Ad Hoc & Sensor Wireless Networks, Vol. 53, pp. 141–164
Reprints available directly from the publisher
Photocopying permitted by license only
DOI: 10.32908/ahswn.v53.9001

A High-Dimensional Timing Data Cleaning Algorithm for Wireless Sensor Networks

JINGJING ZHOU[1], XIAOKANG YU[1], JILIN ZHANG[2], HANXIAO SHI[3], YUXIN MAO[3], JUNFENG YUAN[4] AND DONGYANG OU[4]

[1]School of Information and Electronic Engineering, Zhejiang Gongshang University, Hangzhou 310018, China
[2]School of Cyber Security, Hangzhou Dianzi University, Hangzhou 310018, China
[3]School of Management and E-Business, Zhejiang Gongshang University, Hangzhou 310018, China
[4]School of Computer and Software, Hangzhou Dianzi University, Hangzhou 310018, China

Received: February 11, 2022. Accepted: June 7, 2022.

Wireless Sensor Networks (WSN) use many sensor nodes to monitor various environmental information in designated areas in real-time, which has broad application prospects in many fields and industries. Due to the sensor's physical fault or technical defect, there are some errors in the collected data; therefore, it is necessary to clean and repair the data before they are used. This paper proposes a high-dimensional sequential data cleaning algorithm for WSNs. The algorithm combines the correlation between different dimensions and the temporal correlation characteristics within the same dimension. Firstly, the data is preprocessed, and the abnormal dimension is determined by combining the prior knowledge and correlation calculation. Then, the algorithm of dynamic programming and speed constraint is used to determine the outliers and mark the abnormal dimensions. Finally, the autoregressive model with exogenous variables is used to repair outliers. Experiments are carried out on a real WSN dataset in this paper. The results show that the repair effect of the proposed algorithm is better than the single dimension benchmark algorithm.

Keywords: Wireless sensor networks, data cleaning, high-dimensional time series, speed constraint, dynamic programming

1 INTRODUCTION

With the popularization and development of information technology, wireless sensor networks (WSN) are widely used in intelligent monitoring, behavior

*Corresponding author: E-mail address: yu1063870023@163.com, zhoujingjing@zjgsu.edu.cn

analysis and other fields[1,2]. A large amount of data accumulated by various industries through WSN devices has become the product, providing essential data support for big data and artificial intelligence technology[3,4]. These data are used for knowledge extraction, application decision-making and other intelligent services, so the importance of data quality is beyond doubt[5,6].

Due to the physical and technical characteristics of the WSN, such as harsh sensor deployment environment, limited network bandwidth, environmental noise interference, and other factors[7,8], there will be a certain degree of quality problems in the process of data acquisition, transmission and recording[9,10]. The data with quality problems can not accurately represent the real world, resulting in increased cost and data analysis risk[11]. Eliminating anomalies in data and improving data quality can enhance the optimization effect of big data and artificial intelligence in data analysis[12].

How to effectively identify and repair anomalies in data has become an essential topic in data management[13]. In daily life and industrial field, temperature, humidity, voltage and other data collected by WSN are time-series data[14]; in other words, data usually have specific change rules with the change of time. Data quality is the core factor of the network. Only high-quality data can ensure the effectiveness of WSN services. Therefore, data mining technology extracts knowledge from data to serve users[15,16]. Consequently, it is necessary to repair the outlier in the original time series collected by the sensor, that is, to clean the original data, eliminate the dirty data and improve the data quality[17].

At present, the algorithms for cleaning a single time series using correlation characteristics are mainly divided into three aspects: based on smoothing, statistics, and constraints[18]. The cleaning algorithm based on smoothing can smooth the outlier by adjusting the sliding window and taking part in the calculation with given parameters. The standard techniques are Simple Moving Average (SMA); Exponentially Weighted Moving Average (EWMA). Although the time cost of smoothing technology is minimal, it will change the original normal data, affecting cleaning accuracy. Statistical-based cleaning algorithm cleans data by learning from data[19]. For example, the data cleaning algorithm is based on the Hidden Markov Model (HMM). However, this algorithm relies on establishing the model and cannot solve continuous errors. The constraint-based cleaning algorithm uses the correlation between sequences to determine whether the values of lines are abnormal, for example, sequence constraints and speed constraints. But the rules are usually provided by domain experts, so it is difficult to give a reasonable constraint for the dynamic time-series data[20,21].

The research on anomaly detection of high-dimensional time series data has made some progress in recent years. Data acquisition is usually completed by cooperative wireless sensor equipment groups in practical engineering scenarios; analyzing the correlation between time series of different dimensions can improve the effect and efficiency of data cleaning[22]. The paper[23,24] analyzed the tasks, algorithms and performance of anomaly detection problems on high-dimensional data. The paper[25] proposed a distance-based anomaly detection algorithm for high-dimensional datasets. In the paper[26], the concept of a score vector is used to calculate the probability of abnormal parts. The paper[27] proposed a latent sequence correlation calculation model based on existing work for the anomaly detection of industrial data sequences.

WSN data usually correlates. The data is linked on the time axis, and the data of adjacent nodes are related in space[28,29]. Most previous studies used the correlation characteristics on the time axis and ignored the spatial correlation.

This paper proposes a high-dimensional time series data cleaning (HTD-Cleaning) algorithm based on correlation assistance analysis for WSN. The algorithm combines the correlation between different dimensions and the temporal correlation characteristics within the same dimension. Firstly, the data is preprocessed, and the abnormal dimension is determined by combining the prior knowledge and correlation calculation; secondly, the algorithm of dynamic programming and speed constraint is used to determine the outliers and mark the abnormal dimensions; finally, the autoregressive model with exogenous variables is used to repair the outliers. In this paper, experiments are carried out on a real wireless sensor network dataset collected by 21 Mica2Dot sensors; the root means square error of the HTD-Cleaning algorithm is about 0.5 times that of other comparison algorithms verify the effectiveness of this algorithm in cleaning high-dimensional time series data. Although the time complexity is $O\left(N^2\right)$ and N is the number of time-series data nodes, the repair effect of this algorithm is significantly better than other comparison algorithms.

The main contributions of this paper are as follows:

1. An algorithm for cleaning high-dimensional sequential data for wireless sensor networks is proposed. This algorithm can extract information from high-dimensional data for knowledge reasoning by deeply mining the relevant mechanism of high-dimensional time-series data, which is helpful for comprehensive anomaly detection and repair of high-dimensional data.
2. This paper verifies that the proposed algorithm performs best repair accuracy through experiments on real wireless sensor network time-series datasets.

In section 2, the research problems are introduced, time series, velocity constraints, sequence correlation, data cleaning problems and repair results are defined, and the algorithm framework is briefly described. In Section 3, the cleaning algorithm of high-dimensional sequential data for wireless sensor networks is in detail. Section 4, through the experiments on real datasets, compared with the existing algorithms in terms of repair effect and time cost. Finally, in Section 5, the work of this paper is analyzed and summarized.

2 PRELIMINARIES

Definition 1 (Time Series) Time series $X = x_1, x_2, \ldots, x_n$, It is a series of continuous data nodes with timestamp collected by sensor pieces of equipment, $x_i = \left(d_i,\ t_i\right)\middle|1 \le i \le n$ represents the i-th data node, where d_i is a numerical value, t_i means the timestamp corresponding to the value, n is the number of time-series data nodes.

