# An Anomaly Detection Method Based on GCN and Correlation of High Dimensional Sensor Data in Power Grid System

Liu Weiwei[1(✉)], Lei Shuya[1], Zheng Xiaokun[1], Li Han[1], Wang Xinyu[1], Liang Xiao[1], and Xu Houdong[2]

[1] Artificial Intelligence On Electric Power System State Grid Corporation Joint Laboratory (GEIRI), Global Energy Interconnection Research Institute Co. Ltd., Beijing 102209, China
[2] State Grid, Sichuan Electric Power Company, Chengdu 610041, China

**Abstract.** Monitoring data or sensor data could reflect the working situation of power grid system at a fine-grained level. Specifically, when an anomaly event happened, some variations will appear and propagate between these interrelated sensor data. However, their latent relationship are complex and difficult to capture. To address this challenge, we propose a data-driven anomaly detection method, which performs real-time correlation analysis of sensor data and implements anomaly detection at runtime. Firstly, the method adopts the correlation coefficient calculation methods to obtain the time-varying correlation between sensed data. Additionally, graph is applied to represent the relationship between them. The edges of the graph are labeled with the degree of correlation and the nodes are marked with some statistical characteristics of the original sensor data. Moreover, an anomaly detection algorithm based on graph convolution network is implemented. The effectiveness of this approach is verified based on real power grid datasets.

**Keywords:** Anomaly detection · Time-series data · GCN · Correlation analysis

## 1 Introduction

In recent years, an increasing number of sensors has been deployed in the physical world. Due to the collaboration of large autonomous and heterogeneous sensors, new challenges for developing IoT applications have emerged. Consequently, IoT data and their analysis are becoming a hot topic because they can provide a consistent way to access sensor data from different stakeholders [1–4].

One of the most important purpose of analysis of sensor data is to detect potential anomalies in them. Hawkins [5] defines anomalies as those data that are distinctive in a data set, raising the suspicion that they do not arise from random deviations, but from completely different mechanisms. Anomaly detection is the process of discovering anomaly data in data resources based on various data processing models and techniques. Traditional anomaly detection mostly targets outliers, based on statistics, distance, density, and clustering. However, these methods usually consider sensor data in isolation and ignore correlations between sensors.


X. Wang et al. (Eds.): AICON 2021, LNICST 396, pp. 444–454, 2021.
https://doi.org/10.1007/978-3-030-90196-7_38

In real scenarios, there is a certain connection among the data of sensors. One sensor data may be normal on its own, but may be anomalous when considered together with other sensor data [6]. In addition, if more relevant sensors are considered together, it is possible to detect anomalies earlier and avoid future serious damage. However, due to the complex and diverse correlation existing in the large number of sensors, and the correlations between sensors vary with the situation, it is difficult to handle this situation.

Hence, a correlation-based sensor data anomaly detection method is proposed to address the above issues, which can carry out anomaly detection at runtime based on real-time sensor data correlation analysis.

The main contributions of this manuscript study are listed below.

- We try to capture the correlation of multiple monitoring data and represent it as graph data which will benefit anomaly detection in the power grid system.
- We use GCN to learn the correlation graph data which will detect the changes in the correlation data. The principle behind this method is the assumption that multiple monitoring data will impact others in a certain time.
- Experiments on power quality datasets demonstrate that the proposed method outperforms several anomaly detection methods.

## 2 Related Work

Anomaly detection methods [7–9] for outlier points can be broadly classified into four categories, which are statistical-based, distance-based, density-based, and clustering-based. With the continuous development of anomaly detection techniques, anomaly detection has started to focus more attention not only on outliers but also on the massive amount of temporal data [10–17]. Therefore, some research focus on the area of temporal data anomaly detection. The anomaly detection of time series data usually performs related anomaly detection by using the time characteristics of the data which is analyzing the data pattern in a specific time period.

There are three main cases of anomalies in temporal data, the first is contextual anomalies, which are point anomalies in temporal data, and the contextual anomalies must be in the context of the sequence data. The second type is a subsequence anomaly whose subsequence pattern is very different from the pattern of the overall sequence. The third type is the anomalous sequence that is different from the base sequence. Such anomalies are determined by giving a base sequence, and by comparing the test sequence with the base sequence to determine whether the test sequence is anomalous or not.

Recently, some research on anomaly detection of time series data has been done. Fei Huan et al. [18] used a sliding window model to detect and verify anomalous data but it requires a priori knowledge of the data and has poor applicability.

