

Article

# Detection of Outliers in Time Series Power Data Based on Prediction Errors

Changzhi Li, Dandan Liu \*, Mao Wang, Hanlin Wang and Shuai Xu

College of Electronics and Information Engineering, Shanghai University of Electric Power, No. 185, Hucheng Ring Road, Pudong New Area District, Shanghai 201306, China

\* Correspondence: liudandan@shiep.edu.cn

**Abstract:** The primary focus of smart grid power analysis is on power load forecasting and data anomaly detection. Efficient and accurate power load prediction and data anomaly detection enable energy companies to develop reasonable production and scheduling plans and reduce waste. Since traditional anomaly detection algorithms are typically for symmetrically distributed time series data, the distribution of energy consumption data features uncertainty. To this end, a time series outlier detection approach based on prediction errors is proposed in this paper, which starts by using an attention mechanism-based convolutional neural network (CNN)-gated recursive unit (GRU) method to obtain the residual between the measured value and its predicted value, and the residual data generally conform to a symmetric distribution. Subsequently, for these residual data, a random forest classification algorithm based on grid search optimization is used to identify outliers in the power consumption data. The model proposed in this paper is applied to both classical and real energy consumption datasets, and the performance is evaluated using different metrics. As shown in the results, the average accuracy of the model is improved by 25.2% and the average precision is improved by 17.2%, with an average recall improvement of 16.4% and an average F1 score improvement of 26.8% compared to the mainstream algorithms.

**Keywords:** neural network; forecast error; outlier detection; electricity consumption data



**Citation:** Li, C.; Liu, D.; Wang, M.; Wang, H.; Xu, S. Detection of Outliers in Time Series Power Data Based on Prediction Errors. *Energies* **2023**, *16*, 582. <https://doi.org/10.3390/en16020582>

Academic Editor: Ali Elkamel

Received: 28 November 2022

Revised: 22 December 2022

Accepted: 30 December 2022

Published: 4 January 2023



**Copyright:** © 2023 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 (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Abnormality detection in energy consumption data helps to identify unusual conditions in grid data in time to alert staff who can then conduct an overhaul or maintenance of the grid to ensure continuous operation of the power system. Classic anomaly detection methods are mainly divided into the following categories. (1) Statistic-based outlier anomaly detection: employing statistical principles to identify data with a low probability distribution in the data set as outliers. (2) Clustering-based outlier anomaly detection: performing a cluster analysis on unlabeled data sets by means of grouping. Typically, groups of clusters can be regarded as normal data, and so relatively abnormal data are then compared with the clusters to determine outliers [1]. (3) Classification-based outlier anomaly detection: For data sets with class labels, a classifier is usually trained to distinguish between normal and abnormal data. This method often has specific requirements for samples [2]. (4) Proximity-based outlier anomaly detection: to define the “proximity” between data and then determine outliers based on this value. Typical approaches to proximity-based methods include density-based methods [3], which use neighborhood “density” to reflect proximity, and distance-based methods [3] using the global “distance” to reflect proximity. These approaches have all been widely employed in time series outlier detection; however, there is an issue with these classical methods: The method is effective when the data are symmetrically distributed and significantly correlated, but energy consumption data are not necessarily symmetrically distributed or significantly correlated. To solve such problems, researchers have proposed a combined, residual-based approach for

anomaly detection [4]. An energy consumption prediction model is built and the residual series is obtained by comparing the predicted results of this model to the original data. The anomaly detection algorithm is then applied to the residual series to detect outliers. Points with large differences can be considered anomalies.

A combination of regression methods (e.g., random forest) and adjusted box plot anomaly detection methods was used by Gustavo Felipe Martin Nascimento [5] to detect energy consumption data anomalies by applying them to actual measured values and the differences between measured values and their predicted ones. This approach shows promising potential for detecting such power consumption data quality issues. Tianyu Li [6] proposed a prediction-based anomaly scoring method for anomaly detection. Li used a long short-term memory (LSTM) network for prediction and assigned scores to each anomaly based on the distance between each detected anomaly in the test data and the nearest FPS pattern learned from the training data. Nur Shakirah Md Salleh [7] proposed a combination of regression method, least squares model, and mathematical solution instead of the commonly used classification method for anomaly detection, which is used for power load prediction and identifies the point of occurrence of anomalous events by threshold and prediction error values. Using the idea of prediction error compensation, Qixun Zhou [8] introduced a three-vector PTC method based on online prediction error compensation (TVPTC), which stores and updates the errors of all voltage vectors for reliable compensation to enhance parameter robustness. An adaptive control scheme incorporating uncertainty prediction error estimation was developed by Rusong Zhu and Ping Wang [9] for uncertain nonlinear systems with constrained input actuators (including amplitude saturation and rate saturation). This scheme enhances the estimation of uncertainty in the system and brings stability to the parameter estimation error unavailable with conventional adaptive control. Anil K. Madhusudhanan [10] proposed a fuel flow modeling approach for diesel and compressed gas engines based on prediction error identification and road data collection. Road vehicle data are collected during normal transportation operations and the results show that prediction error expansion (PEE) is typically the most effective mechanism for hiding reversible data using redundancy in the cover medium. A coordinated control strategy for complementary scenic energy storage that balances prediction error compensation and fluctuation suppression was proposed by Shuyan Zhang [11]. The statistical probability distribution is used to analyze the prediction error and power fluctuation, and the target region for compensating the prediction error and smoothing fluctuation is established. Based on the upper and lower limits of the allowable prediction error, the charging and discharging reference power of the target area is defined to compensate for the prediction error. The above research results demonstrate the feasibility of the prediction error method in targeting time series anomaly detection.

As residual-based anomaly detection methods require a high accuracy of prediction results, these methods are still not accurate enough and the generalizability of the model is weak regarding different energy consumption data for residual detection methods. Therefore, this study proposes a combined CNN-GRU-ATTENTION and GridSearchCV-RandomForest method for residuals, hereafter referred to as CGA-RF. In this scheme, the first step uses a CNN-GRU-ATTENTION based prediction model in which a CNN can fully extract the feature vectors from energy consumption data [12]. Dynamic changes are then fully modeled by the GRU [13,14], and an attention mechanism is employed to take a probabilistic view of resource allocation, enhancing the selection of important information [15–17]. High-precision prediction results are then obtained to compare the difference with the original data to obtain a new sequence of reconstructed residuals. In the second step, a random forest classification and detection algorithm based on grid search is used. The random forest algorithm is composed of multiple mutually independent decision tree classifiers [18], which is an integrated learning method based on a decision tree algorithm where the final classification result is determined by the vote of all decision tree algorithms. Here, the selection of classifier parameters is a particularly important step. In this study, a grid search approach was chosen for parameter optimization [19–21],

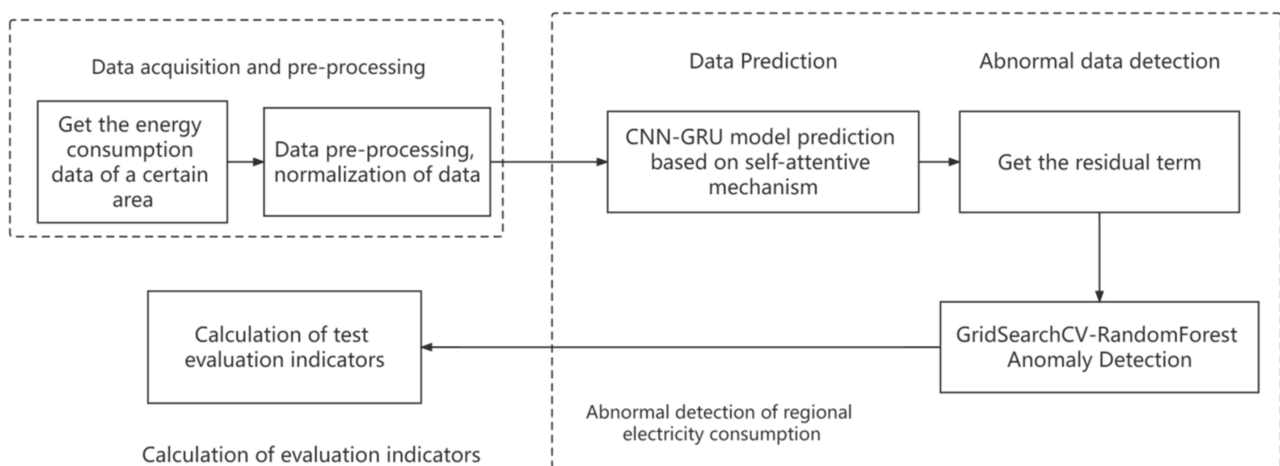
and then applied to the random forest classifier with optimized parameters for anomaly detection in residual sequences. The present research applies three sections of the datasets for validation; the first part is a real-time energy consumption series dataset used to validate the prediction ability of the CNN-GRU-ATTENTION model, the second part is a UCR classical time series classification dataset used to validate the anomaly detection ability of the GridSearchCV-RandomForest model, and the third part is a real data set containing outlier energy consumption data to verify the anomaly detection ability of the combined CGA-RF method chosen for analysis in the present study. Experimental results prove that this new, hybrid method has high detection accuracy of anomalies in energy consumption data.