Definition 2 (Multidimensional Time Series) Multidimensional time series $H = \left\{X_1, \mathrm{X}_2, \ldots, \mathrm{X}_k\right\}$, it is a set of K (k > 1) time series, $X_i = x_{i1}, x_{i2}, \ldots, x_{in}$ represents one time series, i.e., a dimension data, where $x_{i,j}$ represents the data node at j time of the i-th time series.

Definition 3 (Speed Constraint) Given speed constraint $S = \left(S_{min}, S_{max}, w\right)$, In time series $X = x_1, x_2, \ldots, x_n$, any data node $x_i, x_j, \left|t_i - t_j\right| \le w$ are satisfied $s_{min} \le \frac{x_j - x_i}{t_j - t_i} \le s_{max}$, the time series meets the speed constraint S.

Definition 4 (Sequence Correlation) Assumes two-time series X_i and $X_j, n_i = n_j$, defined in sequence $\mathrm{X_i}$ and $\mathrm{X_j}$, the correlation calculation function is $Corr\left(X_i, X_j\right) \in [0,1]$, the correlation between sequence $\mathrm{X_i}$ and $\mathrm{X_j}$ is determined as follows:

1. $Corr\left(X_i, X_j\right) \in [c,1]$, then sequence X_i and X_j has strong correlation;
2. $Corr\left(X_i, X_j\right) \in [0,c]$, then sequence X_i and $X_j$are not relevant.

Definition 5 (WSN Oriented High-Dimensional Time Series Data Cleaning Problem Definition) For a given k-dimensional time series dataset $H = \left\{X_1, \mathrm{X}_2, \ldots, \mathrm{X}_k\right\}, n_1 = n_2 =, \ldots, = n_k$, to achieve the following tasks:

1. Design correlation calculation function $Corr\left(X_i, X_j\right)$, the correlation between any two dimensions in k-dimensions time series is calculated and quantified;
2. According to the correlation matrix in the task (1), the abnormal dimension is determined on the dataset to be detected in H;
3. The set E of outliers in the abnormal dimension is detected, $\frac{x_i - x_j}{t_i - t_j}\left\langle S_{min} \quad or \ \frac{x_i - x_j}{t_i - t_j}\right\rangle S_{max},\ x_i \in E$ and repair the data nodes in E to meet the speed constraint and close to the correct value.

Definition 6 (Repair Result Definition) Repair result of time series X is X', in the window w, the timestamp is defined as t_i, the value of the data node $\mathrm{x_i}$ fixed to $x_i^{'}$, so that the time series satisfies the speed constraint S, and the timestamp remains unchanged after repair.

3 ANOMALY DETECTION AND REPAIR ALGORITHM FOR HIGH-DIMENSIONAL TIME-SERIES DATA

3.1 Overview

An accidental outlier can occur in one or several dimensions of the WSN datasets[30,31]. The possible abnormal dimensions can be screened for the high-dimensional time series with a known correlation relationship by calculating the significant change of the correlation parameters[32]. However, it is still not possible to determine the specific anomaly problem. The main reasons are as follows: (1) The time-series correlation is symmetric and undirected. When the correlation parameters of two dimensions change, it is uncertain which of the two dimensions is abnormal;(2) The location of the outliers in the dimension with the anomaly is not recognized, so the outlier cannot be accurately repaired.

In this paper, the HTD-Cleaning algorithm for wireless sensor networks is proposed. The main idea is to determine the abnormal dimension by using the correlation changes between different dimensions, detect and repair the outliers through the temporal correlation characteristics within the dimension to clean the high-dimensional temporal data of WSN. The algorithm includes four parts: data preprocessing, correlation calculation, anomaly detection and anomaly repair.

(1) Data preprocessing: Some quality problems in the original data collected by sensor equipment[33,34]. Therefore, in the data preprocessing part, timestamp alignment and missing value filling are needed for the original time series data, and the processed data are used as the input of the later analysis module;

(2) Correlation calculation: firstly, the correlation matrix is generated by calculating the correlation of the output data from step (1);

(3) Anomaly detection: this part determines the abnormal dimension through the correlation matrix. The high-dimensional time-series correlation obtained by prior knowledge can evaluate whether the correlation among the measurements in the correlation matrix significantly decreases in this step. If the original high correlation of a dimension is reduced considerably, it can be considered that there is an outlier within the dimension. For the abnormal dimension, the anomaly detection algorithm based on constraint is used to determine the outlier, to label the abnormal dimension;

(4) Anomaly repair: through step (3), we get the abnormal dimension with labeled information. In this part, using the temporal correlation charac-

teristics and the minimum modification principle of data cleaning, we use the autoregressive model with exogenous variables to repair the outliers.

3.2 Abnormal dimension identification

In the high-dimensional time series data collected by wireless sensor networks, there are usually certain relationships between some data dimensions. When the data of a particular dimension is abnormal, the abnormal dimension can be identified by other dimensions in the normal state to assist in identifying and correcting outliers.

Since the original data collected by each sensor node may not be collected at the same time or at different collection frequencies, and missing values are common in time series, the value of time series varies within a small range in a certain period of time, the Piecewise Aggregation Approximation (PAA) [35,36]can be used. The data of each dimension is processed to facilitate the subsequent calculation. At the same time, this step reduces the amount of data and is helpful for anomaly detection[37]. As a classical time series recombination technique, PAA can reduce a sequence of length L to a sequence of length L', which is usually a factor of L. PAA calculates the reconstructed sequence value by the following formula.

$$x'[j] = \frac{L'}{L} \sum_{i=\frac{L}{L'}(j-1)+1}^{\frac{L}{L'}j} x[i] \tag{1}$$

In other words, PAA divides the L-length sequence into parts $L^{'}$, the average value of each segment is calculated as the new data value of the recombination sequence.

For the reconstructed high-dimensional time series $H = \{X_1, X_2, ..., X_K\}$, $n_1 = n_2 =,...,= n_k$, in this paper, the Pearson correlation coefficient is used to describe the correlation between different dimensions. The calculation formula of Pearson correlation coefficient[38] is as follows:

$$corr(x_1, x_2) = \frac{cov(x_1, x_2)}{\sigma_{x1} \cdot \sigma_{x2}} \tag{2}$$

Where cov is the covariance between the two dimensions, σ_{x1} and σ_{x2} are the variances of X_1 and X_2, and the correlation coefficient of dimensional data *is* $corr(x_1, x_2) \in [-1,1]$, the greater the absolute value of the correlation coefficient of the two dimensions, the stronger the correlation. By collecting the original relevant information of the data as prior knowledge, it can help determine which dimensions contain outliers.

In the prior knowledge, if the correlation between the two dimensions is higher than the threshold c, that is correct$(\mathrm{x}_1, \mathrm{x}_2) \geq$ c, the two dimensions are considered highly correlated. If the correlation between the two dimensions X_1, X_2 decreases significantly, that is, the decline exceeds the threshold d, it can be considered that at least one of the two dimensions has an anomaly. The proportion of correlation changes can be counted to judge the abnormal dimensions more accurately. If more than a certain proportion of high correlation of a certain dimension changes significantly, it can be considered that the dimension may have outliers, label the dimension as an abnormal dimension. Subsequently, the outliers in the dimension are discriminated and repaired.

