IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 6, 2019, accepted July 7, 2019, date of publication July 18, 2019, date of current version August 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2929872

Dealing With Alarms in Optical Networks Using an Intelligent System

DANSHI WANG[1], LIQI LOU[1], MIN ZHANG[1], ANTHONY C. BOUCOUVALAS[2], (Fellow IEEE), CHUNYU ZHANG[1], AND XUETIAN HUANG[3]

[1]State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Department of Informatics and Telecommunications, University of Peloponnese, 22100 Tripoli, Greece
[3]China Telecom Corporation, Beijing 100015, China

Corresponding author: Min Zhang (mzhang@bupt.edu.cn)

This work was supported in part by the NSFC Project under Grant 61705016, in part by the National Key Research and Development Program of China under Grant 2016YFB0901200, and in part by the support from the Fundamental Research Funds for the Central Universities under Grant 2019RC12.

ABSTRACT Millions of alarms in the optical layer may appear in optical transport networks every month, which brings great challenges to network operation, administration and maintenance. In this paper, we deal with this problem and propose a method of alarm pre-processing and correlation analysis for this network. During the alarm pre-processing, we use the method of combined time series segmentation and time sliding window to extract the alarm transactions, and then we use the algorithm of combined K-means and back propagation neural network to evaluate the alarm importance quantitatively. During the alarm correlation analysis, we modify a classic rule mining algorithm, i.e., *Apriori* algorithm, into a Weighted *Apriori* to find the high-frequency chain alarm sets among the alarm transactions. Through the actual alarm data from the record in the optical layer of a provincial backbone of China Telecom, we conducted experiments and the results show that our method is able to perform effectively the alarm compressing, alarm correlating, and chain alarm mining. By parameter adjustment, the alarm compression rate is able to vary from 60% to 90% and the average fidelity of chain alarm mining keeps around 84%. The results show our approach and method is promising for trivial alarm identifying, chain alarm mining, and root fault locating in existing optical networks.

INDEX TERMS Alarm pre-processing, K-means, back propagation neural network, alarm compression, alarm correlation analyzing, optical network.

I. INTRODUCTION

As the scale of optical transport network (OTN) expands, the number of alarms in optical layer may reach over one million within only one week, which brings great trouble to the network operation, administration, and maintenance. For example, if alarms appear due to an overtime service delay or an extra-high bit-error-rate (BER), the network administrator has to judge the fault location promptly and correctly in order to repair the fault in time. However, the situation in actual network is very complicated. A fault is often related to a large number of alarms, including many redundant alarms. Meanwhile, there exist lots of false alarms corresponding to no fault. Therefore, it is difficult for people to find useful information promptly from such a large number of alarms. If these alarms can be compressed reasonably and automatically, the alarm analysis will be much easier, which will help the network administrator to find the root cause of the faults and restore the OTN in time. Therefore, the alarm compression has become an urgent and also tough problem in optical networking, and the premise of alarm compression is alarm correlation analysis. As an important part of the network fault management, the alarm correlation analysis is helpful for deleting redundant alarms, predicting chain alarms, and locating faults. Traditional alarm correlation analysis can find the alarms associations to a certain extent and depend mainly on the expert systems or even manual operation of the experienced staff. With the growing number of optical links and OTN systems, it is increasingly difficult for experts to keep up with the rapid changes in the network and then discover truly useful knowledge from the alarms.

Most of the existing studies are based on association rule mining, correlation coefficient, or mutual information to analyze the alarm correlation [1]–[4]. Meanwhile, alarm

The associate editor coordinating the review of this manuscript and approving it for publication was Tianhua Xu.

association analysis based on association rule mining has been widely concerned due to its advantages of compressing alarm volume and finding high-frequency chain alarms. Typical rule mining methods are Apriori-like algorithms [5]–[7], which locate frequent alarm transactions by repeatedly scanning the database and present them as rules. However, there are few studies or reports on alarm analysis for optical networks and the existing methods are not very suitable for optical networks. The alarms in OTNs rise not only from the optical transport plane but also the service plane and the control plane, so that the large amount of alarms from different planes often interact with each other. As a result, if the association rules are directly mined from the original alarms, the performance of the algorithm will be degraded significantly. Moreover, the original alarms from actual optical networks always suffer from several problems (i.e., information redundant, time asynchronous, and ambiguous importance of alarm attributes, etc.), which are detrimental to alarm analysis and compression. Therefore, alarm pre-processing is required.

The common methods of quantitative evaluation of the alarm importance rely on experienced network experts in determining the relative importance of alarms [8]–[10]. Different network experts or researchers may derive different weights for the alarms in the same network. However, the alarm number rises to tens of thousands and also the network is becoming more and more dynamic. It becomes impossible to determine the relative importance of all alarms accurately by manual operation or experts alone. To address the problem of a huge number of alarms with various attributes and of uncertain importance in the dynamic optical network, an effective and objective method of alarm importance evaluation is necessary. In an optical network, the alarm usually contains many attributes and each attribute with a wide range of values makes the alarm importance ambiguous. Therefore, it is necessary to give weights to these attributes and thus evaluate the alarm importance quantitatively. Recently, machine learning is playing an increasingly important role in optical communication research, and has been applied in diverse areas such as predicting equipment failure in optical network [11], reducing nonlinear phase noise [12], [13], compensating physical impairments [14], monitoring optical performance [15], [16], adaptive nonlinear decision at the receivers [17], adaptive demodulator [18], and traffic-aware bandwidth assignment [19].

In this paper, we propose the TKBW method for alarm pre-processing and alarm association rule mining. The time required for different faults to trigger a series of chain alarms is different and, therefore, we use the time sliding window (TT) to divide the original alarms into alarm sequences and thus extract the alarm transactions. Meanwhile, the alarm synchronization and redundancy removal are performed in each of alarm sequence. To address the problem of a huge number of alarms with various attributes and uncertain importance, in this paper we use KB method to give the weights to the Importance on Alarm Attributes (IAAs). After the alarm pre-processing by combined K-means and Back propagation neural network (TT-KB) method, we modified the traditional Apriori algorithm as W-Apriori algorithm to find out the association rules between alarms, and facilitate the alarm compression and fault location.

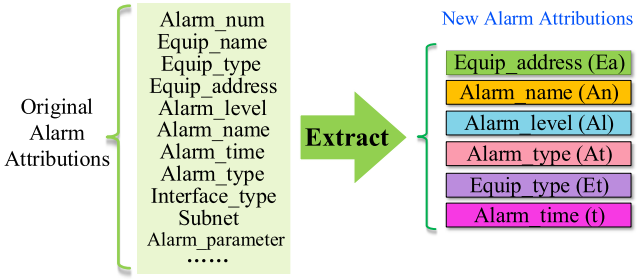

FIGURE 1. Extraction of original alarm attributes.

The rest of the paper is organized as follows: In Section II, we discuss the principles of the alarm analysis scheme, including the TT-KB method for alarm pre-processing, the W-Apriori method for alarm correlation, and their implementation. Section III shows the experimental results and discussion. Finally, we draw conclusions in Section IV.