The rest of the paper is presented as follows: in Section 2, the anomaly detection method based on prediction errors is described. In Section 3, the model is validated using classical and real datasets. In Section 4, the results are analyzed and discussed.

## 2. Anomaly Detection Method Based on Prediction Error

### 2.1. Methodology Overview

The CGA-RF method chosen in the present study is divided into two main parts. Firstly, a CNN-GRU model based on a self-attentive mechanism to effectively model the energy consumption data, and the predicted value of electricity consumption at the current moment as obtained from historical data. To generate a new sequence of residuals, the predicted and true values are subtracted to obtain the residuals. The new, reconstructed sequence of residuals is then classified using the random forest algorithm optimized by grid search to detect any outliers in the energy consumption data. The algorithm progression as shown in Figure 1 is described in the following steps.



**Figure 1.** CGA-RF abnormal data detection flow chart.

(1) Raw energy consumption data pre-processing. Obtain raw energy consumption data (time series data) to form data samples and perform normalized pre-processing on all data samples.

(2) Data prediction. The CNN-GRU model based on a self-recognizing mechanism is used to predict energy consumption data moment by moment and the predicted values of each instant are used to generate a new predicted series. Predicted values of corresponding moments are then subtracted from the true values and a new sequence of residuals is obtained.

(3) Anomaly detection. The random forest algorithm optimized by grid search is used to classify and detect a new sequence of residuals and detect outliers.

(4) Evaluation index calculation. The outliers detected by classification are labeled as anomalies and compared with the true labels of the data samples to calculate the corresponding evaluation metrics.

## 2.2. Predictive Models

### 2.2.1. CNN-GRU Prediction Model Based on the Attention Mechanism

In the study of load forecasting for electrical energy data, the electrical energy load sequence fluctuates due to random and nonlinear input. These influencing factors are diverse and complex (seasonal changes, temperature, humidity, weather, wind speed, holidays, etc.), which makes accurate forecasting a challenge. Neural networks feature self-learning and self-adaptive capabilities with the ability to deal with complexity and nonlinearity, as well as the capacity to adapt to complex and dynamic systems. These new systems can adequately solve the nonlinear problems that exist in large-scale load data and are therefore more widely used in the field of power load forecasting [22,23].

CNN is a commonly used deep learning algorithm that is widely utilized for text and image recognition. CNNs contain convolutional computation and maintain a deep structure of neural networks that often contain multiple hidden layers between a single input and output layer. These hidden layers are: convolutional, dense, max-pooling, dropout, and flatten [23]. The advantage of this structure is that it effectively reduces the number of structural weights, simplifies the network structure, and reduces complexity. The CNN model of convolutional pooling for historical data feature extraction saves computational time, improves computational efficiency, and models time series data well.

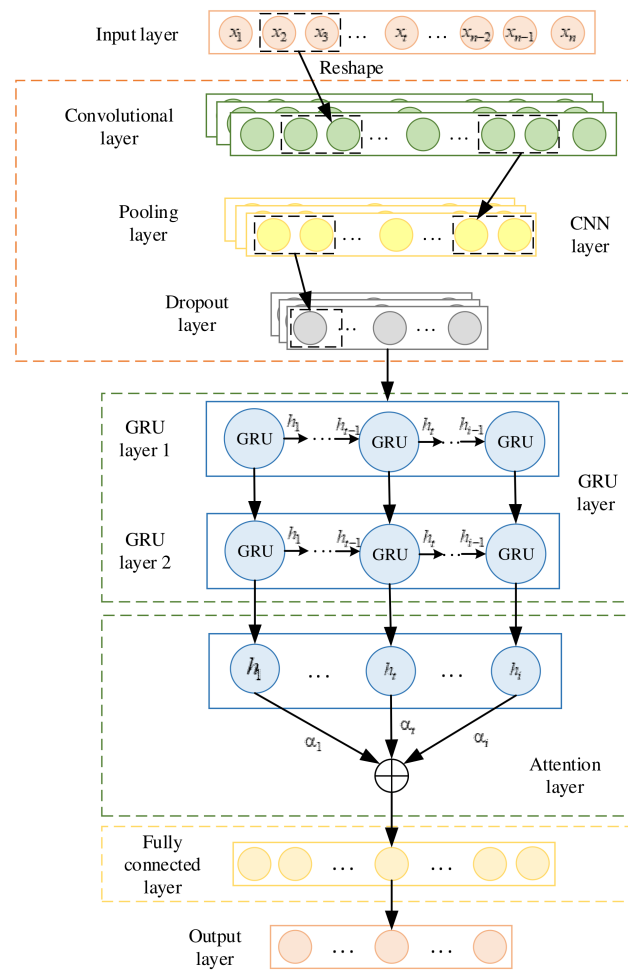
GRU is an updated version of the recurrent neural network (RNN). A GRU network can target the characteristics of load data uncertainty, effectively modeling dynamic time series data. The structure of the GRU has only two primary gates: update gate and reset gate [24]. The role of update gate is intended to regulate how much historical series information needs to be passed forward. It is useful for eliminating the risk arising from gradient descent by remembering historical information and deciding which messages are valid information and which are not. Reset gate is meant to forget some of the invalid information. The feature vector extracted from the CNN is input to the GRU to better learn the periodic change demand pattern in load data. However, the energy consumption data input is excessively long and the GRU network is prone to issues such as missing information and difficulties in modeling.

The attention mechanism is a resource allocation system that assigns different probability weights to the feature vectors input from the GRU layer, mainly to enhance the probability of important information to avoid loss. This, in turn, solves the problem of sequence loss that may occur in the model for load data due to long sequences, improving the model regarding important features in long sequence historical information. The CNN-GRU-ATTENTION model selected for this study uses the CNN model to extract effective feature vectors from load history data, which are then input to the GRU layer for effective modeling. The attention mechanism is employed to avoid the loss of effective information. By combining multiple structures for the effective processing of load data, the accuracy of load prediction is improved.

### 2.2.2. Model Structure

The structure of the CNN-GRU-ATTENTION prediction model proposed in the current study is shown in Figure 2; it is divided into the input layer, convolutional layer, pooling layer, dropout layer, GRU layer, attention layer, fully connected layer, and output layer.

The structure of the CNN-GRU model is based on the attention mechanism input layer. This layer is the beginning of the CNN model, without any weight input. In this stage, historical load data are input into the prediction model with the input vector represented by  $X$ .



**Figure 2.** Structure of CNN-GRU model based on attention mechanism.

**CNN layer:** For the current research, a CNN framework consisting of a convolutional layer, pooling layer and dropout layer was assembled. According to the time series, the convolutional layer is designed as a one-dimensional convolution with the ReLU used for activation. The convolution layer is input with data reshaped from the output of the input layer. The pooling layer uses the maximum pooling method, which is used to retain the maximum historical information of the load. Load data are then mapped to the hidden layer feature space after processing in the convolution layer and pooling layer, and then fed to the dropout layer. The role of the dropout layer is to randomly prevent the weights of certain hidden layer nodes of the network from working temporarily. Those nodes can be temporarily considered not part of the network structure, but their weights must be retained as they may have to function again in the next sample input. The output feature vector  $\mathbf{H}_C$  of the CNN layer can be expressed as:

$$C_1 = f(X \otimes W_1 + b_1) = \text{ReLU}(X \otimes W_1 + b_1) \quad (1)$$

$$P_1 = \max(C_1) + b_2 \quad (2)$$

$$H_C = f(P_1 \times W_2 + b_3) = \text{Sigmoid}(P_1 \times W_2 + b_3) \quad (3)$$

where  $\mathbf{C}_1$  is the output of the convolution layer;  $\mathbf{P}_1$  is the output of the pooling layer;  $\mathbf{W}_1$  is the weight matrix;  $\mathbf{b}_1$ ,  $\mathbf{b}_2$  and  $\mathbf{b}_3$  are the bias terms;  $\max()$  is the maximum function; and the output of the CNN layer is denoted as  $\mathbf{H}_C$ .