Algorithm 1 shows the steps to determine the abnormal dimension. The fourth line of the algorithm checks whether there is a high correlation between the two dimensions in the prior knowledge. Suppose the two dimensions have a high correlation; the algorithm lines 6-9 check whether the correlation between the dimensions has decreased significantly. The algorithm calculates

Algorithm 1: Correlation Evaluation

Input: high-dimensional time series H array, correlation coefficient matrix C, high correlation threshold c, correlation significantly reduced threshold d, correlation relationship significantly changed proportion θ
Output: suspicious sequence set ε

```
1.  foreach $X_k \in H$ do
2.      Initialize $correlated \leftarrow 0$, $declined \leftarrow 0$
3.      foreach $X \epsilon H$ do
4.          if $C_{k,j} \geq c$ then
5.              $correlated \leftarrow correlated + 1$
6.              $r \leftarrow \frac{cov\left(X_k, X_j\right)}{\sigma_{X_k} \cdot \sigma_{X_j}}$
7.              if $r < C_{k,j} - d$ then
8.                  $declined \leftarrow declined + 1$
9.              end
10.         end
11.     end
12.     if $\frac{declined}{correlated} \geq \theta$ then
13.         $\varepsilon \leftarrow \varepsilon \bigcup X_k$
14.     end
15. end
16. return $\varepsilon$
```

the proportion of high correlation decrease of each dimension in lines 12-14; if it exceeds θ, the algorithm considers that there are data anomalies in this dimension, and adds it to the set of abnormal dimensions.

3.3 Identification of outliers

This part will elaborate on speed constraints for outlier detection, to label the abnormal dimension. The paper[21] defines the value change of time series data and the timestamp change at the same time, that is $\frac{x[j]-x[i]}{t[j]-t[i]}$. Unlike the speed constraint for outlier detection and repair in paper[21], this paper uses speed constraint for outlier detection. After removing outliers from the abnormal dimension, the whole dimension data can meet the speed constraint.

Algorithm 2: outlier detection

```
Input: suspicious sequence x, speed constraint $SC(S_{min}, S_{max})$
Output: subscript set of outliers $U$
1: Initialize $anomaly[i] \leftarrow i-1, normal[i] \leftarrow -1, i = 1,\ldots,n$
2: for $j = 1; j \le n$ do
3:   for $i = 1; j > i$ do
4:     if $S_{min} \le \frac{x_j - x_i}{t_j - t_i} \le S_{max}$ and $anomaly[j] > anomaly[i] + (j-i-1)$ then
5:       $anomaly[j] \leftarrow anomaly[i] + (j-i-1)$
6:         $normal[j] \leftarrow i$
7:     end
8:   endfor
9: endfor
10: $index = anomaly[1] + (n-1); res = 1$
11: for $j = 2; j \le n$ do
12:   if $anomaly[j] + (n-j) < index$ then
13:       $index = anomaly[j] + (n-j)$ and $res = j$
14:   end
15: endfor
16: $U \leftarrow \{1,2,\ldots,n\} \setminus \{normal[res]\}$
17: return $U$
```

In this paper, the dynamic programming algorithm solves the set of outliers in the X dimension. X$[1 : i]$ represents the subsequence of the abnormal dimension time series, and $anomaly[i]$ indicates the minimum number of data nodes to be deleted to ensure the subsequence $X[1:\ i]$ under the condition of retaining the i-th data node, the speed constraint is satisfied. If two data nodes x_i, x_j, $(i < j)$, the speed constraint is satisfied, i.e $S_{min} \le \frac{x_j - x_i}{t_j - t_i} \le S_{\max}$, then delete the data nodes from i+1 to j-1 and keep the j-th data nodes and subsequences $X[1:\ j]$ must be satisfied. Then the state transition equation is as follows: $anomaly[\mathrm{j}] = anomaly[\mathrm{i}] + (j - i - 1)$. If we delete the $anomaly[j]$ data nodes, X [1: j] can satisfy the speed constraint, and delete j+1 and its subsequent data nodes to make the whole sequence meet the speed constraint; that is, we need to delete the $anomaly[j] + (n - j)$ data nodes. $normal[j] = i$ is used to recording the update path in the state transition process. After the calculation is completed, all data nodes under reservation are traced according to normal; that is, in the abnormal dimension, normal data nodes are considered correct and marked as true.

In algorithm 2, initializing $anomaly[\mathrm{i}] = i - 1$ means that $X[1, i]$ can meet speed constrain after deleting previous i-1 data nodes; initializing $normal[\ i\ ] = -1$ indicates the i-th data node will be the first remaining data node after deleting $anomaly[i]$ data nodes. Lines 2-7 of the algorithm perform state updating, lines 10-14 can find the last data node in the optimal solution, and line 16 starts from the last data node and traces all the reserved data nodes along normal.

3.4 Outliers repair

Paper[19] combines the time characteristics of anomaly detection with the minimum modification principle in data cleaning and proposes a repair framework IMR based on the autoregressive model with exogenous variables (ARX)[38]. The traditional algorithm directly uses the predicted value in anomaly detection as the candidate results of repair, which may change the data nodes, such as using the Autoregressive Model (AR)[39] or the Autoregressive Model With Exogenous Variables (ARX). The algorithm proposed in paper[19] considers the temporal correlation between errors and the principle of minimum modification in data cleaning; however, the labeled information in this paper comes from experts' manual annotation or highly reliable data sources. HTD-Cleaning algorithm can label abnormal dimension data automatically.

Autoregressive Model (AR)

$$x_t^{'} = c + \sum_{i=1}^{p} \phi_i x_{t-i} + \varepsilon_t \tag{3}$$

Where $x_t^{'}$ is the predicted value of x^t data nodes, p is the order of the model, ϕ_i is the parameter of the model, $c = \mu\left(1 - \sum_{i=1}^{p} \phi_i\right)$ is a constant, μ is the expectation of the sequence, and ε_t is the noise of the sequence, usually Gaussian white noise.

Autoregressive Model with Exogenous Variables (ARX)

$$x_t^{'} = x_t + \sum_{i=1}^{p} \phi_i \left(x_{t-i}^{'} - x_{t-i}\right) + \varepsilon_t \tag{4}$$

$x_t^{'}$ is the predicted value of data node x_t, and the rest is the same as the AR model.

The ARX model's final prediction result is affected by the observed value x_{t-i} and the predicted value $x_{t-i}^{'}$ of the data node before the current data node. Usually, the further distance indicates that the data node may be an outlier, if the predicted value $x_t^{'}$ is very different from observed values x_t, that is $\left|x_t^{'} - x_t\right| > \tau$, in which τ is a pre-defined threshold, the predicted value $x_t^{'}$ will be accepted as the results of cleaning. τ can be obtained from the statistical distribution of the difference between each data node's predicted and fundamental values in the abnormal dimension.