Wang Lei et al. [19] used the support vector regression estimation model, which takes into account the characteristics of the smoothness of the regression curve to achieve the separation of anomaly data and improves the accuracy of the power station performance. However it has poor applicability when targeting subseries.

Chen et al. [20] used network density and decay factor for anomaly detection with high accuracy and low time overhead to satisfy real-time. However, many parameters need to be set.

A Hadoop-based time-series anomaly detection method was proposed by Zhang et al. [21]. To address the problem of high computational complexity of the traditional DTW algorithm, the constraint calculation method of intelligent matching of salient features is introduced, which effectively reduces the time complexity and space complexity of the algorithm while ensuring a high detection accuracy through local restrictions on non-linear paths. However, the Hadoop platform has high latency and the actual operability is yet to be tested.

Cai et al. [22] proposed a new anomaly detection algorithm for time series data by constructing a distributed recursive computation strategy as well as a k-nearest neighbor selection strategy. Yan et al. [23] used a probability density function instead of the Euclidean distance to determine the similarity of two sequences, but this algorithm does not have a suitable method to determine the size of the detection window and has certain data requirements. Xu Junmei et al. [24] set up a constant deviation function to find the minimum value as a way to set the check threshold and use the Kmeans++ algorithm to cluster the data to obtain the set of anomaly detection objects, which effectively improves the real-time anomaly detection.

However, it is necessary to adjust the number of clusters in advance, otherwise the difference between anomalies and normal points may be small.

Qiu Yuan et al. [25] proposed a streaming data anomaly detection method based on long short-term memory (LSTM) network and sliding window. Firstly, data prediction by LSTM to find the predicted difference, and then the distribution of the difference sequence is modeled within the sliding window to dynamically assign a more appropriate anomaly score for each data to improve the accuracy of streaming data anomaly detection. However, it also requires a priori knowledge of the data, which is less applicable, and at the same time, there is often relatively large noise in the sequence. Liu Fenglin et al. [26] proposed a DBSCAN-based threshold selection algorithm for timing data anomaly detection, and did experimental validation based on Yahoo's EGADS framework for secondary development, with good results and practical engineering value. However, the characteristics of periodicity and seasonality are not fully considered in the modeling of time-series data, and the algorithm has a large time window for detecting anomalies, which needs to be improved in the time-series prediction and error metric models.

Li Rui et al. [27] proposed a time-series based anomaly detection method, which firstly models the user behavior to predict the development trend, and then performs anomaly detection based on the actual behavior, which can effectively detect user's violation and network attack, but the limited extracted feature values cannot completely demonstrate the user behavior, and the seasonal ARIMA model cannot fully match all user traffic.

Although many works have been proposed, there are various challenges in anomaly detection of time-series data, such as the length of the anomalous subsequence is difficult to determine effectively. Moreover, the anomaly does not appear in training set, and there is often relatively large noise in the time series, i.e., the distance between the anomaly and the normal point is small or even difficult to distinguish. In addition, anomaly detection methods for temporal data usually utilize only the sensed data itself and the static relationships between sensed data, and do not fully utilize the time-varying correlations between sensors. As a result, we propose a correlation-based anomaly detection method

for sensor data that can dynamically discover the relationships between data at runtime and find data anomalies based on them.

## 3 Methodology

Our method try to capture the correlation of time-series data and its latent relationships. These correlations between time-series data could be represented as graph data. Moreover, we train GCN (graph convolution neural network) to classify the new high dimensional data. More details are shown in Fig. 1. As shown in Fig. 1, the left part is the training process. We calculate the CC (correlation coefficient) and feature of these sensor data to build graph data. The GCN based model is trained as classification model which is applied to detect possible anomaly in un-labeled sensor data.