II. PRINCIPLE OF ALARM ANALYSIS SCHEME

The proposed TKBW method is mainly divided into two parts: alarm pre-processing and alarm correlation analysis.

A. TT-KB BASED ALARM PRE-PROCESSING

Given its present huge scale in trans-provincial backbones and metropolitans networks, there are a large number of original alarms in the existing optical network. Moreover, the original alarm data are facing the problems of information redundant, time asynchronous, and the unclear IAAs. We use the TT method to divide all the original alarms into alarm sequences. Then the alarm synchronization, redundancy removal, and alarm transaction extraction are performed for each alarm sequence via a time sliding window. But there are many alarm attributes and their IAAs are unknown, so that we use the KB method to evaluate the IAAs quantitatively. Firstly, the alarm attributes are quantized, and then the K-means algorithm is used to classify the alarms. Then, the Back propagation neural network (BP-NN) algorithm is used to infer each IAA weight.

According to the network management logs in existing provincial OTN, the original alarms usually contain the following attributes: equipment name, equipment type, equipment address, network element type, alarm level, alarm name, alarm type, alarm time, subnet, and so on. Among these attributes, the equipment name can be determined by the equipment address. The network element type and its subnet are unified within a certain network segment. Therefore, as shown in Fig. 1, to start with, we extract the six attributes, the equipment address (*Ea*), the alarm name (*An*), the alarm level (*Al*), the alarm type (*At*), the equipment type (*Et*) and the alarm time (t) to form a new alarm, which is marked as a 6-tuple $\{Ea, An, Al, At, Et, t\}$.

1) TT METHOD OF ALARM TRANSACTION EXTRACTION

In the existing OTN, the number of alarms for different faults is different and the time required to form an alarm sequence is also different. Therefore, during time-segmenting of a large number of alarms, we have to ensure that a complete alarm sequence is collected. Otherwise it will affect the subsequent alarm correlation analysis. Here, we calculate the time series similarity to implement the time segmenting, which makes the alarm similarity in the same time period maximum while the alarm similarity in different time periods minimal.

According to a direct calculating method of the time series similarity [20], the time interval is used as the criterion for the similarity measure among alarms, x, y as given by Eq. (1),

$$dist(x, y) = \sqrt{\sum (x - y)^2} \tag{1}$$

Thus, the intra-segment similarity function SI(t) is defined as the sum of the squares of the time intervals from each moment to the midpoint of the time period, as follows,

$$\mathrm{SI}(t) = \sum_{i=1}^{k} I(t_i) = \sum_{i=1}^{k} \sum_{t \in c_i} dist(t, c_i) \tag{2}$$

where c_i represents the midpoint of the time period t_i.

The inter-segment similarity function SO(t) is defined as the sum of the time intervals between the midpoint of each time period, as follows

$$\mathrm{SO}(t) = \sum_{1 \le \mathrm{i} \le j \le \mathrm{k}} \mathrm{dist}(c_\mathrm{i}, c_\mathrm{j}) \tag{3}$$

Then the sum of the squared error (*SSE*) is adopted as the objective function and also the index to measure the division of the time window, as given by Eq. (4):

$$SSE = \sum_{i=1}^{k} \sum_{t \in c_i} |t, c_\mathrm{i}|^2 \tag{4}$$

where the optimal result is to obtain the minimal SSE. The main flow of the algorithm is attached as Algorithm I in Appendix.

After the time segmenting, we use the time sliding window method [21] to perform time synchronization, redundancy removal, and alarm transaction extraction for the alarms in each time period. Time synchronization means that the alarms appearing in the same time window are regarded as concurrent alarms and are to be extracted into the same alarm transaction. In addition, if an alarm appears several times in a short interval, it is recorded only once in the same time window in order to eliminate redundant alarms. The alarm transaction refers to the collection of alarms appearing in the same time period.

Regarding the time sliding window method, we give the following definitions:

Definition 1: Time window. The given time window width is V, and the time window is used to slide from the beginning of a time period until the end of the time period.

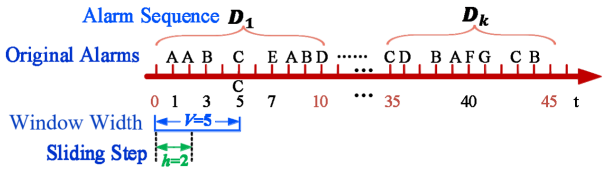

FIGURE 2. Example of alarm transactions extraction using the time sliding window.

TABLE 1. Example of the alarm transaction database TD1.

Transaction Name	Items
T1	A(1), B(3), C(5)
T2	B(3), C(5), E(7)
T3	C(5), E(7), A(8), B(9)
T4	E(7), A(8), B(9)
T5	B(9), D(10)

Definition 2: The window slide step h, which is the length of each movement of the time window.

Figure 2 shows an example that the total alarms {*A, A, B, C,*...,*C, B*} are divided into several alarm sequences (e.g., D_1,D_2,...,D_k) according to the time series similarity calculation. Then given the window width V of 5 and the sliding step size h of 2, the time sliding window method is used to perform alarm transaction extraction for each alarm sequence. If for example alarms B(23) and B(27) occur in the same window, the alarm B is recorded only once. The time when the alarm occurs is recorded as 23s and 27s, which is expressed as B(23,27):1. Similarly if alarms A(28) and A(28) are reported multiple times at the same time, alarm A is recorded once. The alarm occurs for 28s, which is expressed as A(28):1. Table 1 gives an example of the alarm transactions extracted by the time period alarm sequence. The alarm transaction refers to the set of alarms collected in a given time window, such as {A, B, C}, {B, C, E}, {C, E, A, B}, {E, A, B}, {B, D}, and we treat the total alarm transactions as an alarm transaction database TD1 = {{A,B,C}, {B,C,E}, {C,E, A, B}, {E, A, B}, {B, D}}.

2) QUANTITATIVE EVALUATION OF ALARM IMPORTANCE

To evaluate the alarm importance quantitatively, it is necessary to assign IAA weights (Fig.4).

We apply the KB method where the alarm attributes are taken as the input and the alarms with similar attributes are classified into the same class by the K-means algorithm. Then both the alarm attributes and the classifications are taken as the input of the BP-NN to train the connection weights between the neurons, so that the connection weights map the information of the alarm attributes, and thus obtain IAA weights.

According to the advice of experienced network administrators and by observation of the alarm sets and the fault sets in the actual network logs, we select three alarm attributes that are most closely related to the faults as the initial sample

input, i.e. $A = \{Al, At, Et\}$ in Fig.1. Each attribute in the collection has several values that indicate the relative importance. Here we use the K-means algorithm to automatically classify similar alarm samples, and get the comprehensive alarm classifications after fully considering these three alarm attributes.