Let $X^{(i)}$ represent the abnormal dimension in i-th iteration, where $X^{(0)}$ represents the original abnormal dimension of the input. Data nodes labelled as true should not be cleaned, i.e $x_t^{(0)}$ is labelled as true, then $x_t^{(0)} = x_t^{(i)}$. The steps of the outlier repair algorithm are as follows:

The second line of the algorithm uses the abnormal dimension data X and the dimension data in the current iteration round $X^{(i)}$ to estimate the $\mathrm{ARX}(p)$ parameters of the model $\phi^{(i)}$ in this paper, the least square algorithm is used to estimate the parameters. The third line of the algorithm mainly calculates the possible repair candidate values of each data node based on the model constructed in the previous step, and the fourth line selects the only repair candidate as the cleaning result of this round from the repair candidate values generated during the last stage according to the principle of minimum modification. Lines 5-7 of the algorithm terminate the algorithm's execution through the maximum number of iterations or convergence conditions. The convergence conditions are as follows:

$$\left|X_j^{(k)} - X_j^{(k+1)}\right| \le \tau, j = 1,2,\dots,n \tag{5}$$

Algorithm 3: outlier repair

Input: original suspicious sequence x, suspicious sequence with annotation information $X^{(0)}$.
Output: sequence after cleaning $X^{(i)}$

```
for $i \leftarrow to\, MaxNumItetations$ do
    $\phi^{(i)} \leftarrow ParameterEstimation\left(X, X^{(i)}\right)$
    $x'^{(i)} \leftarrow CandidateCalculation\left(X, X^{(0)}, \phi^{(i)}\right)$
    $X^{i+1} \leftarrow AssessmentResult\left(X, X^{(k)}, x'^{(i)}\right)$
    if $Convergence\left(X^{(i)}, X^{(i+1)}\right)$ then
        $normal[j] \leftarrow i$
    end
    $i \leftarrow i+1$
endfor
return $X^{(i)}$
```

4 EXPERIMENT

4.1 Experimeal environment and evaluation criteria

In this section, the experimental evaluation will be carried out on the real wireless sensor networkdataset according to the corresponding evaluation standards. The experimental results will be compared with the existing repair algorithms, including the representative EWMA algothm based on smoothing and the SCREEN algorithm based on speed constraint.

Experimental environment

This paper uses Java language to implement each part in the following environment. The processor is 2.21Ghz Intel Core i7, and the memory is 8GB.

Experimental data

The dataset of the Intel laboratory is used in this paper (http://db.ail.mit.edu/labdata/labdata.html). Intel lab data includes time-series data of four attributes collected by 54 sensors: humidity, temperature, illumination and voltage. In this paper, theemperature data of 21 sensors are selected randomly, and some data nodes are intercepted in each dimension. The first 10K data nodes are used as training data; The last 6K data nodes test the algorithm effect. Using the algorithm proposed by the paper[40], a certain proportion of the sequence is randomly selected as the abnormal sequence in the test data. Some data nodes are randomly chosen in each bizarre line to replace. The replacement value is any value between the minimum and maximum values of the arrangement. The replaced data node is considered as the outlier of injection error.

Evaluation criterion

The metrics used in this paper include:

(1) Root Mean Square Error[41]. Order x_{truth} as the truth value of time series, $_{repair}$. as the repaired time series data, the RMS error is as follows:

$$\Delta\left(x_{truth}, x_{repair}\right) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i^{truth} - x_i^{repair}\right)^2} \tag{6}$$

RMS measures the similarity between the repair results and the true values. The smaller the RMS, the closer the repair results are to the true values, the more accurate the repair results are.

(2) Wrong Distance[21]. That is, the distance between the dirty data (observed value) and the real data. Order x_{dirty} as the observed value of time series, the Error Distance is as follows:

$$\Delta_{fault}\left(x_{truth}, x_{dirty}\right) = \sum_{i=1}^{n}(x_i^{dirty} - x_i^{truth})^2 \tag{7}$$

(3) Abnormal Detection Precision Rate P, Abnormal Detection Recall Rate R. The confusion matrix is used to define this metric, as shown in Table 4.1.

Then the definition of P and R are as follows:

$$P = TP / \left(TP + FP\right) \tag{8}$$

	Abnormal detection	**Normal detection**
Actual exception	TP	FN
Actually normal	FP	TN

TABLE 1
Confusion matrix

$$R = TP / \left(TP + FN\right) \tag{9}$$

4.2 Experimental results

This paper randomly assigns three abnormal dimensions in 21 test dimensions and the high correlation threshold is 0.8. The root mean square error results, error distance results and time cost results of various repair algorithms under different abnormal rates (outliers /total data nodes) and different data volumes are provided. Correlation analysis is performed with the test data to obtain the correlation data between the various dimensions according to the algorithm flow before the experiment starts. This dimension correlation data is used in subsequent abnormal dimension detection. The specific experimental results are shown below.

Table 4.2 lists the P and R of the abnormal dimension identification steps on Intel laboratory data. A recall ratio of 1.0 indicates no missing dimension where an anomaly occurred. Meet the expectation of anomaly detection to reduce the missed detection. At the same time, the higher precision shows that the suspicious dimension selected in this step has high accuracy.

Table 4.3 shows the P and R of the outlier detection steps under different outlier rates for 2.5k data. With the increase of the abnormal rate, the P increases gradually and the R decreases. The experimental results meet the expectation of anomaly detection algorithm: false detection is more acceptable than missing detection.

Figure 1 shows the error distance under different anomaly rates for 2.5k data. The convergence threshold of the algorithm is 0.3 and the model parameter of ARX (P) is P = 3. With the increase of abnormal rate, there are more and more outliers in the dataset, so the distance beeen dirty value and real data is larger.

data set	**Precision ratio p**	**Recall ratio R**
Intel lab data	0.72	1.0

TABLE 2
Correlation analysis results

Abnormal rate	15%	25%	40%
Precision ratio p	0.4062	0.5455	0.7018
Recall ratio R	1.0	1.0	0.50

TABLE 3
Detection effect of different abnormal rates

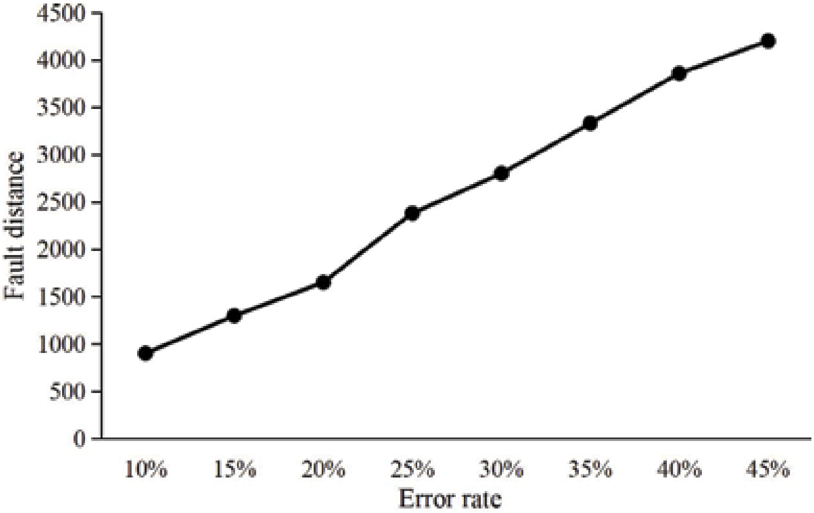

FIGURE 1
The distance between dirty value and actual value under different error rates