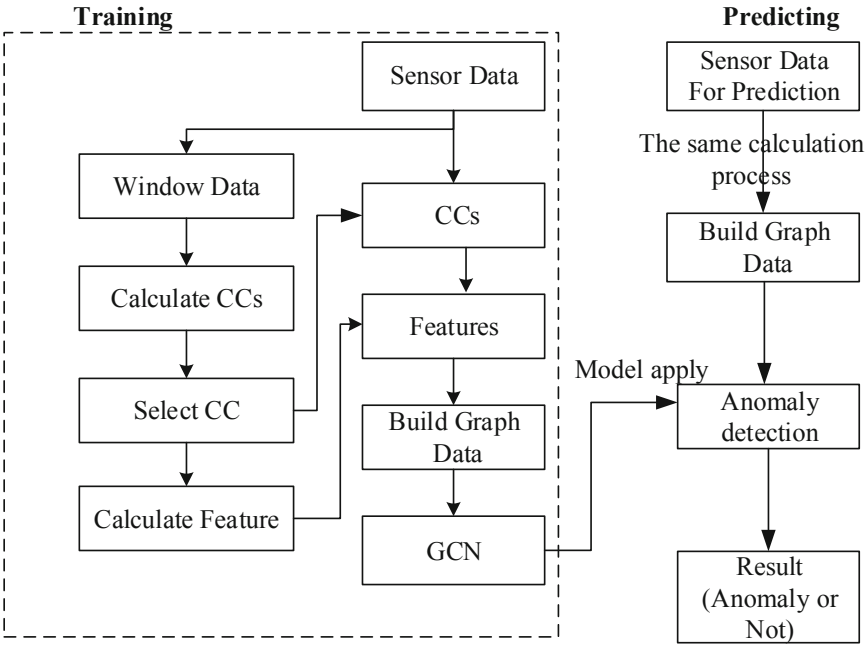


**Fig. 1.** The diagram of our method.

### 3.1 Calculation of Correlation Coefficient

The sensor is often in the form of time-series data, and we can use a fixed time window to calculate the CCs (Correlation Coefficient) for each pair of them. Figure 2 shows an example of the process of calculation of CC between different monitoring points. Each monitoring point owns multiple sensor data, we calculate the data correlation coefficient between each pair of sensor data. One example is shown in Fig. 2.

Function of Correlation Coefficient

| TimeStamp | Mornitoring point1 | | ..... | Mornitoring point1 | | ..... | ..... | CC of MP1,MP2 |
|---|---|---|---|---|---|---|---|---|
| | Sensor 1 | Sensor 2 | | Sensor 1 | Sensor 2 | | | ..... |
| 11:10 AM | 0.7439 | 56.359 | | 0.6501 | 59.307 | | | 0.39 |
| 11:11 AM | 0.7438 | 53.474 | | 0.6505 | 59.861 | | | 0.45 |
| 11:12 AM | 0.6826 | 51.629 | | 0.6499 | 56.359 | | | - |
| 11:13 AM | 0.6586 | 44.368 | | 0.6442 | 51.245 | | | 0.47 |
| 11:14 AM | 0.6567 | 45.203 | | 0.5181 | 50.687 | | | - |
| ... | ... | ... | | ... | ... | | | ... |

**Fig. 2.** Calculation of correlation coefficient (time series)

According to the characteristics of specific time-series data, different correlation evaluation methods can be applied to represent the extent of interrelation of them. As shown in Fig. 2, some widely used methods, such as MIC [9] and dCor [11] are utilized in our method.

We calculate the correlation between the features according to different correlation calculation methods to obtain different time series correlation matrices, and then merge the correlation matrices into a three-dimensional matrix array, as shown in Fig. 3.

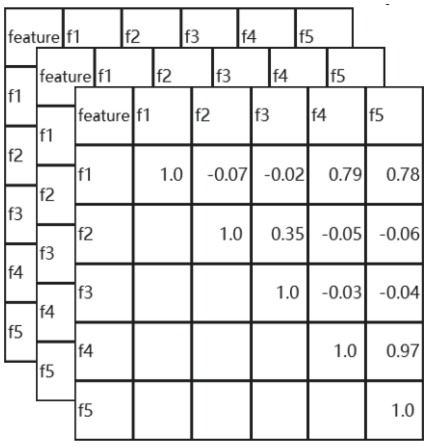


**Fig. 3.** Three-dimensional matrix array

The different correlation calculation methods are as follows. Firs is the Euclidean distance and is shown in Eq. 1.

$$\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{1}$$

The formula of calculating the Chebyshev distance is expressed in Eq. 2.

$$\lim_{p\to\infty}\left(\sum_{i=1}^{n}|x_i-y_i|^p\right)^{1/p}=\max|x_i-y_i| \tag{2}$$

The formula of calculating Cosine Similarity is illustrated in Eq. 3.

$$\cos(\theta)=\frac{\sum_{k=1}^{n}x_{1k}x_{2k}}{\sqrt{\sum_{k=1}^{n}x_{1k}^2}\sqrt{\sum_{k=1}^{n}x_{2k}^2}} \tag{3}$$

The formula of calculating Pearson Correlation Coefficient is shown in Eq. 4.