GRU layer: The main role of the GRU layer is to fully learn the sequence feature vector of the CNN layer input. The output of the GRU layer is denoted as **MISSING**, and the output at step  $t$  is denoted as  $H_C$ .

$$h_t = \text{GRU}(H_{C,t-1}, H_{C,t}), t \in [1, i] \quad (4)$$

Attention layer: The output vector of the GRU network layer is the input for the attention layer, and, according to the weight assignment of the self-attentive mechanism, the probabilities of different feature vectors are updated and iterated to calculate an improved weight parameter matrix.

$$e_t = u \tanh(wh_t + b) \quad (5)$$

$$\alpha_t = \frac{\exp(e_t)}{\sum_{j=1}^t e_j} \quad (6)$$

$$s_t = \sum_{i=1}^i \alpha_i h_t \quad (7)$$

where  $e_t$  denotes the value of the attention probability distribution determined by the GRU layer output vector  $h_t$  at moment  $t$ ;  $u$  and  $w$  are the weight coefficients;  $b$  is the bias term; and the output of the attention layer at moment  $t$  is  $s_t$ .

Output layer: The input of the output layer is the result generated by the attention mechanism layer. The mechanism of the output layer is the output of the computation through the fully connected layer. The prediction formula is expressed as follows:

$$y_t = \text{Sigmoid}(W_0 + b_0) \quad (8)$$

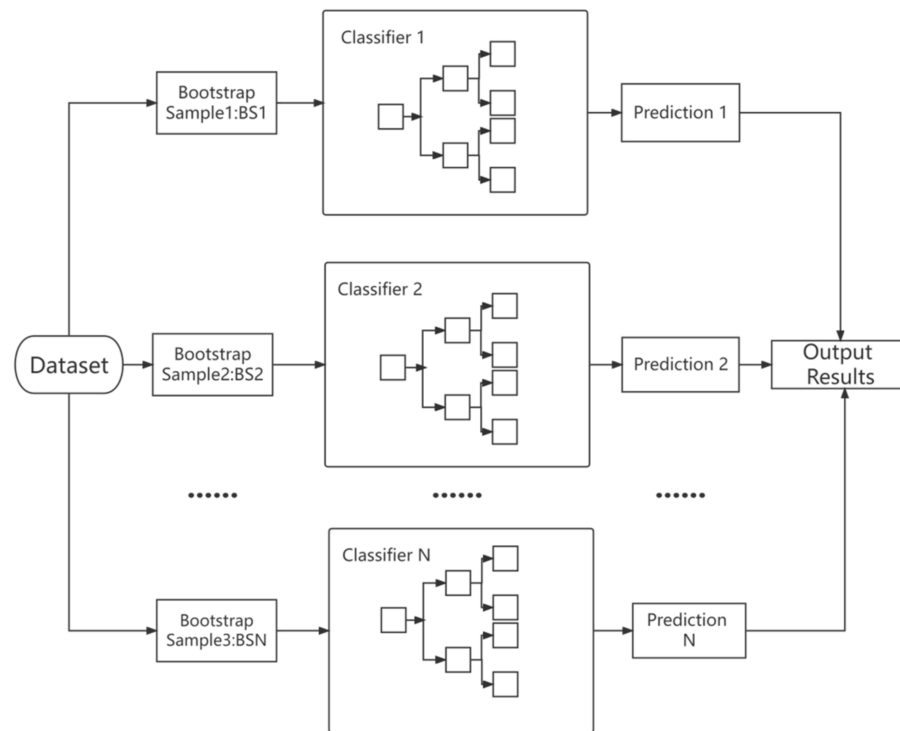
where  $y_t$  denotes the predicted output value at time  $t$ ;  $W_0$  is the weight matrix;  $b_0$  is the deviation terms; and the ReLU function is the activation function of the dense layer for the present research.

### 2.3. Detection Models

#### 2.3.1. Random Forest Classification Detection Model

The random forest algorithm is derived from integrated learning theory [25–31]. The model used in this study combines several independent classifiers. In order to simplify non-parameters and improve computational efficiency, classifiers are usually selected as regression trees to be used in the model algorithm [26]. Each classifier independently bootstraps the dataset randomly. The structure of the classification model is shown in Figure 3. For the classification detection problem, it is first assumed that the training data contain  $N$  observations, and in order to reduce errors associated with classification, an overlapping sampling solution called “bagging” is used in the model [27]. Specifically, the algorithm extracts observations by substitution, which, in turn, leads to the generation of independent bootstrap samples in the dataset. Each classifier is then trained from different bootstrap samples, thus increasing the diversity of the tree. To better reduce the correlation between various classifiers, the best splitting scheme for each node is obtained by randomly selecting a subset of  $M$  features instead of all  $M$  features. As a result, the classifiers within the model can continue to grow without pruning, which in turn reduces the computational burden. Moreover, by using different random samples and node features, the noise immunity of the model can be improved with the help of averaging various de-correlated classifiers [28,29]. In addition, for each classifier in the model, a bagging solution is adopted that utilizes a sampling method with put-back to generate training data. Through multiple rounds of random sampling of the initial training set with put-backs, multiple training sets are generated in parallel, corresponding to multiple base classifiers (without strong dependencies between base classifiers), and then these base classifications are combined to build a strong classifier. The essence of this is the introduction of sample perturbation, which reduces the variance by increasing sample

randomness. In this way, the model is able to achieve unbiased estimation without using external subsets of data [30,31].

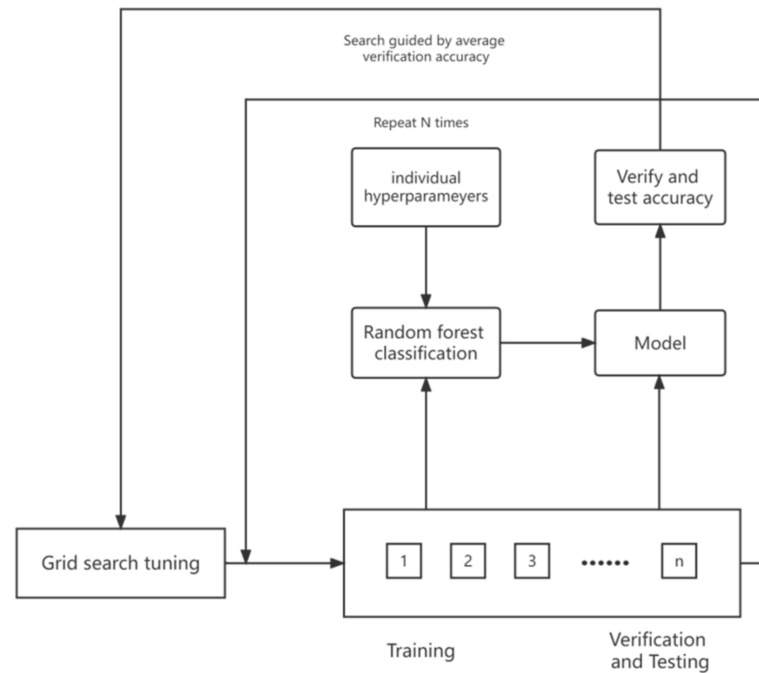


**Figure 3.** Random forest classification model.

### 2.3.2. Optimization of Grid Search Parameters

The selection of suitable parameters is the best way to achieve optimal detection. The fine tuning of parameters to obtain optimal model performance is called parameter tuning and a common method of tuning parameters is grid search [32,33], essentially an exhaustive method. The combination of model and hyper-parameters is chosen by exhaustively enumerating all parameter combinations required in the model and comparing, analyzing, and verifying each combination one by one. The purpose of the grid search is to identify the combinations that yield the best model performance, which can then be selected for use as a predictive model. A comparative analysis is performed to obtain an optimal set of hyper-parameters. To understand the origin of the name “grid search” [34], we first assume the existence of two hyper-parameters of the model, while each hyper-parameter has a set of candidate parameters and both sets of parameters are simultaneously parallel. These two sets of candidate parameters can be combined in pairs with all combinations classified as two-dimensional lattices (the case of multiple sets of hyperparameters combined in pairs can be considered lattices in higher dimensional spaces). The model then traverses all nodes in the lattice to select the optimal solution. Therefore, it is called grid search.