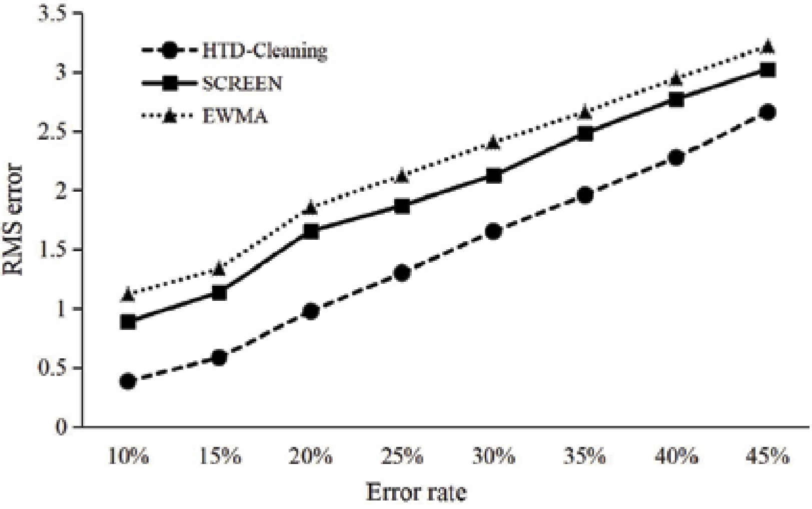

FIGURE 2
Root mean square error of each algorithm under different error rates

Figure 4.2 shows the RMS of each algorithm under different anomaly rates for 2.5k data. The convergence threshold of the algorithm is 0.3 and the model parameter of ARX (P) is P = 3. With the increase of anomaly rate, the RMS errors of the three algorithms are on the rise. Because the algorithm based on smoothing predicts the current time value through the historical value and takes the predicted value as the repair value, a large number of original normal data will be modified, so the RMS error of the EWMA algorithm is the highest. However, the algorithm based on speed constraint only restores the outliers to the boundary values of speed constraints, but it does not have the problem of over modification, so the RMS error of the screen algorithm is slightly lower. Firstly, the HTD-Cleaning algorithm uses speed constraints to detect outliers and uses the ARX model for iterative cleaning. Compared with the constraint boundary value of the screen algorithm, the HTD-Cleaning algorithm can repair outliers more thoroughly. Therefore, with the increase of anomaly rate, the RMS error of the HTD-Cleaning algorithm is always the lowest; that is to say, the repair effect is the best.

Figure 4.3 shows each algorithm's root mean square error under different data volumes when the anomaly rate is 15%. The convergence threshold of the algorithm is 0.3 and the model parameter of ARX (P) is P = 3. Under the fixed anomaly rate, with the increase of data volume, the proportion of normal data is larger, the RMS errors of the three algorithms are gradually reduced. The RMS error of the EWMA algorithm is still the highest due to the over-modified data,. The screen algorithm still has the speed constrained boundary value problem, so the RMS error is slightly lower. The HTD-Cleaning algorithm will automatically label the abnormal dimension data. With the increase of the proportion of normal data, more data labeled as true can participate in the training of the ARX model so that the RMS error decrease trend will be faster than the comparison algorithm.

Figure 4.4 shows the time cost of each algorithm under different anomaly rates with 2.5k data. The convergence threshold of the algorithm is 0.3 and

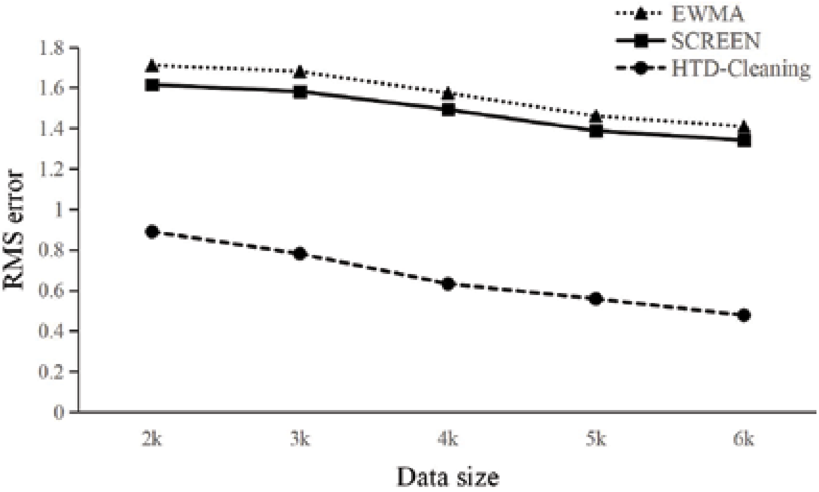

FIGURE 3
Root mean square error of each algorithm under different data volume

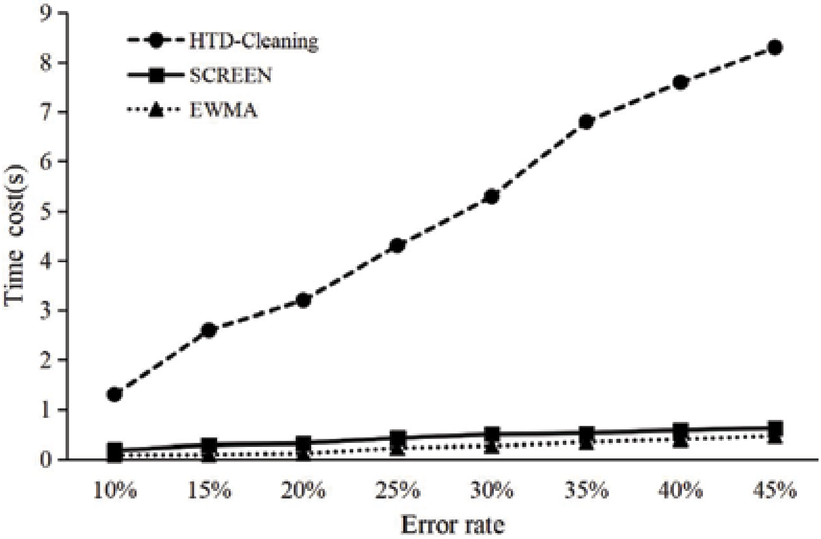

FIGURE 4
Time cost of each algorithm under different error rates

the model parameter of ARX (P) is P = 3. With the increase of anomaly rate, the time cost of algorithm -based smoothing and algorithm -based constraint tends to be flat. EWMA algorithm predicts the current time value by historical value and repairs the current value. Although the abnormal rate gradually increased, the amount of data processed by the EMWA algorithm remains unchanged. Hence, the time cost of the EWMA algorithm tends to be stable and the lowest. The HTD-Cleaning algorithm relies on the automatically labeled information of the abnormal dimension when repairing the outliers. With the increase of the abnormal rate, the proportion of the data marked as true decreases, and the number of rounds of iterative cleaning is bound to increase. The time cost of the HTD-Cleaning algorithm increases dramatically. In contrast, the SCREEN only traverses the abnormal dimension data once, and its time cost is positively correlated with the speed constraint value. Therefore, the time cost of the SCREEN algorithm is slightly higher than that of the EWMA algorithm and much lower than that of the HTD-Cleaning algorithm.