$$\rho_{X,Y}=\frac{\sum XY-\frac{\sum X\sum Y}{N}}{\sqrt{\left(\sum X^2-\frac{(\sum X)^2}{N}\right)\left(\sum Y^2-\frac{(\sum Y)^2}{N}\right)}} \tag{4}$$

### 3.2 Feature Selection

Obviously, as the number of sensors increases, the number of CCs will also in-crease. When time series data are multi-dimensional data, correlation calculations will become very complicated. Many anomaly detection algorithms are difficult to solve high-dimensional data anomaly detection well. Existing work [8] proposed a latent correlation vector, which composes all CCs into a vector. Considering that not all sensors have a correlation relationship, the CC between two uncorrelated sensors has no effective value. Therefore, based on this consideration, we designed a correlation selection function, which is used to select CCs according to the relationship between the two data relevant strength and stability.

$$x^2=\sum\frac{(A-E)^2}{E} \tag{5}$$

### 3.3 GCN Anomaly Detection Model

Graph Convolutional Neural Network (GCN) can learn the correlation between nodes effectively and be successfully applied to analyze social network.

We take advantage of GCN's in extracting correlation features between nodes and apply it to the power indicator correlation graph G = (V,E) for anomaly detection. Among them, V represents a power feature, and E represents the relationship between power indicators. Assuming that the number of vertices of the index feature correlation graph is N and the number of edges is M, then the graph can be represented by an adjacency matrix A of size N × N. The attribute of the vertex is the feature vector of each indicator,

and the attribute of the edge is the correlation between the two indicators, which is calculated by the correlation calculation methods introduced in the previous section.

Our goal is to establish a correlation relationship between various power indicators within a period of time, so as to detect anomaly event when it occurs or will occur in the power network. Firstly, a correlation coefficient-based graph data are prepared based on different data source. What is more, some certain anomaly and non-anomaly graph data are labeled to construct a supervised dataset. In addition, we trained GCN to find the relationship between the power indicators which can classify and detect anomaly event in these time series data. The process is shown in Fig. 4, which shows the graph data constructing process. There are many power indicators, such as voltage, current, harmonics, etc.

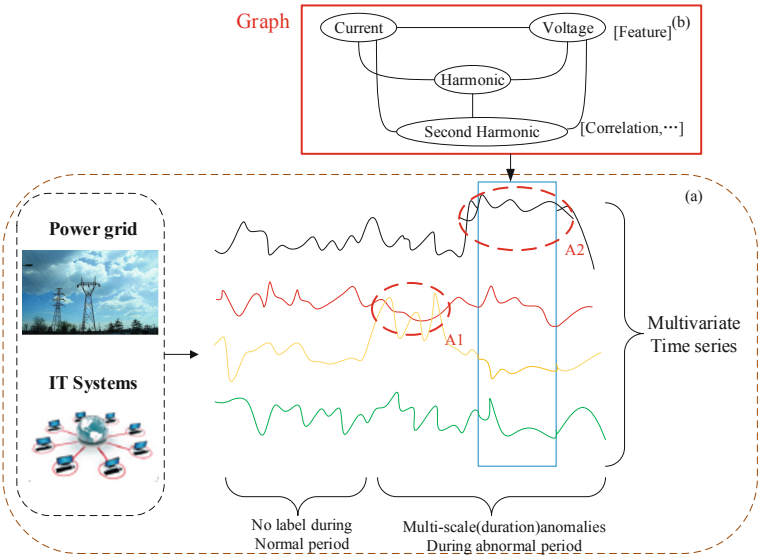


**Fig. 4.** Unsupervised anomaly detection and diagnosis in multivariate time series data.

## 4 Experiments

### 4.1 Dataset

The datasets used in our experiments are the real sensor data and artificial maintenance records from power quality data of charging posts in a region. Dozens of power quality data measurement points for the number of charging posts have been deployed in the region. Each measurement point consists of 4 attributes, including harmonic current content, total voltage harmonic distortion rate, fundamental current content, and flicker composition. Each measurement point generated a set of values per 30 s. We choose three datasets to verify the effectiveness of our data-driven anomaly detection method, and construct the experimental data set by adjusting the time period. The detail information of the datasets is shown in Table 1.