We rewrite Eq. (1) as Eq. (5), where x_i and x_j represent the attribute values, and D represents the number of attributes.

$$dist(x_i, x_j) = \sqrt{\sum_{d=1}^{D} (x_{i,d} - x_{j,d})^2} \tag{5}$$

All alarms are divided into K classes, denoted as $\mathbf{C_1}$, $\mathbf{C_2}$, $\ldots\ldots\mathbf{C_k}$. Then, we take $A = \{Al, At, Et\}$ as the input and the K alarm classes as the labels to train a stable BP-NN model owing to the strong self-learning and fitting ability of the neural network. Thus, via the method in [22], the connection weights of neurons are mapped to the IAA weights, as given by Eq. (6):

$$Feature_{\mathbf{X}} = \sum\nolimits_{\mathbf{Y}} Hidden_{\mathbf{XY}} \tag{6}$$

where $\mathbf{X}$ is the weight matrix between the input layer neurons and the hidden layer neurons, and $\mathbf{Y}$ is the weight matrix between the output layer neurons and the hidden layer neurons.

Finally, we give the score of the alarm importance as Eq. (7), where N is the number of inputs.

$$W = \frac{1}{N}\left(Feature_1 \times Al + Feature_2 \times At + Feature_3 \times Et\right) \tag{7}$$

B. W-APRIORI BASED ALARM CORRELATION ANALYSIS

The score of the alarm importance is helpful for setting a threshold and thus discard those trivial alarms and false alarms. For the remaining alarms, correlation analysis is still needed to obtain chain-alarms.

The existing correlation analysis method is not designed for optical networks and they do not consider the difference in alarm importance. For example, the typical Apriori algorithm treats different alarms as equal, and there is no difference in the alarm attributes. However, in an actual optical network, an alarm usually consists of many attributes, and different attribute combinations indicates different alarm severities.

When mining the alarm correlation, we improve the Apriori algorithm as W-Apriori, in which the score of alarm importance is used as the weight.

Given a database with a set of alarm transactions $\mathbf{D} = \{\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_m\}$, where $\mathbf{T}_j (j = 1,2,\ldots,m)$ is the set of alarms collected within time window. $\mathbf{I} = \{i_1, i_2, \ldots, i_n\}$ is the set of all alarms in the database, and each alarm transaction set $\mathbf{T_j}$ is a subset of $\mathbf{I}$ (i.e. $\mathbf{T}_j \subset \mathbf{I}$). The set of alarm scores in the corresponding alarm set I can be expressed as $\mathbf{W} = \{W_1, W_2, \ldots, W_n\}$ $(0 \le W_j \le 1, j = 1,2,\ldots,n)$, and W_j is the score of the alarm i_j.

In traditional Apriori, the association rule is the expression of the form $\mathbf{X} \Rightarrow \mathbf{Y}$, where $\mathbf{X}$ and $\mathbf{Y}$ are disjoint item sets (i.e. $\mathbf{X} \cap \mathbf{Y} = \emptyset$). The strength of the association rule is measured by its support and confidence. The support determination rules can be used for the frequency of the given data set, and the confidence determines how frequently $\mathbf{Y}$ appears in transactions that contain $\mathbf{X}$. The W-Apriori algorithm redefines the above two.

Definition 3: Weighted Support. The traditional support of pattern $\mathbf{X}(\mathbf{X} \subseteq \mathbf{I})$ is denoted as $Support(\mathbf{X}) = \text{Count}(\mathbf{X})/|\mathbf{D}|$, where $\text{Count}(\mathbf{X}) = |\mathbf{T}_j | \mathbf{X} \in \mathbf{T}_j, \mathbf{T}_j \in \mathbf{D}|$ is the number of transactions for which item set $\mathbf{X}$ appears in $\mathbf{D}$, and $|\mathbf{D}|$ is the total number of transactions in $\mathbf{D}$. Then the weighted support ($wSupport$) of $\mathbf{X}$ is defined as Eq. (8), where n is the number of items in $\mathbf{X}$. Meanwhile, $minwSup$ is used to represent the minimum weighted support threshold, which is used to evaluate the minimum limit of the transaction frequency.

$$wSupport(\mathbf{X}) = Support(\mathbf{X}) \times \left(\sum\nolimits_{I_j \in \mathbf{X}, j=1}^{n} W_j\right) \tag{8}$$

Definition 4: Weighted Confidence. The traditional confidence of a rule $\mathbf{X} \Rightarrow \mathbf{Y}$ is $Confidence(\mathbf{X} \Rightarrow \mathbf{Y}) = Support(\mathbf{X} \cup \mathbf{Y}) / Support(\mathbf{X})$. Then the weighted confidence ($wConfidence$) of a rule $\mathbf{X} \Rightarrow \mathbf{Y}$ is defined as Eq. (9), where n is the number of items containing the union of $\mathbf{X}$ and $\mathbf{Y}$, and m is the number of items in $\mathbf{X}$. Meanwhile, $minwConf$ is used to represent the minimum weighted confidence threshold, which is used to evaluate the minimum limit of the transaction association.

$$wConfidence(\mathbf{X} \Rightarrow \mathbf{Y}) = \frac{Support(\mathbf{X} \cup \mathbf{Y}) \times \left(\sum_{I_j \in (\mathbf{X} \cup \mathbf{Y}), j=1}^{n} W_j\right)}{Support(\mathbf{X}) \times \left(\sum_{I_j \in \mathbf{X}, k=1}^{m} W_k\right)} \tag{9}$$

Definition 5: Weighted Frequent itemsets. Given a database $\mathbf{D}$ and a weighted support threshold $minwSup$, if a pattern $\mathbf{X}$ satisfies: $wSupport(\mathbf{X}) \ge minwSup$. Then $\mathbf{X}$ is the weighted frequent itemsets.

Therefore, the purpose of the W-Apriori algorithm is to find all the association rules where the $wSupport$ and the $wConfidence$ satisfy the conditions $wSupport \ge minwSup$ and $wConfidence \ge minwConf$ in the given alarm database $\mathbf{D}$ respectively. The description of the W-Apriori algorithm is given as Algorithm II (seen in Appendix).

Figure 3 is an example to illustrate the implementation of the W-Apriori algorithm. We use the alarm transaction database $\mathbf{D}_1$ given in Table I, which contains 5 alarm transaction sets: {A, B, C}, {B, C, E}, {C, E, A, B}, {E, A, B}, {B, D}, where $minwSup = 50\%$ is used as the minimum support threshold. The W-Apriori algorithm uses an iterative strategy of layer-by-layer search: in the k-th cycle, frequent k-itemsets are generated through a combination of the transaction database and candidate k-itemsets, and then a new candidate $(k+1)$-itemsets are generated based on the k-itemsets. And so on, the algorithm stops until the maximum itemset of a cycle is empty. Finally, we filter out high-frequency chain-alarm {A, B}, {B, C}, and {B, E}.

