic power output and photovoltaic power output at the next moment. Secondly, the training sets are divided into several groups based on the similarities of each feature, and then GRU network training is applied to each group. The output of each GRU network is averaged to obtain the next photovoltaic power. The experiments allow us to reach the following conclusions:

- (1) Grouping training sets based on the similarity of input features and using the GRU network of the group to which the current object belongs can improve the accuracy of the prediction. The error decreases as the number of groups increases.
- (2) The Pearson coefficient can not only extract the main features that affect the photovoltaic power, but also qualitatively analyze the relationship between historical photovoltaic power and next

moment's power. Through qualitative analysis and quantitative analysis, it is found that a suitable number of historical time series of photovoltaic power can improve the forecasting accuracy. In this dataset, the optimal number of historical time series is 4.

- (3) As for the forecasting accuracy, the GRU network can simultaneously consider the influence of features and historical photovoltaic power output on the next moment's photovoltaic power, which leads to a higher accuracy of the GRU than that of BP, SVM, ARIMA, and LSTM. In addition, compared with LSTM, GRU has fewer parameters and shorter training time. Compared to LSTM, the advantage of GRU is even more obvious when the data set is particularly large.

Author Contributions: Data curation, W.L.; Formal analysis, Y.C.; Software, Y.C.; Supervision, W.L.; Validation, Y.W.; Writing—review and editing, Y.W.

Funding: This research received no external funding.

Acknowledgments: The authors are grateful to Xiang Ren, Keqin Zeng and Yin Zhan for their suggestions.

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

References

1. Fonseca, A.G.; Tortelli, O.L.; Lourenço, E.M. A reconfiguration analysis tool for distribution networks using fast decoupled power flow. In Proceedings of the 2015 IEEE PES Innovative Smart Grid Technologies Latin America (ISGT LATAM), Montevideo, Uruguay, 5–7 October 2015; pp. 182–187.
2. Hu, Y.; Lian, W.; Han, Y.; Dai, S.; Zhu, H. A seasonal model using optimized multi-layer neural networks to forecast power output of pv plants. *Energies* **2018**, *11*, 326. [[CrossRef](#)]
3. Liu, W.; Liu, C.; Lin, Y.; Ma, L.; Xiong, F.; Li, J. Ultra-short-term forecast of photovoltaic output power under fog and haze weather. *Energies* **2018**, *11*, 528. [[CrossRef](#)]
4. Sheng, H.; Xiao, J.; Cheng, Y.; Ni, Q.; Wang, S. Short-term solar power forecasting based on weighted gaussian process regression. *IEEE Trans. Ind. Electr.* **2018**, *65*, 300–308. [[CrossRef](#)]
5. Huang, C.Y.; Tzeng, W.C.; Liu, Y.W.; Wang, P.Y. Forecasting the global photovoltaic market by using the gm(1,1) grey forecasting method. In Proceedings of the 2011 IEEE Green Technologies Conference (IEEE-Green), Baton Rouge, LA, USA, 14–15 April 2011; pp. 1–5.
6. Qijun, S.; Fen, L.; Jialin, Q.; Jinbin, Z.; Zhenghong, C. Photovoltaic power prediction based on principal component analysis and support vector machine. In Proceedings of the 2016 IEEE Innovative Smart Grid Technologies—Asia (ISGT-Asia), Melbourne, Australia, 28 November–1 December 2016; pp. 815–820.
7. Jang, H.S.; Bae, K.Y.; Park, H.S.; Sung, D.K. Solar power prediction based on satellite images and support vector machine. *IEEE Trans. Sustain. Energy* **2016**, *7*, 1255–1263. [[CrossRef](#)]
8. Liu, Y.; Li, Z.; Bai, K.; Zhang, Z.; Lu, X.; Zhang, X. Short-term power-forecasting method of distributed pv power system for consideration of its effects on load forecasting. *J. Eng.* **2017**, *2017*, 865–869. [[CrossRef](#)]
9. Ojha, U.; Adhikari, U.; Singh, D.K. Image annotation using deep learning: A review. In Proceedings of the 2017 International Conference on Intelligent Computing and Control (I2C2), Coimbatore, India, 23–24 June 2017; pp. 1–5.
10. Neo, Y.Q.; Teo, T.T.; Woo, W.L.; Logenthiran, T.; Sharma, A. Forecasting of photovoltaic power using deep belief network. In Proceedings of the TENCON 2017—2017 IEEE Region 10 Conference, Penang, Malaysia, 5–8 November 2017; pp. 1189–1194.
11. Yacef, R.; Benhanem, M.; Mellit, A. Prediction of daily global solar irradiation data using bayesian neural network: A comparative study. *Renew. Energy* **2012**, *48*, 146–154. [[CrossRef](#)]
12. Cao, J.C.; Cao, S.H. Study of forecasting solar irradiance using neural networks with preprocessing sample data by wavelet analysis. *Energy* **2006**, *31*, 3435–3445. [[CrossRef](#)]
13. Qing, X.; Niu, Y. Hourly day-ahead solar irradiance prediction using weather forecasts by lstm. *Energy* **2018**, *148*, 461–468. [[CrossRef](#)]
14. Alzahrani, A.; Shamsi, P.; Ferdowsi, M.; Dagli, C. Solar irradiance forecasting using deep recurrent neural networks. In Proceedings of the 2017 IEEE 6th International Conference on Renewable Energy Research and Applications (ICRERA), San Diego, CA, USA, 5–8 November 2017; pp. 988–994.