**Table 1.** Detail information of datasets in our experiments

| Datasets | Time period | Number of measuring points | Time window |
|---|---|---|---|
| DS1 | 2015–04-29 00:00:00 to 23:59:59 | 5 | 1 h |
| DS2 | 2015–09-20 00:00:00 to 23:59:59 | 7 | 30 min |
| DS3 | 2015–11-15 00:00:00 to 23:59:59 | 6 | 2 h |

### 4.2 Data Preprocessing

Standardization requires calculating the mean and standard deviation of features. The formula is as follows:

$$x' = \frac{x - \overline{X}}{S} \tag{6}$$

For feature encoding of time series data, the core is to set a threshold. The value greater than the threshold is assigned to 1, and the value less than or equal to the threshold is assigned to 0. The formula is as follows:

$$x' = \begin{cases} 1, x > threshold \\ 0, x \leq threshold \end{cases} \tag{7}$$

The missing values of time series data are calculated by linear regression. Based on the current data set, the regression equation is established. For the objects with null values, the known attribute values are substituted for the equation to estimate the unknown attribute values, then the estimated values are used to fill lost value.

### 4.3 Experiment Setup

Our experiment is conducted in a desktop computer with configuration as follows: CentOS 7.7, four Intel Core i7–8750, 8.00 GB RAM, 200 GB hard disk and GTX2080Ti GPU. All the algorithms mentioned above are implemented in Python and Pytorch 1.6.

The model settings for GCN is as follows. The node feature matrix $X \in \mathbb{R}^{N \times d}$ is constructed with $d = 4$ corresponding to the minimum and maximum latitude and longitude extents of the zone corresponding to the node. The encoder uses $L = 2$ layers of GCN to learn the node-level embedding. The model is trained for 200 epochs with a batch size of 30. In training set, 10% is kept out as validation set for early stopping.

### 4.4 Result and Analysis

Baselines. We compare our method with two commonly used anomaly detection methods.

- SVM. The baseline method uses One-Class SVM model (OC-SVM), it applies the SVM to learn a model to determine whether the new data belongs to a specific class (whether it is normal data), if it does not belong to this class, then it is anomaly.

- CNN. The baseline method uses CNN to detect anomalies in time series data. The convolutional neural network has four layers. Each input channel shares the same convolutional layer. The essence of the MaxPool layer is to take the largest element operation of the vector for each channel to adapt to different lengths of time series data.
- GCN_Corr. This is our method. We capture the correlation of time-series data and their latent relationships. We train GCN to classify the new high dimensional data based on these correlations between time-series data.

**Table 2.** The AUC score for anomaly detection in different datasets.

| Method dataset | SVM | CNN | GCN_Corr |
|---|---|---|---|
| DS1 | 0.352 | 0.561 | 0.712 |
| DS2 | 0.290 | 0.623 | 0.689 |
| DS3 | 0.411 | 0.518 | 0.774 |

The performance of different methods for anomaly detection is reported in Table 2, we can see that GCN_Corr constantly outperforms the other methods in different datasets.

Compared with SVM and CNN, our approachpay attention to the correlation of each time series data. By modeling the varying relationship of each time series data, we can better mine the relationship between the abnormality and the change trend of each time series data, and then use the graph convolutional neural network to detect the time series anomalies.

## 5 Conclusion

The correlation between sensor data is dynamic and time dependent. According to the correlation between sensor data, more related sensors are able to be detected, and the effectiveness of anomaly detection for these sensor data can be promoted. In this paper, we propose a data-driven anomaly detection method for handling sensor data. The proposed method builds graph data based on correlation analysis, and then uses graph convolution to capture changes of sensor data, which can analyze changes of dynamic sensor data in the IoT environment. Our method performs anomaly detection of multi-dimensional sensors and obtain higher accuracy. We apply our method to the anomaly detection of the power quality data in power grid system, and verify that our anomaly detection method can detect anomalies with high precision and recall through a series of experiments.

**Acknowledgement.** This work is supported by the science and technology project of State Grid Corporation of China: “Research on data governance and knowledge mining technology of power IOT based on Artificial Intelligence” (Grand No.5700-202058184A-0–0-00).