The grid search tuning process is shown in Figure 4, where the input data are first partitioned into a training set, a test set, and a validation set. The data set is divided into  $n$  parts to perform parameter tuning. Then, the parts of the random forest classification model that need parameter tuning are trained in  $(n - 2)$  parts for each alternative. A validation set is used to verify the performance of the model after parameter optimization and, finally, is used to test the model performance. The detection and validation accuracy of the tuned model are evaluated, and then the hyper-parameters are determined from the training set and tuning technique. The above steps are repeated  $n$  times and the average validation accuracy is taken as the fitness value. Finally, the highest test accuracy is determined.



**Figure 4.** Hyper-parameter tuning architecture.

## 2.4. Evaluation Metrics of Prediction and Detection Models

### 2.4.1. Evaluation Indices of Prediction Models

In order to evaluate the accuracy of the prediction model, the mean absolute error MAE, mean absolute percentage error MAPE, mean square error MSE, root mean square error RMSE, and coefficient of determination R2 (R-squared) were chosen for evaluation criteria, and they are expressed as:

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{y}_i - y_i| \quad (9)$$

$$MAPE = \frac{100\%}{n} \sum_{i=1}^n \left| \frac{\hat{y}_i - y_i}{y_i} \right| \quad (10)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2 \quad (11)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \quad (12)$$

$$R2 = 1 - \frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i (\bar{y}_i - y_i)^2} \quad (13)$$

where  $X$  is the total number of prediction results;  $y_i$  and  $\hat{y}_i$  are the actual load value and predicted load value of the  $i$ th sampling point of the prediction, respectively. MAE and MAPE can measure the superiority of the prediction model results, and RMSE and MSE can evaluate the accuracy of the prediction, which is sensitive to large or small errors in the results. R2 can describe the quality of the model. Generally, the larger the value of R2, the better the model fit, and the smaller the value of the above evaluation indicator, the more accurate the load prediction result. Smaller values of the above evaluation indices indicate more accurate load forecasting results.

### 2.4.2. Evaluation Index of Detection Model

Labeling load data as normal or abnormal points is essentially a classification problem. Several methods are used to evaluate the performance of models in solving such problems. In this paper, the concepts of accuracy, precision, recall, and F-score (or  $F_1$ ) are applied. These metrics are defined by the following equations:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (14)$$

$$Precision = \frac{TP}{TP + FP} \quad (15)$$

$$Recall = \frac{TP}{TP + FN} \quad (16)$$

$$F_1 = 2 \times \left( \frac{Precision \times Recall}{Precision + Recall} \right) \quad (17)$$

where TP is the number of true positives classified (outliers detected in practice), FP is the number of false positives classified (normal samples mistakenly detected as outliers), FN is the number of false negatives (no outliers detected), and TN is the number of true negatives for classification (error samples are not detected as outliers). In the outlier detection task, the accuracy rate indicates the proportion of points rated as normal by the model in all data. Recall rate indicates the proportion of normal data from all data that were attempted to be detected (resulting in normal data detected as normal and abnormal data detected as abnormal). F1 score is the summed average of the accuracy rate and recall rate.

## 3. Results and Analysis

### 3.1. Performance Analysis and Comparison of Load Consumption Data Prediction Models

#### 3.1.1. Comparison of Prediction Results for the Spanish Wind Power Dataset

In this study, a Spanish wind power dataset was used for training and prediction. The training and test subsets were divided in a ratio of 8:2. The parameters of the model are reported in Table 1.

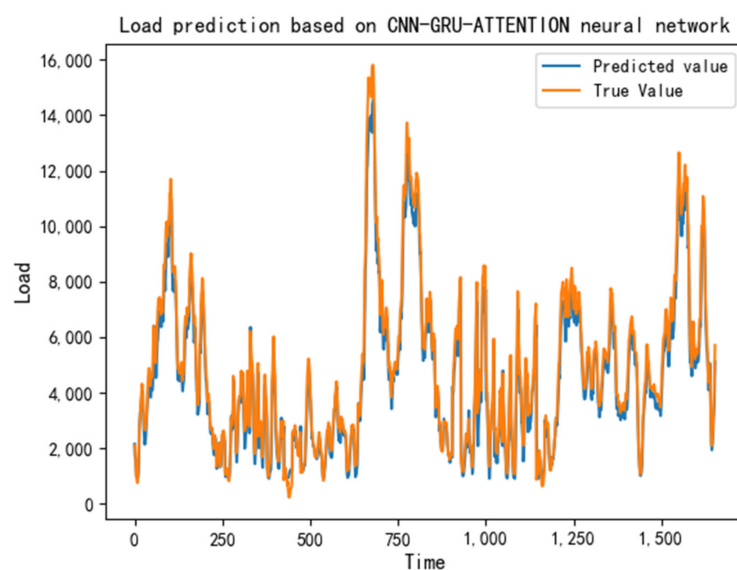
**Table 1.** Spanish Wind Power Dataset Model Parameters.

Parameters of the Model	Value
CNN layer	1
Pooling layer	1
Activation function	ReLU
Dropout	0.1
GRU layer	2
Learning Rate	0.01
ATTENTION	20
Dense	1
Activation function	ReLU
Epoch	30

In order to verify the validity and stability of the model, daily load predictions were performed on a test set covering one week per month over six months. The prediction results are shown in Table 2, and a comparison of the predicted and true values is shown in Figure 5.

**Table 2.** Comparison of forecast accuracy by month.

Date	Model	Evaluation Indicators				
		RMSE	MSE	MAPE	MAE	R <sup>2</sup>
7.7–7.13	CNN	1135.941	1,290,361.632	12.214	585.238	0.778
	GRU	1255.148	1,575,397.086	19.12	806.572	0.729
	CNN-GRU	1021.171	1,042,789.913	12.725	550.938	0.821
	CNN-GRU-ATTENTION	954.175	910,450.656	8.769	421.474	0.843
8.9–8.15	CNN	850.32	723,044.915	19.473	670.68	0.861
	GRU	1046.284	1,094,710.078	25.731	827.541	0.789
	CNN-GRU	528.219	279,015.322	12.111	385.355	0.946
	CNN-GRU-ATTENTION	381.397	145,463.416	6.923	225.526	0.972
9.3–9.10	CNN	899.017	808,231.025	18.751	582.62	0.81
	GRU	910.147	828,368.466	18.7	579.017	0.806
	CNN-GRU	813.783	662,242.698	17.11	462.629	0.845
	CNN-GRU-ATTENTION	715.628	512,124.114	11.532	343.945	0.88
10.10–10.6	CNN	1136.644	1,291,958.965	21.276	953.556	0.814
	GRU	1367.021	1868745.27	27.603	1211.353	0.731
	CNN-GRU	733.993	538,745.206	15.761	590.726	0.922
	CNN-GRU-ATTENTION	271.683	73,811.518	4.718	210.27	0.989
11.11–11.17	CNN	552.078	304,790.535	17.406	381.762	0.83
	GRU	723.403	523,311.337	26.181	586.918	0.708
	CNN-GRU	568.095	322,731.953	17.555	391.809	0.82
	CNN-GRU-ATTENTION	444.987	198,013.861	12.606	261.476	0.89
12.20–12.26	CNN	704.854	496,819.766	15.71	565.275	0.896
	GRU	939.185	882,067.55	19.323	825.093	0.816
	CNN-GRU	545.412	297,474.436	13.008	447.483	0.938
	CNN-GRU-ATTENTION	299.295	89,577.487	6.884	239.814	0.981

**Figure 5.** Load prediction based on CNN-GRU-ATTENTION neural network.

From each error indicator, it can be seen that the prediction results are accurate to each day, the value of MAPE is relatively low, the prediction error is small, and the prediction accuracy is relatively high. From the analysis of the prediction results for a single day compared with the other three methods, the method used in this study (CNN-GRU-ATTENTION) has the highest prediction accuracy, and the MAPE values are reduced by 3.445%, 10.351%, 3.956%; 1.984%, 2.072%, 0.939%; 1.589%, 1.852%, 1.122%; 0.61%, 0.786%, 0.185%; 1.32%, 0.7%, 0.27%; 1.308%, 1.652%, 0.564%; 2.597%, 1.361%, 0.993%; 1.908%, 2.906%, 2.26%. In the comprehensive analysis, the study's hybrid method has significantly reduced the MSE, RMSE, MAE, and MAPE indices and generated a significant increase in the R2 index, which indicates that the overall prediction accuracy and model performance have been greatly improved for the prediction process.