Figure 4.5 shows the time cost of each algorithm under different data volumes when the abnormal rate is 15%. The convergence threshold of the algorithm is 0.3 and the model parameter of ARX (P) is P = 3. EWMA and SCREEN algorithms only need to traverse the abnormal dimension data once. With the increase of data volume, the time cost of the two algorithms increases, but the trend is slow. The SCREEN algorithm detects outliers through speed constraints; the speed constrained boundary nodes are used to

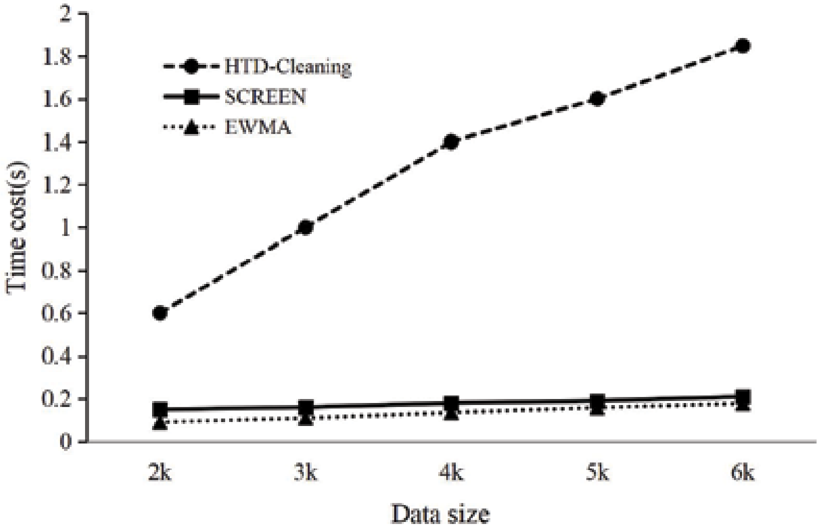

FIGURE 5
Time cost of each algorithm under different data volume

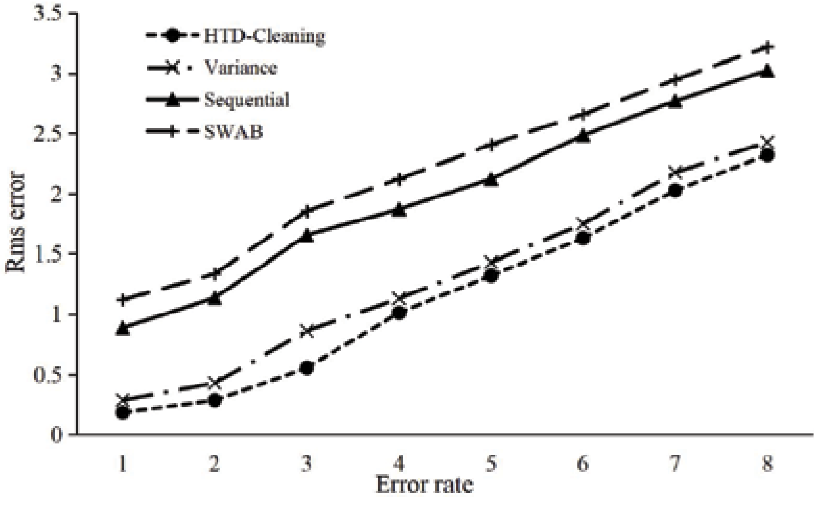

FIGURE 6
Root mean square error of each algorithm under different error rates

repair them. The EWMA uses only the predicted value as the repair value, so the EWMA algorithm has the lowest time cost. HTD-Cleaning algorithm uses iterative cleaning in data cleaning, so the time cost increases rapidly with the increase of data volume.

Figure 4.6 shows the root mean square error of each algorithm under different anomaly rates with 2.5K data, and the convergence threshold of the algorithm is 0.3 and the model parameter of ARX (P) is P = 3. Similar to the trend in Figure 4.2, root mean square error increases gradually with the increase of anomaly rate. The smooth-based SWAB algorithm has the highest RMS due to excessively changing the original correct value and removing the weight setting compared to EWMA algorithm. The Sequential algorithm only considers the limitation of data changes between two adjacent points to clean data, and it ignores the timestamp change. Variance algorithm cleans data by limiting the difference change in the specified window (the variance constraint window in this paper is 10), and it is insensitive to the error value with small deviation from the correct value. As can be seen from the figure, since variance constraint considers data in a window, more data can participate in the repair of abnormal values, and the RMS of Variance algorithm is slightly lower than that of Sequential algorithm. The HTD-Cleaning algorithm detects outliers according to the speed constraint, and then repairs outliers by iterative method, avoiding excessive changes to the original data and repairing outliers thoroughly at the same time.

Figure 4.7 shows the root mean square error of each algorithm under different data amounts when the anomaly rate is 20%, the convergence threshold of the algorithm is specified as 0.3 and the model parameter of ARX (P) is P = 3. Similar to the trend in Figure 4.3, RMS of each algorithm gradually decreases with the increase of data volume. The HTD-Cleaning algorithm will automatically mark abnormal dimension data. As the proportion of normal data increases, more data marked True can participate in the training of ARX model, so the RMS error decrease trend is faster than that of the com-

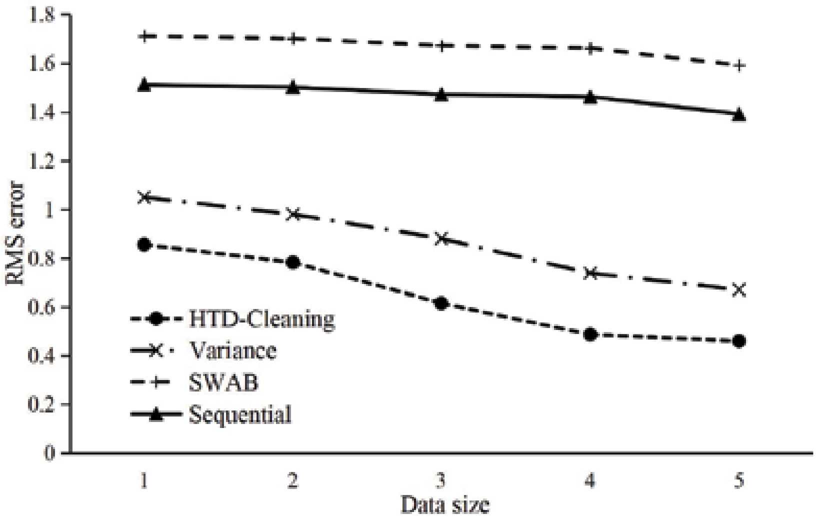

FIGURE 7
Root mean square error of each algorithm under different data volume

parison algorithm. The Variance algorithm also tends to decline faster, because more correct data enables more accurate Variance constraints on data within the same window. The SWAB algorithm always has the problem of excessively changing the correct value. The Sequentail algorithm only calculates numerical changes between adjacent data points, so the RMS of the SWAB algorithm and the Sequential algorithm tends to be flat as the data volume changes.

Figure 4.8 shows the time cost and root mean square error of HTD-Cleaning algorithm with different model order P when the anomaly rate is 15% and the data volume is 3K. The higher the order P, the more historical data will predict the current data node. Due to the iterative strategy and minimum modification principle, the HTD-Cleaning algorithm can obtain lower RMS error even if P = 1; however, the time required for iteration increases rapidly, so too large a P can not bring significant improvement of root mean square error. In other words, the accuracy performance of different model order P is the same, but higher-order will bring more time cost.