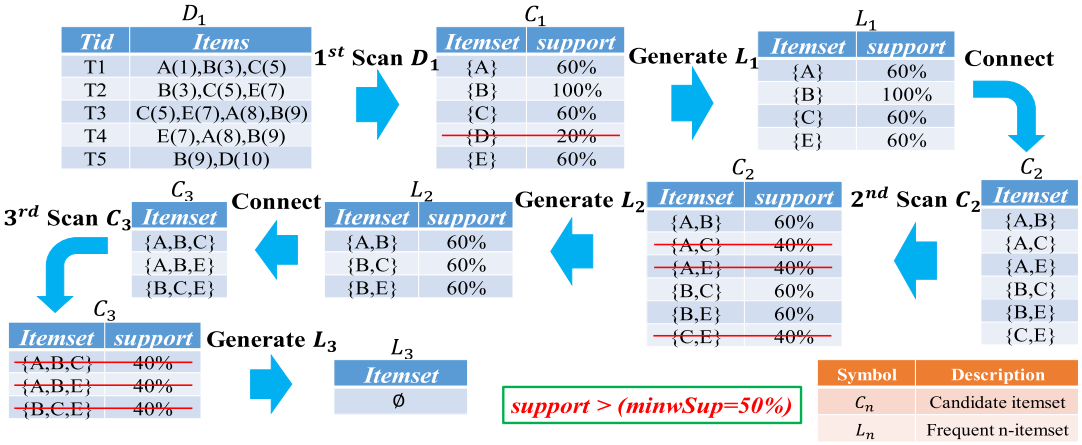

FIGURE 3. Execution process of W-Apriori algorithm.

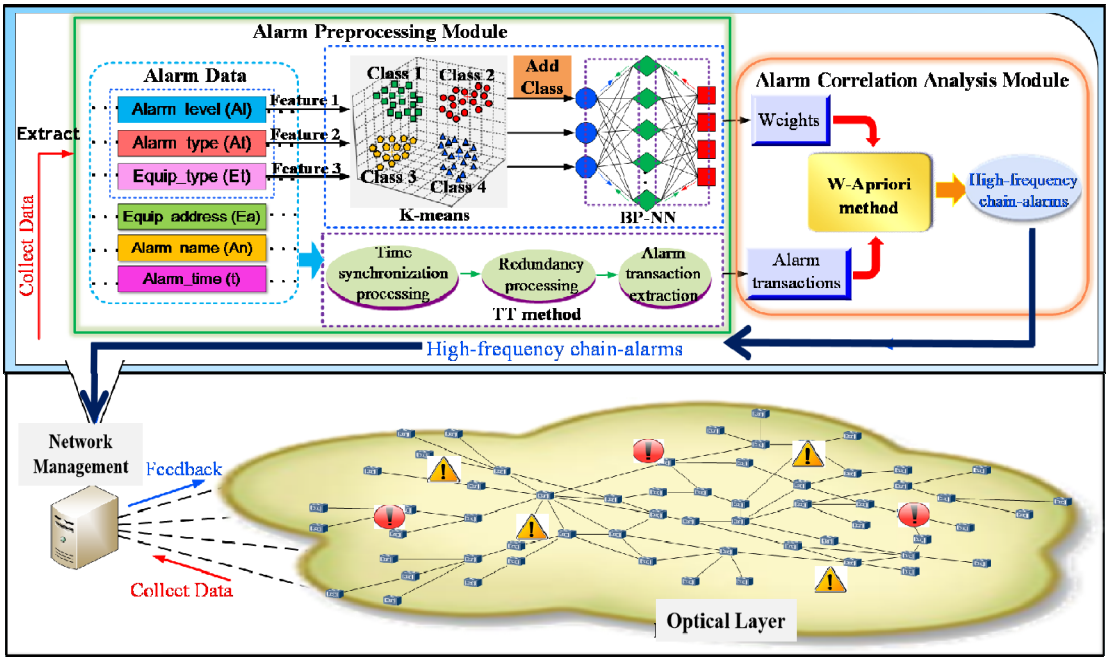

FIGURE 4. Application scenario of the alarm analysis scheme.

C. IMPLEMENTATION OF TKBW ALGORITHM

The application scenario of the alarm analysis scheme is shown in Fig. 4. In the OTN network, the alarm data in optical layer collected through the network management system or the controller in software defined optical network (SDON), and then the TKBW method is used for alarm analysis.

Figure 5 illustrate the overall principle of the TKBW method. TKBW consists of two modules, i.e. an alarm preprocessing module and an alarm correlation analysis module. The alarm pre-processing module extracts the alarm transaction and evaluates the alarm weight and converts the alarm data into alarm transactions suitable for correlation analysis. The alarm correlation analysis module is used to mine the association rules from the alarm transactions, and thus finds out the chain alarms. The procedure of the TKBW method is shown as follows.

Step 1: Collecting the alarms and select the useful alarm attributes;

Step 2: Extracting the alarm transactions by using TT method;

Step 3: Scoring the alarm importance by using KB method;

Step 4: Analyzing the alarm correlation with W-Apriori algorithm and find out the chain alarms.

III. EXPERIMENTS AND DISCUSSIONS

Experiments were conducted with the alarm data from the network log of optical layer equipment in a provincial backbone of China Telecom that contains 441 OTN nodes. We collected 5,100,000 original alarms within 30 days. The proposed TKBW method was developed via Python and implemented on a computer with Windows 7 operating system, Intel(R) Core (TM) processor i5-4345 with 2G main memory.

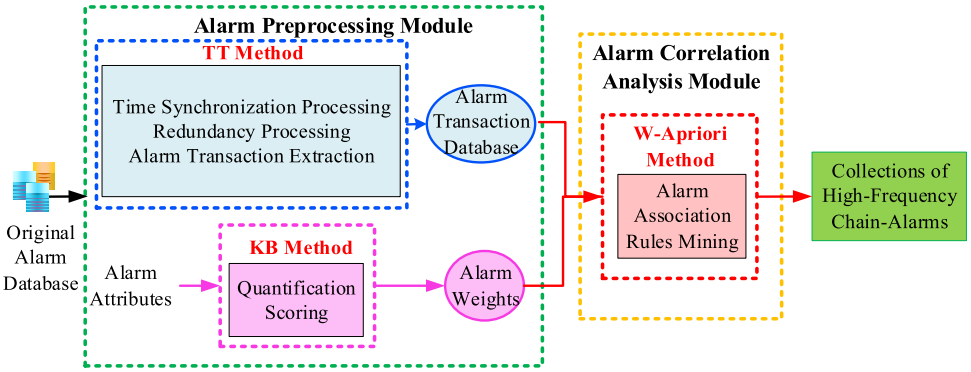

FIGURE 5. Schematic diagram of TKBW method.

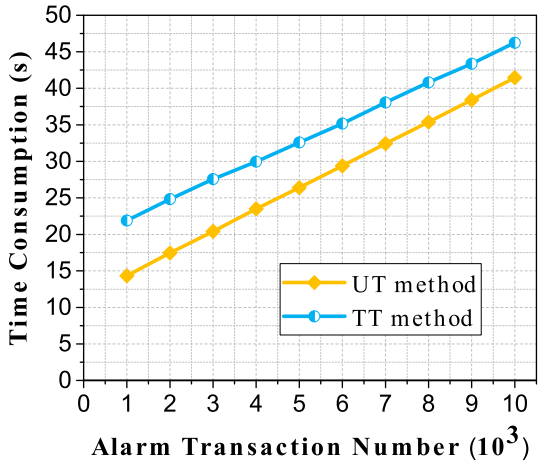

FIGURE 6. Comparison of the time consumptions in extracting alarm transactions by TT method and UT method.