### 3.1.2. Comparison of Forecast Results for the Australian Electricity Price Dataset

For the purpose of this study, the Australian load output dataset was used for training and prediction. As before, the training and test sets were divided in a ratio of 8:2. The parameters of the model are reported in Table 3.

**Table 3.** Australian Load Output Dataset Model Parameters.

Parameters of the Model	Value
CNN layer	1
Pooling layer	1
Activation function	ReLU
Dropout	0.2
GRU layer	2
Learning Rate	0.01
ATTENTION	50
Dense	1
Activation function	ReLU
Epoch	30

In order to verify the validity and stability of the model, daily load predictions were performed in a test set covering one week per month. Prediction results are shown in Table 4, and a comparison of predicted and true values is shown in Figure 6.

**Table 4.** Comparison of prediction accuracy for consecutive weeks.

Date	Model	Evaluation Indicators				
		RMSE	MSE	MAPE	MAE	R <sup>2</sup>
11.5	CNN	242.513	58,812.314	3.032	197.926	0.868
	GRU	249.603	62,301.758	3.12	204.326	0.861
	CNN-GRU	161.28	26,011.357	1.987	128.545	0.942
	CNN-GRU-ATTENTION	87.472	7651.373	1.048	69.202	0.983
11.6	CNN	224.038	50,192.963	2.661	171.889	0.9
	GRU	238.851	57,049.804	2.924	192.601	0.887
	CNN-GRU	191.55	36,691.222	2.194	142.646	0.927
	CNN-GRU-ATTENTION	87.584	7670.973	1.072	71.059	0.985

Table 4. Cont.

Date	Model	Evaluation Indicators				
		RMSE	MSE	MAPE	MAE	R <sup>2</sup>
11.7	CNN	204.26	41,722.352	2.371	161.488	0.912
	GRU	217.894	47,477.897	2.547	174.975	0.899
	CNN-GRU	163.287	26,662.551	1.946	133.39	0.944
	CNN-GRU-ATTENTION	135.814	18,445.367	1.761	118.073	0.961
11.8	CNN	226.874	51,472.017	2.718	194.502	0.913
	GRU	184.299	33,966.042	2.098	147.527	0.943
	CNN-GRU	154.429	23,848.364	1.668	115.811	0.96
	CNN-GRU-ATTENTION	114.103	13,019.567	1.398	96.397	0.978
11.9	CNN	252.1	63,554.307	2.836	196.709	0.902
	GRU	254.582	64,811.78	3.18	224.129	0.9
	CNN-GRU	191.682	36,741.817	2.092	145.075	0.943
	CNN-GRU-ATTENTION	144.512	20,883.604	1.528	108.124	0.968
11.10.	CNN	307.486	94,547.492	3.843	260.032	0.871
	GRU	220.718	48,716.269	2.607	181.417	0.934
	CNN-GRU	200.243	40,097.387	2.239	152.155	0.945
	CNN-GRU-ATTENTION	110	12,099.979	1.246	86.342	0.983
11.11	CNN	274.768	75,497.405	3.285	227.867	0.9
	GRU	383.623	147,166.845	4.283	300.485	0.804
	CNN-GRU	311.187	96,837.065	3.637	251.12	0.871
	CNN-GRU-ATTENTION	127.859	16,348.029	1.377	96.911	0.978

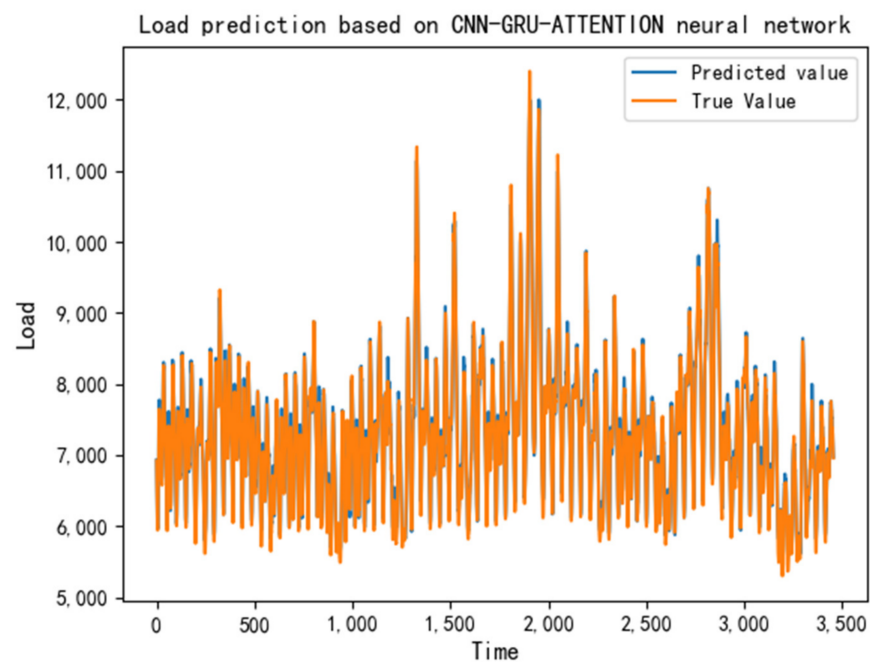


Figure 6. Load prediction based on CNN-GRU-ATTENTION neural network.

From each error indicator, it can be seen that the prediction results are accurate to each day, the value of MAPE is relatively low, the prediction error is small, and the prediction accuracy is relatively high (1.122%; 0.61%, 0.786%, 0.185%; 1.32%, 0.7%, 0.27%; 1.308%, 1.652%, 0.564%; 2.597%, 1.361%, 0.993%; 1.908%, 2.906%, 2.26%). In a comprehensive analysis, the methods used in this study have significantly reduced MSE, RMSE, MAE, and MAPE indicators, leading to a significant increase in R2 indicators, which shows a significant improvement in the overall prediction accuracy and model performance of the prediction process.

### 3.2. Performance Analysis and Comparison of Outlier Point Detection Models

This study conducted experiments using the UCR time series classification dataset, which consists of publicly available time series datasets that vary according to the number of samples and the length of the time series. A standard partition is used to divide each dataset into a training set and a test set. For this research, a typical time series classification dataset with different time series lengths for different numbers of classifications was extracted. From the results in the table below, it is shown that the GridSearchCV-RandomForest algorithm selected for this study has significantly improved accuracy, precision, recall, and F1 score compared to other machine learning algorithms.

From Tables 5–9, it is evident that the grid search optimized random forest algorithm utilized in this study is superior for detection than traditional integrated classification algorithms. From the FreezerRegularTrain dataset, it is evident that compared to RandomForest, DecisionTree [35–37], and AdaBoost [38] algorithms, GridSearchCV-RandomForest offers significant improvements in accuracy, precision, recall, and F1 score; in the PowerCons dataset, the GridSearchCV-RandomForest algorithm showed 100% accuracy in precision, recall, and F1 score. In the Wafer dataset, the GridSearchCV-RandomForest algorithm detected 620 true anomalies with 2.05% and 23.28% improvements in accuracy and recall, respectively, compared to the RandomForest algorithm. In the Italian electricity demand dataset, the GridSearchCV-RandomForest algorithm also achieved an accuracy and recall score of 0.970. Similarly, the GridSearchCV-RandomForest algorithm showed high detection capability in the Mote strain dataset.