15. Liu, B.; Fu, C.; Bielefeld, A.; Liu, Q.Y. Forecasting of chinese primary energy consumption in 2021 with gru artificial neural network. *Energies* **2017**, *10*, 1453. [[CrossRef](#)]
16. Ugurlu, U.; Oksuz, I.; Tas, O. Electricity price forecasting using recurrent neural networks. *Energies* **2018**, *11*, 1255. [[CrossRef](#)]
17. Hong, T.; Pinson, P.; Fan, S.; Zareipour, H.; Troccoli, A.; Hyndman, R.J. Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond. *Int. J. Forecast.* **2016**, *32*, 896–913. [[CrossRef](#)]
18. Wang, X.; Lee, W.J.; Huang, H.; Szabados, R.L.; Wang, D.Y.; Olinda, P.V. Factors that impact the accuracy of clustering-based load forecasting. *IEEE Trans. Ind. Appl.* **2016**, *52*, 3625–3630. [[CrossRef](#)]
19. Wu, W.; Peng, M. A Data Mining Approach Combining K-Means Clustering With Bagging Neural Network for Short-Term Wind Power Forecasting. *IEEE Internet Things J.* **2017**, *4*, 979–986. [[CrossRef](#)]
20. Xu, Q.; He, D.; Zhang, N.; Kang, C.; Xia, Q.; Bai, J.; Huang, J. A short-term wind power forecasting approach with adjustment of numerical weather prediction input by data mining. *IEEE Trans. Sustain. Energy* **2015**, *6*, 1283–1291. [[CrossRef](#)]
21. Agrawal, R.K.; Muchahary, F.; Tripathi, M.M. Long term load forecasting with hourly predictions based on long-short-term-memory networks. In Proceedings of the 2018 IEEE Texas Power and Energy Conference (TPEC), College Station, TX, USA, 8–9 February 2018; pp. 1–6.
22. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
23. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv* **2014**.
24. Jesus, O.D.; Hagan, M.T. Backpropagation algorithms for a broad class of dynamic networks. *IEEE Trans. Neural Netw.* **2007**, *18*, 14–27. [[CrossRef](#)] [[PubMed](#)]
25. Yazan, E.; Talu, M.F. Comparison of the stochastic gradient descent based optimization techniques. In Proceedings of the 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Turkey, 16–17 September 2017; pp. 1–5.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).