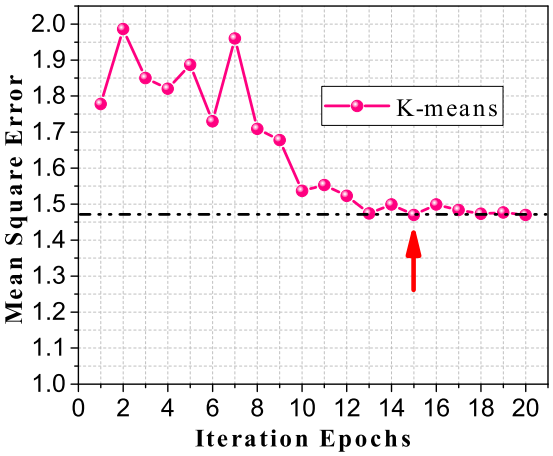

FIGURE 7. Mean square error varies with the iteration epoch of K-means.

A. ALARM TRANSACTION EXTRACTION

First, the original alarm data of the first 20 days were used as the test data, and Step 1 and Step 2 in TKBW procedure were performed to extract the alarm transaction. In the actual OTN, the operator generally uses the alarm data extracted every 15 minutes, which is not extracted in real time. Therefore, we divided all the alarms per day by intervals of 15 minutes and thus got averagely 90 alarm transaction databases per day, namely $D1$, $D2$, ... and $D90$. Then, a time sliding window processing was performed for each alarm transaction database to extract the alarm transactions. Here we set the window width V as 5 minutes and the sliding step h as 2 minutes.

We compared the time consumption of alarm transaction extraction by the TT method and the uniform time (UT) window method. It can be seen from Fig. 6 that the more complex TT method takes slightly longer time than the UT method does. And as the alarm transaction number increases, the difference fades away. We still prefer to use the TT method because it fully considers the unevenness of the distribution of alarm data, and also can remove the isolated points and abnormal points as invalid alarms.

B. QUANTITATIVE ANALYSIS OF ALARM IMPORTANCE

In step 3, according to the advice of experienced network administrators, we selected three alarm attributes that are the most closely related to the faults, namely, alarm level Al, alarm type At, and alarm equipment type Et. These three attributes were initialized. For example, for alarm level AL, we initialized emergency as 1, importance as 2, secondary as 3 and prompt as 4. Then, the alarm transactions with initialized attributes were taken as the input of K-means algorithm, where the alarms with similar attributes were divided into the same class. As shown in Fig. 7, after about 15 iterations, the mean squared error of classification converged to 1.5, indicating that the K-means algorithm is trained.

After the K-means classification, all the alarms were automatically classified into four classes, recorded as $C1$, $C2$, $C3$ and $C4$, respectively. Then, this classification was used as the input label of BP-NN, where the connection weights of all BP-NN neurons were obtained through BP-NN feedback learning. Here, we considered the effect and complexity of the algorithms, and set numbers of the input, output and hidden layers of BP-NN as 3, 4 and 10 respectively, and set the learning rate as 0.01. Fig. 8 shows the connection weight between the input neurons and the first hidden layer neurons during BP-NN training, which changes with the number of

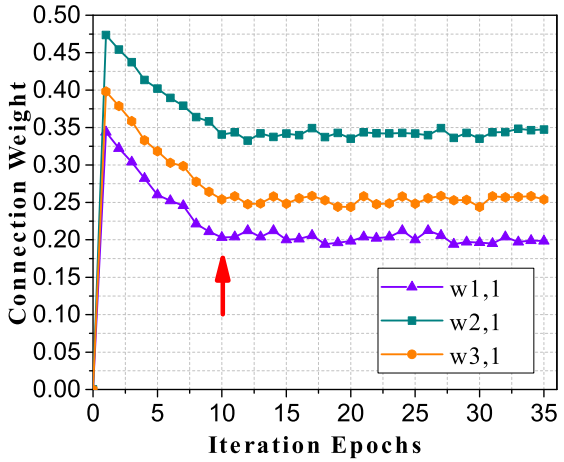

FIGURE 8. Connection weights vary with the iteration epoch of BP-NN.

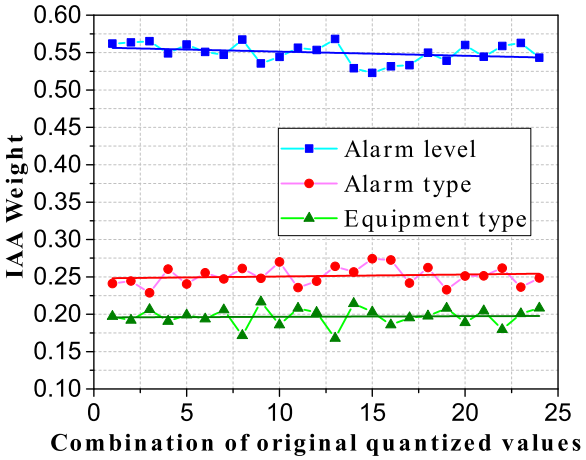

FIGURE 9. IAA weights for different quantitative combinations.

training iterations. It can be seen that after 10 epochs of iteration, the connection weight tends to be convergent, which shows that it is feasible to score the alarms and their attributes.

In the above, we marked the output classifications ($C1$ to $C4$) by K-means as four different values, but these values are in fact only tags of the classes and they are not necessarily the only representations. Therefore, there are 24 kinds of combinations when assigning 1∼4 to $C1\sim C4$ respectively. We repeated the above experiment with different combinations of the initial values of $C1\sim C4$ and then calculated the IAA weights given by Eq. (6). The results are shown in Fig. 9. It can be seen that, although with different initial values, via the classification of K-means and BP-NN, the IAA weights are able to be stabilized at around 0.55, 0.25 and 0.20 respectively. It indicates that the combined algorithm of K-means and BP-NN works independent of the initially quantized values. Therefore, the idea of giving quantitative evaluation to the alarms and the abstract alarm attributes works. Then, the scores of alarms can be calculated according to Eq. (7), and the *wSupport and wConfidence* of alarms can be calculated according to Eq. (8) and Eq. (9). Thereafter, as illustrated by the C_1 step in Fig.3, we obtained the

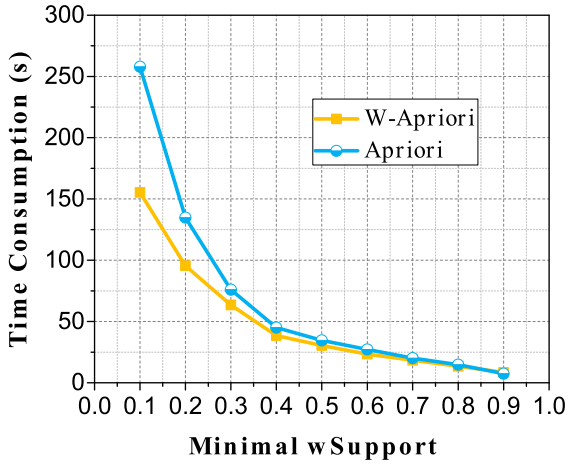

FIGURE 10. Time consumption varies with minwSups.

wSupport for each alarm and then deleted those alarms whose *wSupport* s are less than the *minwSup*. Here, the *minwSup* acts as a threshold for the preliminary alarm compression and this threshold should be assigned according to the requirements of the actual network management.