**Table 5.** FreezerRegularTrain dataset.

	True Positives	False Negatives	False Positives	Accuracy	Precision	Recall	F1 Score
DecisionTree	1261	164	157	0.887	0.889	0.884	0.887
AdaBoost	1370	55	99	0.947	0.932	0.961	0.946
RandomForest	1402	23	117	0.841	0.952	0.950	0.950
GridSearchCV-RandomForest	1402	23	120	0.949	0.951	0.949	0.949

**Table 6.** PowerCons dataset.

	True Positives	False Negatives	False Positives	Accuracy	Precision	Recall	F1 Score
DecisionTree	90	0	2	0.988	0.978	1.0	0.989
AdaBoost	90	0	0	1.0	1.0	1.0	1.0
RandomForest	90	0	0	1.0	1.0	1.0	1.0
GridSearchCV-RandomForest	90	0	0	1.0	1.0	1.0	1.0

**Table 7.** Wafer dataset.

	True Positives	False Negatives	False Positives	Accuracy	Precision	Recall	F1 Score
DecisionTree	397	268	121	0.936	0.766	0.596	0.671
AdaBoost	605	60	60	0.980	0.909	0.909	0.909
RandomForest	503	162	8	0.972	0.984	0.756	0.855
GridSearchCV-RandomForest	620	45	12	0.992	0.981	0.932	0.956

**Table 8.** Italy power demand dataset.

	True Positives	False Negatives	False Positives	Accuracy	Precision	Recall	F1 Score
DecisionTree	489	24	15	0.962	0.970	0.953	0.961
AdaBoost	465	48	18	0.935	0.962	0.906	0.933
RandomForest	497	16	16	0.968	0.968	0.968	0.968
GridSearchCV-RandomForest	498	15	15	0.934	0.970	0.970	0.970

**Table 9.** Mote strain dataset.

	True Positives	False Negatives	False Positives	Accuracy	Precision	Recall	F1 Score
DecisionTree	552	123	144	0.786	0.793	0.817	0.805
AdaBoost	570	105	147	0.798	0.794	0.844	0.818
RandomForest	603	72	71	0.835	0.894	0.893	0.893
GridSearchCV-RandomForest	612	63	88	0.879	0.874	0.906	0.890

### 3.3. Validation of Outlier Detection Algorithm for Energy Consumption Data Based on the Prediction Error of Real Data Sets

The real data were obtained from actual energy consumption data recorded in a city in the Zhejiang Province of China for one year in 2020, collected every fifteen minutes. A partial screenshot of the dataset is shown in Figure 7. To measure the effectiveness of the electricity data anomaly detection algorithm, these data were manually labeled, meaning that any anomalies have been identified. These labels were used only when evaluating the strengths and weaknesses of the algorithm and were not used in model training. The parameters of the model are reported in Table 10.

**Table 10.** Real Dataset Model Parameters.

Parameters of the Model	Value
CNN layer	1
Pooling layer	1
Activation function	ReLU
Dropout	0.2
GRU layer	2
Learning Rate	0.01
ATTENTION	20
Dense	1
Activation function	ReLU
Epoch	30

date	active power(kW)	Reactive power(kvar)	A-phase current(A)	B-phase current(A)	C-phase current(A)	A-phase voltage(V)	B-phase voltage(V)	B-phase voltage(V)	Normal or abnormal	Inbound time
2020-01-01	185.28	12	232	256	296	240	240	240		1,2020-01-01 00:02:05
2020-01-01	159.2	12	220	240	212	240	240	240		1,2020-01-01 00:17:08
2020-01-01	136.6	8	216	180	180	240	241	240		1,2020-01-01 00:32:07
2020-01-01	121.48	12	172	172	172	241	241	241		1,2020-01-01 00:47:10
2020-01-01	99.96	12	140	148	140	241	241	241	0.99	2020-01-01 01:02:09
2020-01-01	112.92	12	152	168	156	241	241	241	0.99	2020-01-01 01:17:09
2020-01-01	107.8	8	136	144	176	241	242	241		1,2020-01-01 01:32:08
2020-01-01	96.36	8	100	160	152	242	242	242		1,2020-01-01 01:47:06
2020-01-01	93.32	12	76	172	152	242	242	242	0.99	2020-01-01 02:02:08
2020-01-01	94.92	8	100	168	136	242	242	242		1,2020-01-01 02:25:37
2020-01-01	111.24	8	164	164	144	241	242	242		1,2020-01-01 02:32:09
2020-01-01	81.56	8	84	136	128	242	242	242		1,2020-01-01 02:48:02
2020-01-01	87.56	8	96	132	144	242	242	242		1,2020-01-01 03:02:09
2020-01-01	85.48	8	116	120	128	242	242	242		1,2020-01-01 03:17:09
2020-01-01	71.68	8	84	96	128	242	242	242	0.99	2020-01-01 03:32:06
2020-01-01	77.2	8	100	120	108	242	242	243	0.99	2020-01-01 03:47:08
2020-01-01	70.96	8	108	88	108	242	242	242	0.99	2020-01-01 04:02:07
2020-01-01	72.6	8	88	108	116	242	242	243	0.99	2020-01-01 04:17:06
2020-01-01	69.8	8	72	120	104	242	242	242	0.99	2020-01-01 04:32:09
2020-01-01	80	8	112	112	116	242	242	243		1,2020-01-01 04:47:06
2020-01-01	68.76	8	72	108	112	242	242	242	0.99	2020-01-01 05:02:09
2020-01-01	68.68	8	76	128	92	242	242	242	0.99	2020-01-01 05:17:09
2020-01-01	73.04	8	96	120	96	242	242	243	0.99	2020-01-01 05:32:09
2020-01-01	77.2	12	112	132	84	242	242	243	0.99	2020-01-01 05:47:09
2020-01-01	66.84	12	76	112	100	242	242	242	0.98	2020-01-01 06:02:06
2020-01-01	80.04	4	84	144	112	242	242	242		1,2020-01-01 06:17:09
2020-01-01	99.56	8	132	144	144	242	242	242		1,2020-01-01 06:32:08
2020-01-01	100.36	8	164	144	116	241	241	241		1,2020-01-01 06:47:09
2020-01-01	95.4	8	120	120	160	241	241	241		1,2020-01-01 07:02:18
2020-01-01	109.44	8	140	144	180	240	240	240		1,2020-01-01 07:26:41

Figure 7. Partial screenshots of real data sets.

### 3.3.1. Real Data Prediction

Figure 8 shows the prediction results compared to the original data. It is clear that the prediction model forecasts items very effectively for real data. The quantitative results are shown in Table 9. Imperfections in prediction results are to be expected as there are outliers in the original data, which are to be identified. The new sequence of residuals reconstructed after prediction, shown in Figure 9, is used for the next step of outlier detection.

Table 11 shows the experimental results of comparing the CNN-GRU-ATTENTION prediction model used in this study with several other classical prediction models. In terms of RMSE, the method here is 6.056, 30.089, and 4.536 lower than for CNN, GRU, and CNN-GRU, respectively; MSE is 510.064, 2191.232, and 442.583 lower; MPAAE is 4.545%, 14.349%, and 2.214% lower; MAE is 5.55, 25.812, 3.361 lower; and R2 improved by 0.056, 0.162, 0.04. The method of this research has significantly improved five prediction evaluation indices compared with the other three methods, and the prediction accuracy is relatively high. Overall, it seems that the prediction method chosen for study has the best prediction performance.

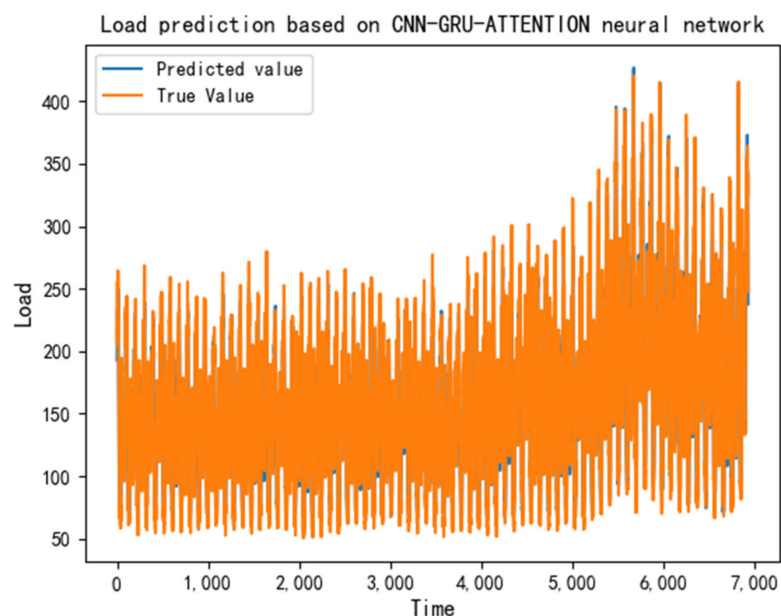
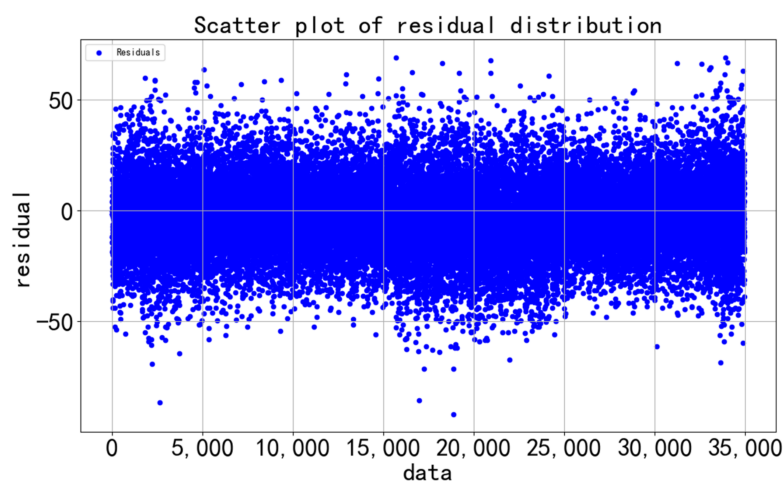


Figure 8. Real data predicted value vs. real value.