## References


1. Han, Y.B., Liu, C., Su, S., et al.: A decentralized and service-based approach to proactively correlating stream data. In: International Conference on Internet of Things, pp. 93–100 (2016)
2. Chu, V.W., Wong, R.K., Liu, W., et al.: Traffic analysis as a service via a unified model. In: IEEE International Conference on Services Computing, pp. 195–202. IEEE (2014)
3. Zhang, J., Radia, N., Li, Z., et al.: An infrastructure supporting considerate sensor service provisioning. In: The 6th IEEE International Conference on Service Oriented Computing and Applications (SOCA), pp. 69–76. IEEE (2013)
4. Guilly, T.L., Olsen, P., Ravn, A.P., et al.: HomePort: middleware for heterogeneous home automation networks. In: IEEE International Conference on Pervasive Computing and Communications Workshops, pp. 627–633. IEEE (2013)
5. Atkinson, A.C., Hawkins, D.M., et al.: Identification of outliers. Biometrics **37**(4), 860 (1981)
6. Budgaga, W., Malensek, M., Pallickara, S.L., et al.: A framework for scalable real-time anomaly detection over voluminous, geospatial data streams. In: Concurrency & Computation Practice & Experience, pp. 1–24 (2017)
7. Kieu, T., Yang, B., Jensen, C.S., et al.: Outlier detection for multidimensional time series using deep neural networks. In: 2018 19th IEEE International Conference on Mobile Data Management (MDM), pp. 125–134 (2018)
8. Subramaniam, S., Palpanas, T., Papadopoulos, D.: Online outlier detection in sensor data using non-parametric models. In: 32nd International Conference on Very Large Data Bases, pp. 187–198 (2006)
9. Nguyen, H.T., Thai, N.H.: Temporal and spatial outlier detection in wireless sensor networks. ETRI J. **41**(8), 437–451 (2019)
10. Huang, H.: Data anomaly detection method of sensor nodes in Internet of Things. Computer Simul. **05**, 167–170 (2012)
11. Qi, Z., Yupeng, H., Cun, J.: Edge computing application: real-time anomaly detection algorithm for sensing data. J. Comput. Res. Dev. **55**(3), 524–536 (2018)
12. Xie, M., Hu, J., Guo, S.: Distributed segment-based anomaly detection with kullback–leibler divergence in wireless sensor networks. IEEE Trans. Inf. Forensics Secur. **12**(1), 101–110 (2017)
13. Tian, L., Zhang, D.: An anomaly detection method of sensor data based on information entropy. Comput. Eng. Softw. **39**(09), 77–81 (2018)
14. Grabaskas, N., Si, D.: Anomaly detection from kepler satellite time-series data. In: International Conference on Machine Learning & Data Mining in Pattern Recognition, pp. 220–232 (2017)
15. Khatkhate, A., Ray, A., Keller, E., et al.: Symbolic time-series analysis for anomaly detection in mechanical systems. IEEE/ASME Trans. Mechatron. **11**(4), 439–447 (2006)
16. Laptev, N., Amizadeh, S., Flint, I., et al.: Generic and scalable framework for automated time-series anomaly detection. In: 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1939–1947 (2015)
17. Burgess, M.: Two dimensional time-series for anomaly detection and regulation in adaptive systems. In: 13th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management, pp. 169–180 (2002)
18. Fei, H., Xiao, F., Li, G., et al.: An anomaly detection method of wireless sensor network based on multi-modals data stream. Chin. J. Comput. **40**(8), 1829–1842 (2017)
19. Wang, L., Zhang, R., Sheng, W., et al.: Regression forecast and abnormal data detection based on support vector regression. Proc. CSEE **08**, 94–98 (2009)
20. Chen, Y.: Density-based clustering for real-time stream data. In: ACM International Conference on Knowledge Discovery & Data Mining, pp. 133–142 (2007)

21. Zhang, J., Li, B., Liu, X., et al.: Abnormal time series detection in wireless sensor network based on hadoop. Chin. J. Sens. Actuators **12**, 1659–1665 (2014)
22. Cai, L., Thornhill, N., Kuenzel, S., et al.: Real-time detection of power system disturbances based on k-nearest neighbor analysis. IEEE Access **5**, 5631–5639 (2017)
23. Yan, Q.Y., Xia, S.X., Feng, K.W., et al.: Probabilistic distance based abnormal pattern detection in uncertain series data. Knowl.-Based Syst. **36**, 182–190 (2012)
24. Xu, J.M.: Anomaly detection of mobile network interaction behavior based on Internet of Things. J. Eastern Liaoning Univ. (Nat. Sci. Ed.) **28**(01), 34–38 (2021)
25. Qiu, Y., Chang, X., et al.: Stream data anomaly detection method based on long short-term memory network and sliding window. J. Comput. Appl. **40**(05), 1335–1339 (2020)
26. Liu, F.: Research on threshold selection algorithm of time series data anomaly detection based on DBSCAN. Modern Comp. **04**, 3–6 (2020)
27. Li, R., Jia, Y., Jiao, Z., et al.: Network behavior anomaly detection based on time series. Commun. Technol. **53**(10), 2550–2554 (2020)