Figure 4.9 shows the time cost and root mean square error of the HTD-Cleaning algorithm based on the different convergence thresholds τ when the anomaly rate is 15%, p=3 and the data volume is 3K. If the convergence threshold τ is small, the algorithm needs to run more iterations to converge, and the time cost will increase. However, the RMS error does not increase significantly if the convergence threshold is further reduced (from 0.1 to 0.05), but the time cost increases rapidly. With the increase of the convergence threshold, the time cost is gradually reduced, and the RMS error

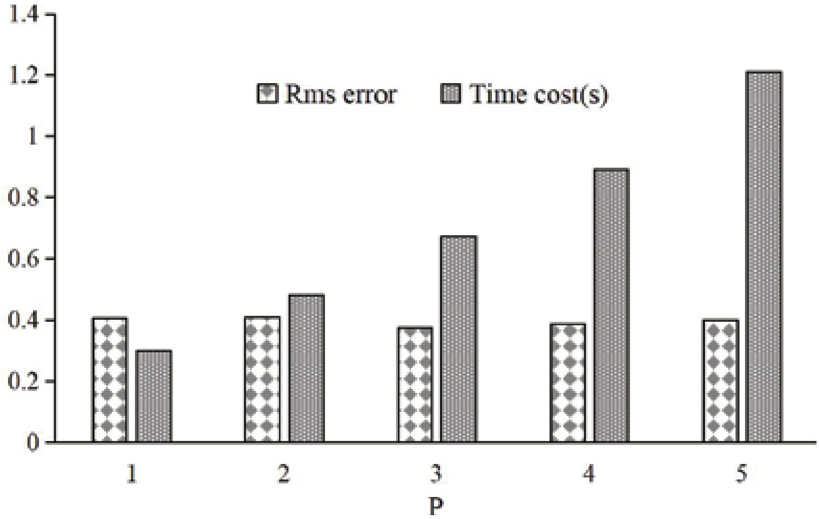

FIGURE 8
Algorithm time cost and root mean square error under d ifferent order P

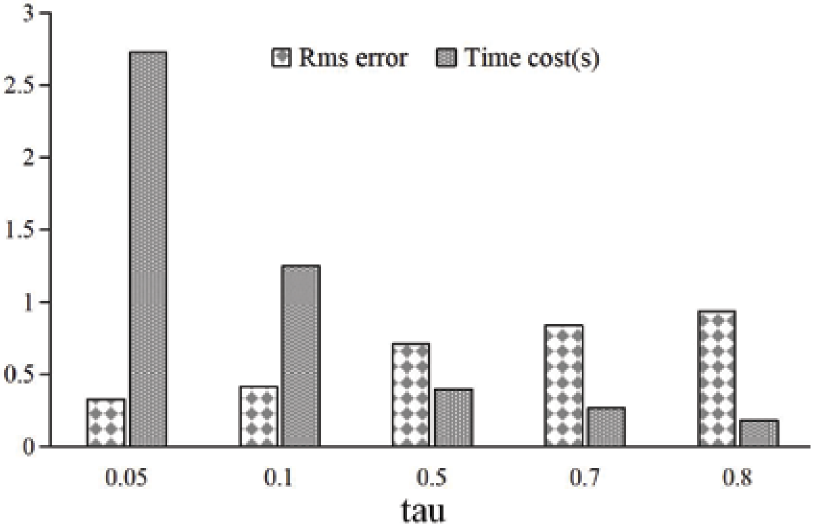

FIGURE 9
Algorithm time cost and root mean square error under different convergence threshold theta

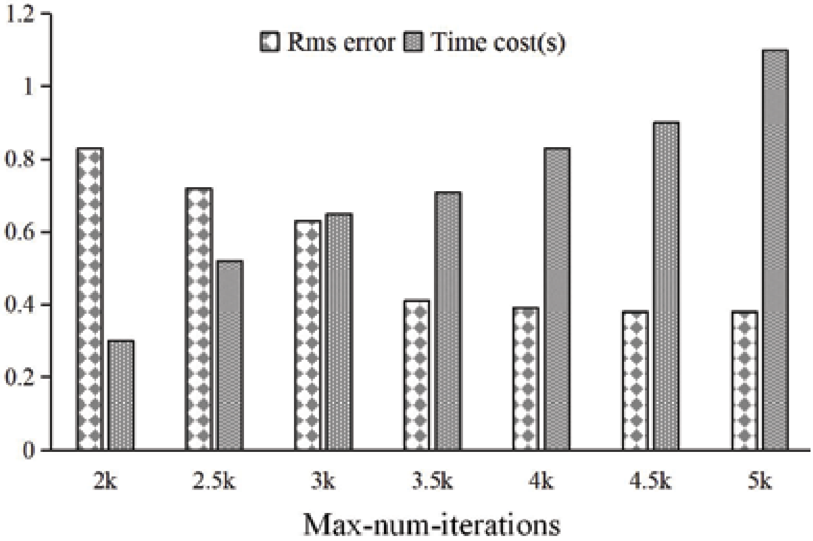

FIGURE 10
Algorithm time cost and root mean square error under different maximum number of iterations

increases, so the convergence threshold τ can be used as a trade-off between time cost and repair effect.

Figure 4.10 shows the time cost and root mean square error of the HTD-Cleaning algorithm under different maximum iterations when the anomaly rate is 15%, p=3 and the amount of data is 2.5K. Theoretical analysis shows that the algorithm converges in two cases, namely, reaching the maximum number of iterations or satisfying Formula (5), but a maximum number of iterations is still needed to reduce the algorithm time. In other words, the algorithm can stop when the maximum number of iterations is reached, and the algorithm does not converge. According to Figure 4.10, the maximum number of iterations of the intermediate size can already obtain a good experimental effect, and the effect at this time is close to the cleaning result of the algorithm convergence. With the increase of the maximum number of iterations, the time cost of the algorithm rises gradually. However, when the algorithm converges, the time cost tends to be stable with the increase of the maximum number of iterations, which further illustrates the necessity of setting the maximum number of iterations.

5 CONCLUSION

Considering that the high-dimensional time series collected by wireless sensor networks often have a specific correlation, and the time series of a single sensor has strong temporal correlation characteristics, this paper proposes an HTD-Cleaning algorithm for wireless sensor networks. Firstly, the abnormal dimension is determined according to prior knowledge, and then anomaly detection is carried out in the abnormal dimension; then, the abnormal dimension is labeled. Finally, the abnormal dimension with labeled information is repaired iteratively according to the principle of minimum modification. In this paper, the proposed algorithm is verified on a specified wireless sensor network dataset. The experimental results show that, compared with other existing cleaning algorithms, the HTD-Cleaning algorithm has the lowest root mean square error under each abnormal rate and data volume and has the best repair effect. At the same time, the time cost of each cleaning algorithm is evaluated. Compared with other cleaning algorithms, which only traverse the data volume once, the proposed algorithm has a higher time cost because of the iterative cleaning.

Although this paper has achieved phased results in WSN data cleaning, due to the complexity of the network itself and many difficulties in data cleaning, the following work is worth further discussion.

Cleaning algorithm design. Since the HTD-Cleaning algorithm adopts the autoregressive model with exogenous variables, the model has obvious directionality. We can consider caching the time series data collected at a specific time, modeling and cleaning the data from the positive and negative aspects of the time sequence.

Build cleaning tools. In addition to providing a core cleaning algorithm, it is also crucial to build WSN data cleaning tools. Building a cleaning tool suitable for WSN datasets can be critical in the next stage, considering the current mainstream relational data cleaning tools.

ACKNOWLEDGEMENTS

The work is partially supported by Grants from the Natural Science Foundation of Zhejiang Province (NO.LY19F020006, NO.LY19F020002), the Key Research and Development Project of Zhejiang Province(NO.2021C03187), theNationalKeyResearchandDevelopmentProgram(NO.2019YFB2102101), and the National Natural Science Foundation of China(NO. 62072146).

REFERENCE

[1] PANDIAN P S, SRINIVASAN S. A Unified Model for Preprocessing and Clustering Technique for Web Usage Mining. Journal of Multiple-Valued Logic & Soft Computing, 2016, 26(3-5): 205-220.

[2] TURKES O, BAYDERE S. Priority-Based Voice Segmentation and Transmission in Quality-Driven Wireless Audio Sensor Networks. Ad Hoc & Sensor Wireless Networks, 2015, 29(1-4): 311-331.