**Figure 9.** Residual distribution chart.

**Table 11.** Real dataset prediction results.

	RMSE	MSE	MAPE	MAE	R2
CNN	22.959	527.094	13.090	18.354	0.879
GRU	46.992	2208.262	22.894	38.616	0.773
CNN-GRU	21.439	459.613	10.759	16.165	0.895
CNN-GRU-ATTENTION	16.903	17.030	8.545	12.804	0.935

### 3.3.2. Outlier Detection

The CGA-RF method selected for anomaly detection in this study was compared with the RandomForest classification algorithm, DecisionTree classification algorithm, and AdaBoost classification algorithm. The anomaly detection methods for comparison used normalized energy consumption data as input, while the method in this study first models the temporal characteristics of electricity consumption data using a CNN-GRU model based on a self-focus mechanism and performs prediction to obtain a prediction sequence. Then, the predicted and true values are subtracted to obtain a new sequence of residuals, which is classified using a random forest classification algorithm optimized by grid search to identify anomalies. In this study, accuracy, precision, recall, and F1 score were selected as evaluation criteria for the detection effectiveness of all methods.

Detection results are shown in Table 12. We applied DecisionTree, AdaBoost, RandomForest, and GridSearchCV-RandomForest to detect outliers in the load data. It is clear that the random forest algorithm optimized by grid search shows better overall detection advantages compared with other algorithms in this research, detecting 1624 outliers with 95.9% accuracy, 96.4% precision, 89.3% recall, and an F1 score of 0.933. Among them, the decision tree algorithm, although performing relatively well in the number of detected anomalies, detected 369 fewer anomalies than the research method in this paper. Although AdaBoost detected more anomalies, it misclassified 3813 points as normal and the detection performance was poor. The random forest algorithm without grid search parameter optimization, however, can be seen to have lower detection ability than the random forest algorithm with grid search parameter optimization in several evaluation metrics. In addition, the method in this study achieves the highest recall rate, which means that relatively more outlier points are found. On the one hand, this is due to the fact that the model uses CNN-GRU based on a self-attentive mechanism to model the temporal correlation of power data, which better utilizes historical information to predict data at the current moment and thus reconstructs the load sequence, making it easier to distinguish normal data from abnormal data in the new sequence of residuals. Finally, the results of the GridSearchCV-RandomForest algorithm and other comparative models show the advantage of GridSearchCV-RandomForest in detecting anomalies with asymmetric distribution.

**Table 12.** Real dataset detection results.

	True Positives	False Negatives	False Positives	Accuracy	Precision	Recall	F1 Score
DecisionTree	1254	562	0	0.900	0.936	0.845	0.874
AdaBoost	1801	15	3813	0.324	0.513	0.500	0.251
RandomForest	1254	562	25	0.896	0.926	0.842	0.869
GridSearchCV-RandomForest	1623	193	39	0.959	0.964	0.893	0.933

#### 4. Conclusions

In this study, a combined CNN-GRU prediction model was proposed based on a self-attentive mechanism and a random forest detection model based on grid search optimization (CGA-RF) for targeted detection of anomalies in time series energy consumption data. The central points of the full study are summarized as follows.

(1) Combining the CNN-GRU-ATTENTION model with the GridSearchCV-RandomForest model, the CNN-GRU-ATTENTION model makes full use of a CNN to extract the feature vectors of energy consumption data, and then of the GRU to accurately model its dynamic changes. The attention mechanism layer then probabilistically assigns important resources, enhances the selection of important information, and makes full use of historical information to predict power consumption at any given moment. The use of integrated learning methods such as random forest for anomaly detection of residual terms of predicted and true values can effectively improve the accuracy of detection, while the grid search parameter optimization method can reduce the manual tuning time for parameters and effectively improve the speed and efficiency of anomaly detection.

(2) Energy consumption data essentially satisfy time series distribution in terms of time series characteristics such as trend, periodicity, and seasonality. An empirical analysis of the selected combined method (CGA-RF) for anomaly detection from actual energy consumption data in this paper verifies the effectiveness of the method for anomaly detection of electricity consumption data.

(3) The current combination of the CNN-GRU model based on the self-attention mechanism and the GridSearchCV-RandomForest algorithm can only detect anomalies at a single point in time in electricity consumption data. In future studies, a method to detect and identify anomalous time periods can be considered. Meanwhile, since the method selected in this study mainly focuses on high accuracy prediction, subsequent research can be performed to further develop the accuracy of the prediction model.

**Author Contributions:** Conceptualization, C.L., D.L. and S.X.; Methodology, C.L., D.L. and H.W.; Investigation, M.W.; Writing—original draft, C.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The data presented in this study are available on request from the first author.

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

#### References

1. Lan, T.; Lin, Y.; Wang, J.; Leao, B.; Fradkin, D. Unsupervised Power System Event Detection and Classification Using Unlabeled PMU Data. In Proceedings of the 2021 IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe), Espoo, Finland, 18–21 October 2021; pp. 1–5. [\[CrossRef\]](#)
2. Rao, S.; Muniraju, G.; Tepedelenlioglu, C.; Srinivasan, D.; Tamizhmani, G.; Spanias, A. Dropout and Pruned Neural Networks for Fault Classification in Photovoltaic Arrays. *IEEE Access* **2021**, *9*, 120034–120042. [\[CrossRef\]](#)
3. Mandhare, H.C.; Idate, S.R. A comparative study of cluster based outlier detection, distance based outlier detection and density based outlier detection techniques. In Proceedings of the 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 15–16 June 2017; pp. 931–935. [\[CrossRef\]](#)