C. ALARM CORRELATION ANALYSIS AND ALARM COMPRESSION

After the preliminary alarm compression, we analyzed the correlation among the remaining alarms and explored ways to compress the alarms in terms of alarm compression rate and fidelity.

First, the time consumptions of the traditional Apriori and the modified W-Apriori were compared, as shown in Fig. 10, where the number of alarm transactions sets is 100,000 and *minwSup* changes from 0.1 to 0.6. As the *minwSup* increases, the time consumptions of both algorithms decrease gradually. The main reason is that when the threshold *minwSup* is small, more alarm transaction sets need to be processed. In addition, the W-Apriori takes less time than the Apriori does.

Next, we tested the effects of alarm number on the performance of both algorithms. Fig. 11 shows the time consumptions for mining 10,000 to 100,000 alarm transactions by both algorithms, with a fixed *minwSup* of 0.2. As the alarm number increases, the time consumption of both algorithms become longer, but the W-Apriori outperforms the Apriori gradually. With larger alarm number, the advantages of W-Apriori becomes more obvious.

Both Fig. 10 and Fig. 11 show that the modified W-Apriori outperforms W-Apriori in terms of time consumption, since the W-Apriori considers the alarm importance and thus removes a large number of redundant alarms (e.g., false alarms).

Figure 12 presents the comparison of the compressed alarm collection number by both algorithms, with various *minwSup* s and with a fixed alarm transaction number of 10,000. To some extent, the same *minwSup* means that the same compression fidelity can be obtained. Fig. 12 indicated

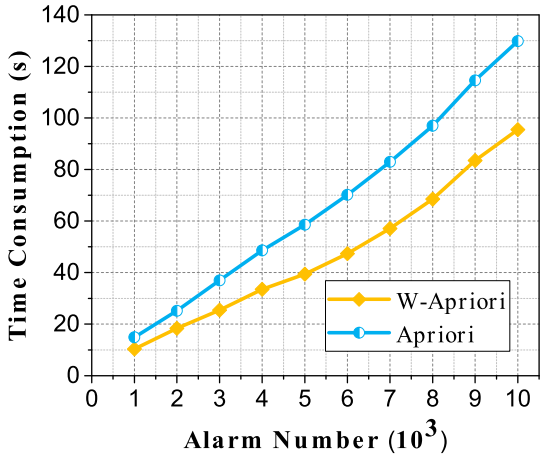

FIGURE 11. Time consumption varies with alarm number.

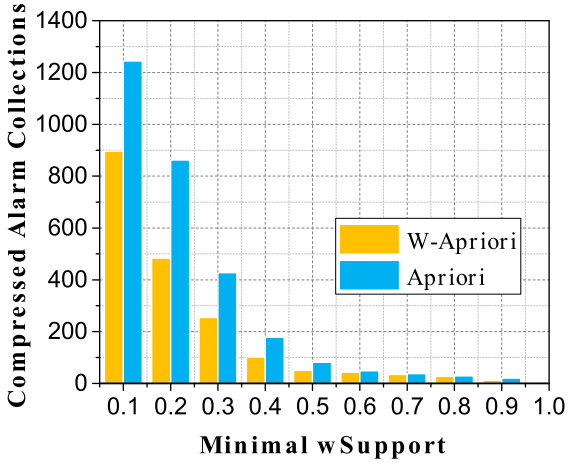

FIGURE 12. Compressed alarm collections by both algorithm with various minwSups.

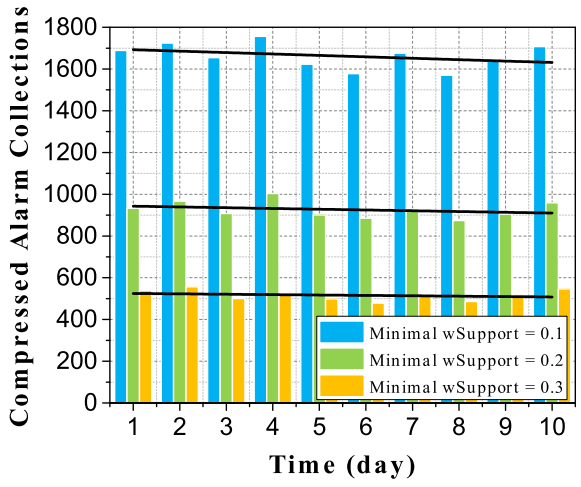

FIGURE 13. Number of compressed alarm collections v.s. minwSup.

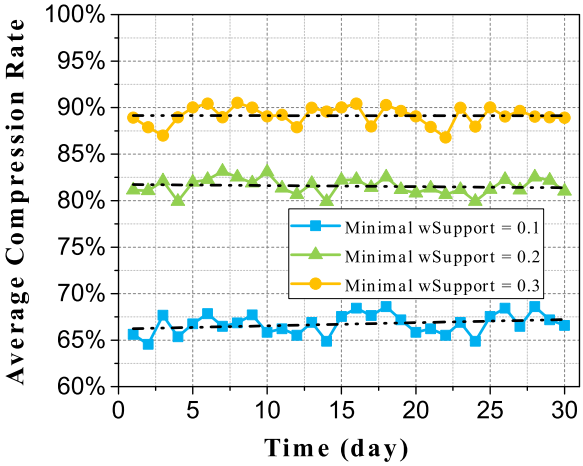

FIGURE 14. Compression rate with different minwSups.

with the same compression fidelity, W-Apriori works more effectively and leads to fewer alarm collections.

We applied the TKBW trained by the alarms in the first 20 days and then analyzed the alarms in the last 10 days in order to evaluate the alarm compression performance. Here, the average number of alarm transactions generated per day is approximately 5,000. Fig. 13 shows the variation of compressed alarm collections with various *minwSup*, in which, as *minwSup* increases from 0.1 to 0.3, the alarm collections are obviously compressed. In particular, when *minwSup* is 0.2, the number of compressed alarm collections is about only half of that when *minwSup* is 0.1. This indicates that different *minwSup* means different compression rates. Therefore, the bigger the *minwSup* is, the greater the compression rate is, and the fewer alarm collections are finally obtained.

In order to show the effects of *minwSup* on the alarm compression more directly, we plotted the alarm compression rate with different *minwSup* for total 30 days, as shown in Fig. 14. The alarm compression rate here refers to the ratio of the number of compressed alarm collections over the number of total original alarm collections. It can be seen that as *minwSup* increases from 0.1 to 0.2 and then to 0.3, the alarm compression rates rises from 67% to 82% and then to 89%, respectively. That is, larger *minwSup* yields higher compression rate. In other words, with larger *minwSup*, more alarms will be removed as false alarms.

Moreover, given a fixed *minwSup*, the compression rate remains almost stable, independent of both the training samples in the first 20 days and the application samples in the last 10 days, which indicates that the W-Apriori works stable and has little deviation between the training samples and the application samples.

The results above also indicate that, by adjusting the *minwSup*, we are able to obtain different compression rates for different purposes. Higher alarm compression rate is not necessarily desired, for that we must consider the compression fidelity, which is defined as follows.