[3] ZHANG J, QIN W, BAO JS, et al. Big Data in Manufacturing Industry. Shanghai: Shanghai Science and Technology Press, 2016. (in Chinese).

[4] SHARSHEMBIEV K, YOO S M, ELMAHDI E. Protocol Misbehavior Analysis using Multivariate Statistical Analysis and Machine Learning in Vehicular Ad Hoc Networks. Ad hoc & Sensor Wireless Networks, 2021, 49(3/4): 247-267.

[5] JI Y, CHAI Y, ZHOU X, et al. Smart intra-query fault tolerance for massive parallel processing databases. Data Science and Engineering, 2020, 5(1): 65-79.

[6] GONZALEZ O B, CHILO J. WSN IoT Ambient Environmental Monitoring System. 5th International Symposium on Smart and Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems. IEEE, 2020: 1-4.

[7] ZHANG W, TAN G Z, DING N. Traffic Information Detection Based on Scattered Sensor Data: Model and Algorithms. Ad hoc & Sensor Wireless Networks, 2013, 18(3/4): 225-240.

[8] KHANI M J, SHIRMOHAMMADI Z. SRCM: An Efficient Method for Energy Consumption Reduction in Wireless Body Area Networks based on Data Similarity. Ad hoc & Sensor Wireless Networks, 2022, 51(1-3):173-187.

[9] BOUBICHE D E, PATHAN A S K, LLORET J, et al. Advanced industrial wireless sensor networks and intelligent IoT. IEEE Communications Magazine, 2018, 56(2): 14-15.

[10] Industrial Big Data Special Unit of Industrial Internet Alliance. Industrial Big Data Technology and Application Practice. Beijing: Publishing House of Electronics Industry, 2017. (in Chinese).

[11] WANG JM. Summary of industrial big data technology. Big Data Research, 2017, 6: 3–14. (in Chinese).

[12] LI J, NI J, WANG AZ. From Big Data to Intelligent Manufacturing. Shanghai: Shanghai Jiaotong University Press, 2017. (in Chinese).

[13] WANG X, WANG CHEN. Time Series Data Cleaning: A Survey. IEEE Access, 2020, 8: 1866-1881.

[14] DAS R, DASH D. A Comprehensive Survey on Mobile Sink-Based Data Gathering Schemes in WSNs. Ad hoc & Sensor Wireless Networks, 2022, 52(1/2):1-43.

[15] White M. Enterprise information portals. The Electronic Library, 2000, 18(5): 354-362.

[16] LI X, DONG X L, LYONS K, et al. Truth finding on the deep web: Is the problem solved?. Proceedings of the VLDB Endowment, 2012, 6(2).

[17] KLEIN A, LEHNER W. Representing data quality in sensor data streaming environments. Journal of Data and Information Quality, 2009, 1(2): 1-28.

[18] DIALLO O, RODRIGUES J J P C, SENE M, et al. Distributed database management techniques for wireless sensor networks. IEEE Transactions on Parallel and Distributed Systems, 2013, 26(2): 604-620.

[19] ZHANG A, SONG S, WANG J, et al. Time series data cleaning: From anomaly detection to anomaly repairing. Proceedings of the VLDB Endowment, 2017, 10(10): 1046-1057.

[20] GAO F, SONG S, WANG J. Time Series Data Cleaning under Multi-speed Constraints. International Journal of Software and Informatics, 2021, 11(1): 29-54.

[21] SONG S, ZHANG A, WANG J, et al. SCREEN: stream data cleaning under speed constraints. Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, 2015, 827-841.

[22] ZAMENI M, GHAFOORI Z, SADRI A, et al. Change point detection for streaming high-dimensional time series. International Conference on Database Systems for Advanced Applications. 2019: 515-519.

[23] LI A G, QIN Z. Dimensionality reduction and similarity search in large time series databases. Chinese Journal of Computers-Chinese Edition, 2005, 28(9): 1467.

[24] ZHANG J, WANG H. Detecting outlying subspaces for high-dimensional data: the new task,algorithms, and performance. Knowledge and information systems, 2006, 10(3): 333-355.

[25] GHOTING A, PARTHASARATHY S, OTEY M E. Fast mining of distance-based outliers in high-dimensional datasets. Data Mining and Knowledge Discovery, 2008, 16(3): 349-364.

[26] KARIM S M A, RANJAN N, SHAH D. A Scalable Approach to Time Series Anomaly Detection & Failure Analysis for Industrial Systems. 2020 10th Annual Computing and Communication Workshop and Conference. IEEE, 2020: 678-683.

[27] DING X O, YU S J, WANG M X, et al. Anomaly detection on industrial time series based on correlation analysis. Journal of Software, 2020, 31(3): 726-747.

[28] DING X, WANG H, SU J, et al. Cleanits: a data cleaning system for industrial time series. Proceedings of the VLDB Endowment, 2019, 12(12): 1786-1789.

[29] EICHMANN P, SOLLEZA F, TATBUL N, et al. Visual exploration of time series anomalies with metro-viz. Proceedings of the 2019 International Conference on Management of Data. 2019: 1901-1904.

[30] ZHOU K, FU C, YANG S. Big data driven smart energy management: From big data to big insights. Renewable and Sustainable Energy Reviews, 2016, 56: 215-225.

[31] DING N, TAN G Z, ZHANG W, et al. Character-Aware Traffic Flow Data Quality AnalysisBased on Cusp Catastrophe Theory and Wireless Sen Network. Ad hoc & Sensor Wireless Networks, 2013, 18(3/4): 277-292.

[32] LV Z, SINGH A K. Big data analysis of Internet of Things system. ACM Transactions on Internet Technology, 2021, 21(2): 1-15.

[33] BOHANNON P, FAN W, FLASTER M, et al. A cost-based model and effective heuristic forrepairing constraints by value modification. Proceedings of the 2005 ACM SIGMOD international conference on Management of data. 2005: 143-154.

[34] DING J, LIU Y, ZHANG L, et al. An anomaly detection approach for multiple monitoring dataseries based on latent correlation probabilistic model. Applied Intelligence, 2016, 44(2): 340-361.

[35] HILL D J, MINSKER B S. Anomaly detection in streaming environmental sensor data: A data-driven modeling approach. Environmental Modelling & Software, 2010, 25(9): 1014-1022.

[36] ANAVA O, HAZAN E, MANNOR S, et al. Online learning for time series prediction. Proceedings of the 26th Annual Conference on Learning Theory. 2013, 172-184.

[37] GAO C, CHEN Y, WANG Z, et al. Anomaly detection frameworks for outlier and pattern anomaly of time series in wireless sensor networks. 2020 International Conference on Networking and Network Applications. IEEE, 2020: 229-232.

[38] BROWN R G. Smoothing, forecasting and prediction of discrete time series. Courier Corporation, 2004.

[39] VELASCO F A, PALOMARES J M, OLIVARES J. Lightweight method of shuffling overlapped data-blocks for data integrity and security in WSNs. Computer Networks, 2021, 199: 108470.

[40] ASHWINI U, KALAIVANI K, ULAGAPRIYA K, et al. Time Series Analysis based Tamilnadu Monsoon Rainfall Prediction using Seasonal ARIMA. 2021 6th International Conference onInventive Computation Technologies. IEEE, 2021: 1293-1297.

[41] PARK G, RUTHERFORD A C, SOHN H, et al. An outlier analysis framework for impedance-based structural health monitoring. Journal of Sound and Vibration, 2005, 286(1-2): 229-250.