4. Wang, H.; Bah, M.J.; Hammad, M. Progress in Outlier Detection Techniques: A Survey. *IEEE Access* **2019**, *7*, 107964–108000. [[CrossRef](#)]
5. Nascimento, G.F.M.; Wurtz, F.; Kuo-Peng, P.; Delinchant, B.; Batistela, N.J. Outlier Detection in Buildings' Power Consumption Data Using Forecast Error. *Energies* **2021**, *14*, 8325. [[CrossRef](#)]
6. Li, T.; Comer, M.L.; Delp, E.J.; Desai, S.R.; Mathieson, J.L.; Foster, R.H.; Chan, M.W. Anomaly Scoring for Prediction-Based Anomaly Detection in Time Series. In Proceedings of the 2020 IEEE Aerospace Conference, Big Sky, MT, USA, 7–14 March 2020; pp. 1–7. [[CrossRef](#)]
7. Salleh, N.S.M.; Saripuddin, M.; Suliman, A.; Jorgensen, B.N. Electricity Anomaly Point Detection using Unsupervised Technique Based on Electricity Load Prediction Derived from Long Short-Term Memory. In Proceedings of the 2021 2nd International Conference on Artificial Intelligence and Data Sciences (AiDAS), Ipoh, Malaysia, 8–9 September 2021; pp. 1–5. [[CrossRef](#)]
8. Zhou, Q.; Liu, F.; Gong, H. Robust three-vector model predictive torque and stator flux control for PMSM drives with prediction error compensation. *J. Power Electron.* **2022**, *22*, 1917–1926. [[CrossRef](#)]
9. Zhu, R.; Wang, P. Adaptive Control of Nonlinear System Under Input Constraints Combined with Prediction-Error Estimation for Uncertainty. In Proceedings of the 2022 IEEE 17th International Conference on Control & Automation (ICCA), Naples, Italy, 27–30 June 2022; pp. 63–67. [[CrossRef](#)]
10. Madhusudhanan, A.K.; Na, X.; Ainalis, D.; Cebon, D. Engine Fuel Consumption Modelling Using Prediction Error Identification and On-Road Data. Available online: <http://eprints.soton.ac.uk/id/eprint/457356> (accessed on 29 December 2022).
11. Zhang, S.; Zhang, G.; Zhang, K. Coordinated Control Strategy of Wind-Photovoltaic Hybrid Energy Storage Considering Prediction Error Compensation and Fluctuation Suppression. In Proceedings of the 2021 IEEE 2nd International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA), Chongqing, China, 17–19 December 2021; pp. 1185–1189. [[CrossRef](#)]
12. Peñaloza, A.K.A.; Balbinot, A.; Leborgne, R.C. "Review of Deep Learning Application for Short-Term Household Load Forecasting. In Proceedings of the 2020 IEEE PES Transmission & Distribution Conference and Exhibition—Latin America (T&D LA), Montevideo, Uruguay, 28 September–2 October 2020; pp. 1–6. [[CrossRef](#)]
13. Shahi, T.B.; Shrestha, A.; Neupane, A.; Guo, W. Stock Price Forecasting with Deep Learning: A Comparative Study. *Mathematics* **2020**, *8*, 1441. [[CrossRef](#)]
14. Jung, S.; Moon, J.; Park, S.; Hwang, E. An Attention-Based Multilayer GRU Model for Multistep-Ahead Short-Term Load Forecasting. *Sensors* **2021**, *21*, 1639. [[CrossRef](#)]
15. Meng, Z.; Xie, Y.; Sun, J. Short-term load forecasting using neural attention model based on EMD. *Electr. Eng.* **2022**, *104*, 1857–1866. [[CrossRef](#)]
16. Park, J.; Hwang, E. A Two-Stage Multistep-Ahead Electricity Load Forecasting Scheme Based on LightGBM and Attention-BiLSTM. *Sensors* **2021**, *21*, 7697. [[CrossRef](#)]
17. Lin, T.; Pan, Y.; Xue, G.; Song, J.; Qi, C. A Novel Hybrid Spatial-Temporal Attention-LSTM Model for Heat Load Prediction. *IEEE Access* **2020**, *8*, 159182–159195. [[CrossRef](#)]
18. Xia, X.; Togneri, R.; Sohel, F.; Huang, D. Random forest classification based acoustic event detection. In Proceedings of the 2017 IEEE International Conference on Multimedia and Expo (ICME), Munich, Germany, 16 November 2017; pp. 163–168. [[CrossRef](#)]
19. Nagaraj, P.; Muneeswaran, V.; Deshik, G. Ensemble Machine Learning (Grid Search & Random Forest) based Enhanced Medical Expert Recommendation System for Diabetes Mellitus Prediction. In Proceedings of the 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 17–19 August 2022; pp. 757–765. [[CrossRef](#)]
20. Siji George, C.G.; Sumathi, B. Grid Search Tuning of Hyperparameters in Random Forest Classifier for Customer Feedback Sentiment Prediction. *Int. J. Adv. Comput. Sci. Appl. IJACSA* **2020**, *11*, 173–178.
21. Abokhzam, A.A.; Gupta, N.K.; Bose, D.K. Efficient diabetes mellitus prediction with grid based random forest classifier in association with natural language processing. *Int. J. Speech Technol.* **2021**, *24*, 601–614. [[CrossRef](#)]
22. Shi, H.; Wang, L.; Scherer, R.; Wozniak, M.; Zhang, P.; Wei, W. Short-Term Load Forecasting Based on Adabelief Optimized Temporal Convolutional Network and Gated Recurrent Unit Hybrid Neural Network. *IEEE Access* **2021**, *9*, 66965–66981. [[CrossRef](#)]
23. Pavićević, M.; Popović, T. Forecasting Day-Ahead Electricity Metrics with Artificial Neural Networks. *Sensors* **2022**, *22*, 1051. [[CrossRef](#)]
24. Ayub, N.; Irfan, M.; Awais, M.; Ali, U.; Ali, T.; Hamdi, M.; Alghamdi, A.; Muhammad, F. Big Data Analytics for Short and Medium-Term Electricity Load Forecasting Using an AI Techniques Ensembler. *Energies* **2020**, *13*, 5193. [[CrossRef](#)]
25. Liu, K.; Hu, X.; Zhou, H.; Tong, L.; Widanalage, D.; Marco, J. Feature Analyses and Modelling of Lithium-ion Batteries Manufacturing based on Random Forest Classification. *IEEE/ASME Trans. Mechatron.* **2021**, *26*, 2944–2955. [[CrossRef](#)]
26. Sales, M.H.R.; de Bruin, S.; Souza, C.; Herold, M. Land Use and Land Cover Area Estimates from Class Membership Probability of a Random Forest Classification. *IEEE Trans. Geosci. Remote. Sens.* **2022**, *60*, 4402711. [[CrossRef](#)]
27. Zhang, L.; Liu, K.; Wang, Y.; Omariba, Z.B. Ice Detection Model of Wind Turbine Blades Based on Random Forest Classifier. *Energies* **2018**, *11*, 2548. [[CrossRef](#)]
28. Xiong, F.; Cao, C.; Tang, M.; Wang, Z.; Tang, J.; Yi, J. Fault Detection of UHV Converter Valve Based on Optimized Cost-Sensitive Extreme Random Forest. *Energies* **2022**, *15*, 8059. [[CrossRef](#)]
29. Sun, Y.; Que, H.; Cai, Q.; Zhao, J.; Li, J.; Kong, Z.; Wang, S. Borderline SMOTE Algorithm and Feature Selection-Based Network Anomalies Detection Strategy. *Energies* **2022**, *15*, 4751. [[CrossRef](#)]
30. Dudek, G. A Comprehensive Study of Random Forest for Short-Term Load Forecasting. *Energies* **2022**, *15*, 7547. [[CrossRef](#)]
31. Lu, Y.; Li, Y.; Xie, D.; Wei, E.; Bao, X.; Chen, H.; Zhong, X. The Application of Improved Random Forest Algorithm on the Prediction of Electric Vehicle Charging Load. *Energies* **2018**, *11*, 3207. [[CrossRef](#)]

32. Chi, Y.; Zhang, Y.; Li, G.; Yuan, Y. Prediction Method of Beijing Electric-Energy Substitution Potential Based on a Grid-Search Support Vector Machine. *Energies* **2022**, *15*, 3897. [[CrossRef](#)]
33. Xia, D.; Zheng, Y.; Bai, Y.; Yan, X.; Hu, Y.; Li, Y.; Li, H. A parallel grid-search-based SVM optimization algorithm on Spark for passenger hotspot prediction. *Multimedia Tools Appl.* **2022**, *81*, 27523–27549. [[CrossRef](#)]
34. Zhang, J.; Wang, J.; Wei, M.; Zheng, Y.; Yang, Z. Optimal PI controller tuning for dynamic TITO systems with rate-limiters based on parallel grid search. In Proceedings of the 2021 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 22–24 May 2021; pp. 5808–5811. [[CrossRef](#)]
35. Kaewwiset, T.; Temdee, P. Promotion Classification Using DecisionTree and Principal Component Analysis. In Proceedings of the 2022 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON), Chiang Rai, Thailand, 26–28 January 2022; pp. 489–492. [[CrossRef](#)]
36. Sadouni, O.; Zitouni, A. Task-based Learning Analytics Indicators Selection Using Naive Bayes Classifier and Regression Decision Trees. In Proceedings of the 2021 International Conference on Theoretical and Applicative Aspects of Computer Science (ICTAACS), Skikda, Algeria, 15–16 December 2021; pp. 1–8. [[CrossRef](#)]
37. Rahman, A.; Akter, Y.A. Topic Classification from Text Using Decision Tree, K-NN and Multinomial Naïve Bayes. In Proceedings of the 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, 3–5 May 2019; pp. 1–4. [[CrossRef](#)]
38. Zheng, H.; Xiao, F.; Sun, S.; Qin, Y. Brillouin Frequency Shift Extraction Based on AdaBoost Algorithm. *Sensors* **2022**, *22*, 3354. [[CrossRef](#)]

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