After the alarm compression, the collections of high-frequency chain alarms were obtained. We assume that the high-frequency chain alarms are the substantial key alarms that can represent the actual faults, and the location of the

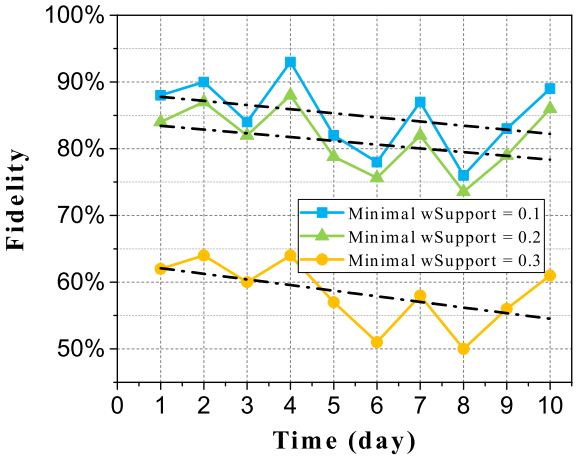

FIGURE 15. Compression fidelity with different minwSups.

fault nodes are known. Then, we define compression fidelity as the proportion of the compressed high-frequency chain alarms occurrence location containing the actual fault node location in each time period.

We measured the quality of the alarm compression in terms of compression fidelity. Fig. 15 shows the variation of compression fidelity with different *minwsups* when *minwConf* is 0.1. Here, we set *minwConf* to a smaller value, mainly in order to retain a large number of original alarm collections and facilitate the statistical experiment results. It can be seen that as the *minwSup* is 0.1 or 0.2, the fidelity keeps relatively high, namely 84% and 80% respectively, indicating that after alarm compression with such a *minwSup,* the high-frequency chain alarms can reflect the actual fault situation to a large extent. However, as the *minwSup* is 0.3, too much alarms are removed during compression and thus the fidelity remains even lower than 60%, which means in this case the alarm compression is not very believable.

Moreover, from Figs. 14 and 15, it can be seen that when *minwSup* is 0.2, the obtained alarm compression rate is 15% larger than when *minwSup* is 0.1, which indicates that when the *minwSup* is 0.2, the original alarms can be compressed to a larger extent while relatively high compression fidelity is kept. Therefore, considering both the alarm compression rate and fidelity, it may be appropriate to select the *minwSup* threshold as 0.2. However, in actual optical network, the *minwSup* threshold should be set to a particular optimal range of values according to the network scale. If the threshold is set too small, many redundant alarms (e.g., false alarms) will be retained, which is not conducive to the subsequent alarm correlation analysis and alarm compression; if the threshold is set too large, some important alarms (e.g., substantial alarms) will be removed. Therefore, in practical operation, it is necessary to set a threshold according to actual OAM requirements and operational effects.

According to the experimental results, the TKBW method can effectively perform alarm compression, alarm correlation analysis and chain alarm mining to implement the compression of alarms and obtain high-frequency chain alarms. This means that we can use high frequency chain alarms to find the root fault nodes location. Moreover, the method is not complicated in implementation. Therefore, the proposed method is easy to set up in the controller.

However, since the experimental data have come from the actual alarm monitoring record in optical layer of a provincial backbone of China Telecom, the number of alarm attributes of the fiber layer equipment is small and the attributes are relatively thick. Our method cannot fully guarantee that all OTNs are applicable. If there are more data from different OTNs, a more complete analysis method can be proposed and the universality of the method can be verified. In addition, we extracted six important and valid attributes in the experiment. If we change the number of alarm attributes (e.g., reduce the number of alarm attributes) and re-experiment, weather we get the same results will require further testing and analysis. Whatever the outcome, the proposed scheme offers a good promising model, which is interesting and will needs further future research.

IV. CONCLUSION

To deal with the problem of coping with the huge number of Alarms in OTNs, we have proposed the TKBW method, which offers alarm compression and high-frequency chain alarms mining. By taking actual data of network logs from China Telecom, we have conducted experiments, the results of which show that the TKBW method is able to give a score to indicate the importance of each alarm and the score is independent of the initial quantization values. In addition, the TKBW method is able to mine the chain alarms and compress the alarms to a rate on demand by adjusting parameters, such as *minwSup*. The results are promising and helpful for false alarm identifying and root failure locating.

APPENDIX

See Algorithms I and II.

Algorithm 1 Time Division Method Based on Similarity of Time Series

Input: k: the number of time segments n: database containing n alarm events

Output: k: k time intervals with minimum sum of squared errors

1: Select initial cluster center points for k time segments;
2: **for** (j = 1; j <= n; j++){
3: Assign eacht_j to the time interval which has the closest mean; }
4: **for** (i = 1; i <= k; i++){
5: find the cluster center point c_i for each time segment; }
6: compute $SSE = \sum_{i=1}^{k} \sum_{t \in c_i} |t, c_i|^2$;
7: **Repeat** until *SSE* conversed.

Algorithm 2 W-Apriori Algorithm

C_k: Candidate itemsets of size k
L_k: frequent itemsets of size k
Input: D: transaction database
minsup: minimum support threshold
Output: L: frequent itemsets of D
1: $L_1 = \text{FindFrequent\_1\_itemsets}(D)$;
2: //generate the frequent 1-itemsets
3: **for** $(k = 2; L_{k-1}\mathrm{L}_{k-1} \neq \emptyset \neq \emptyset; k++)$
4: $C_k = \text{Genetate Candidates}\ (L_{k-1}, minsup)$;
5: //generate the new candidate itemsets
6: **for** each transaction $t \in D$
7: $C_t = \text{subset}(C_k, t)$;
8: //find out all candidate k-itemsets contained in transaction t
9: **end for**
10: $L_k = \{c \in C_k \mid \text{support} \geq minwsup\}$
11:**end for**
12: **return** $L = \cup L_k$;

ACKNOWLEDGMENT

We would like to acknowledge the China Telecom Corporation for providing the network data

REFERENCES

[1] N. Amani, M. Fathi, and M. Dehghan, "A case-based reasoning method for alarm filtering and correlation in telecommunication networks," in *Proc. Can. Conf. Elect. Comput. Eng.*, May 2005, pp. 2182–2186.
[2] M. Klemettinen, H. Mannila, and H. Toivonen, "Rule discovery in telecommunication alarm data," *Netw. Syst. Manage. J.*, vol. 7, no. 4, pp. 395–423, 1999.
[3] W. Jian and L. X. Ming, "An effective mining algorithm for weighted association rules in communication networks," *J. Comput.*, vol. 3, no. 10, pp. 20–27, 2008.
[4] L. Yan and C. Li, "Incorporating pageview weight into an association-rule-based Web recommendation system," in *Proc. Austral. Conf. Artif. Intell.*, 2006, pp. 577–586.
[5] R. Agrawal and R. Srikant, "Fast algorithm for mining association rules," in *Proc. 20th VLDB Conf.*, 1994, pp. 487–499.
[6] G. Grahne and J. Zhu, "High performance mining of maximal frequent itemsets," in *Proc. 6th Int. Workshop High Perform. Data Mining*, 2003, pp. 135–143.
[7] T. Wang and P.-L. He, "Database encoding and an anti-apriori algorithm for association rules mining," in *Proc. Int. Conf. Mach. Learn. Cybern.*, Aug. 2006, pp. 1195–1198.
[8] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," *Data Mining Knowl. Discovery*, vol. 8, no. 1, pp. 53–87, 2004.
[9] J. Han and Y. Fu, "Discovery of multiple-level association rules from large databases," in *Proc. 21st Int. Conf. Very Large Data Bases*, 2002, pp. 420–431.
[10] C. H. Cai, A. W. C. Fu, C. H. Cheng, and W. W. Kwong, "Mining association rules with weighted items," in *Proc. Int. Database Eng. Appl. Symp.*, Jul. 1998, pp. 68–77.
[11] Z. Wang, M. Zhang, D. Wang, C. Song, M. Liu, J. Li, L. Lou, and Z. Liu, "Failure prediction using machine learning and time series in optical network," *Opt. Express*, vol. 25, no. 16, pp. 18553–18565, 2017.
[12] D. Wang, M. Zhang, Z. Li, Y. Cui, J. Liu, Y. Yang, and H. Wang, "Nonlinear decision boundary created by a machine learning-based classifier to mitigate nonlinear phase noise," in *Proc. Eur. Conf. Opt. Commun. (ECOC)*, Sep./Oct. 2015, pp. 1–3.
[13] D. Wang, M. Zhang, Z. Li, C. Song, M. Fu, J. Li, and X. Chen, "System impairment compensation in coherent optical communications by using a bio-inspired detector based on artificial neural network and genetic algorithm," *Opt. Commun.*, vol. 399, pp. 1–12, Sep. 2017.
[14] D. Wang, M. Zhang, M. Fu, Z. Cai, Z. Li, H. Han, Y. Cui, and B. Luo, "Nonlinearity mitigation using a machine learning detector based on k-nearest neighbors," *IEEE Photon. Technol. Lett.*, vol. 28, no. 19, pp. 2102–2105, Oct. 1, 2016.
[15] D. Wang, M. Zhang, J. Li, Y. Xin, J. Li, M. Wang, and X. Chen, "Intelligent optical spectrum analyzer using support vector machine," in *Proc. IEEE Photon. Soc. Summer Top. Meeting (SUM)*, Jul. 2018, pp. 239–240.
[16] D. Wang, M. Zhang, J. Li, Z. Li, J. Li, C. Song, and X. Chen, "Intelligent constellation diagram analyzer using convolutional neural network-based deep learning," *Opt. Express*, vol. 25, no. 15, pp. 17150–17166, 2017.
[17] Y. Yuan, M. Zhang, P. Luo, Z. Ghassemlooy, D. Wang, X. Tang, and D. Han, "SVM detection for superposed pulse amplitude modulation in visible light communications," in *Proc. 10th Int. Symp. Commun. Syst., Netw. Digit. Signal Process. (CSNDSP)*, Jul. 2016, pp. 1–5.
[18] J. Li, M. Zhang, and D. Wang, "Adaptive demodulator using machine learning for orbital angular momentum shift keying," *IEEE Photon. Technol. Lett.*, vol. 29, no. 17, pp. 1455–1458, Sep. 1, 2017.
[19] C. Song, M. Zhang, X. Huang, Y. Zhan, D. Wang, M. Liu, and Y. Rong, "Machine learning enabling traffic-aware dynamic slicing for 5G optical transport networks," in *Proc. Conf. Lasers Electro-Opt. (CLEO)*, 2018, pp. 1–2, Paper JTu2A.44.
[20] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "Segmenting time series: A survey and novel approach," in *Data Mining in Time Series Databases*, vol. 57. 2003, pp. 1–21.
[21] M. Hauptmann, J. H. Lubin, P. Rosenberg, J. Wellmann, L. Kreienbrock, "The use of sliding time windows for the exploratory analysis of temporal effects of smoking histories on lung cancer risk," *Statist. Med.*, vol. 19, no. 16, pp. 2185–2194, Aug. 2000.
[22] J. D. Olden, M. K. Joy, and R. G. Death, "An accurate comparison of methods for quantifying variable importance in artificial neural networks using simulated data," *Ecol. Model.*, vol. 178, nos. 3–4, pp. 389–397, 2004.

DANSHI WANG received the Ph.D. degree in electromagnetic field and microwave technology from the Beijing University of Posts and Telecommunications (BUPT), in 2016, where he is currently a Lecturer with the Institute of Information Photonics and Optical Communications. He has published more than 80 articles. His research interests include optical transmission and optical signal processing, artificial intelligence, machine learning, and wavelength switched optical networks.

LIQI LOU is currently pursuing the M.S. degree with the State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications. She has authored or coauthored 4 technical papers in international journals and conferences. Her current research interests include optical communication networks, alarm correlation analysis, fault prediction, machine learning, deep learning, and date mining.

MIN ZHANG received the Ph.D. degree in optical communications from the Beijing University of Posts and Telecommunications (BUPT), China, where he is currently a Professor, the Deputy Director of the Sate Key Laboratory of Information Photonics and Optical Communications, and the Deputy Dean of the School of Optoelectronic Information. He holds 45 China patents. He has authored or coauthored more than 300 technical papers in international journals and conferences, and 12 books in the areas of optical communications. His current research interests include optical communication systems and networks, optical signal processing, and optical wireless communications.

ANTHONY C. BOUCOUVALAS received the B.Sc. degree in electrical and electronic engineering from Newcastle upon Tyne University, U.K., in 1978, the M.Sc. and D.I.C. degrees in communications engineering from Imperial College, University of London, U.K., in 1979, and the Ph.D. degree in fibre optics from Imperial College, in 1982. He joined the GEC Hirst Research Centre, and became a Group Leader and a Divisional Chief Scientist working on fibre optic components, measurements and sensors, until 1987, when he joined Hewlett Packard Laboratories as a Project Manager. At HP, he worked in the areas of optical communication systems, optical networks, and instrumentation, until 1994, when he joined Bournemouth University. In 1996, he became a Professor in multimedia communications, and in 1999, became the Director of the Microelectronics and Multimedia Research Centre. In 2006, he joined the University of Peloponnese, Greece, where he served as the Head of the Telecommunication Sciences and Technology Department, for six years, where he is currently a Professor. He has been invited to give many conference keynote addresses, in fiber optics, in web applications, intelligent systems, and informatics for earthquake prediction.

His research interests include optical wireless communications, optical fibre communications, network protocol performance, and sensors, and HCI communications and interfaces. He has published more than 350 papers. He is a Fellow of the Institute of Engineering and Technology (FIET).

CHUNYU ZHANG is currently pursuing the Ph.D. degree in electronic science and technology with the Beijing University of Posts and Telecommunications. Her research interests include fault prediction and fault management in optical networks, machine learning, deep learning, and artificial intelligence.

XUETIAN HUANG received the Ph.D. degree in optical communications from the Beijing University of Posts and Telecommunications (BUPT), China. He is currently an Engineer of China Telecom Corporation. His current research interests include optical communication networks and optical signal processing.