Automation in Construction 125 (2021) 103624

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Generic visual data mining-based framework for revealing abnormal operation patterns in building energy systems

Chaobo Zhang, Yang Zhao *, Tingting Li, Xuejun Zhang, Meriem Adnouni

Institute of Refrigeration and Cryogenics, Zhejiang University, Hangzhou, China

ARTICLE INFO

Keywords:
Building energy systems
Pattern identification
Building energy conservation
Visual data mining
Data visualization
Maximal frequent subgraph mining

ABSTRACT

The abnormal operation patterns in building energy systems can be revealed by analyzing their historical operational data. In practice, the amount of data is so tremendous that manual data analysis is challenging. Visual data mining is a promising solution to this problem. This study proposes a generic visual data mining-based framework for extracting abnormal operation patterns in building energy systems from their historical operational data. The framework consists of three steps. First, a kernel density estimation-based approach is utilized to preprocess the raw data. Then, a decision tree-based approach is adopted to identify the system operation conditions. Finally, a maximal frequent subgraph mining-based approach is developed to reveal the system operation patterns. The framework is applied to analyze the one-year operational data of a chiller plant. This study proves that the framework can appropriately interpret the data mining results, and can make the analysis of the results more convenient.

1. Introduction

The building sector is responsible for approximately one-third of the total global energy consumption [1]. Accordingly, it is crucial to improve the energy efficiency of building energy systems [2]. For instance, Katipamula and Brambley estimated that approximately 15%–30% of the energy used in commercial buildings was wasted owing to improperly controlled, poorly maintained, and degraded equipment in building energy systems [3]. As building automation systems have gained popularity, massive amounts of historical operational data have been stored from building energy systems [4]. Thus, using these data to reveal patterns related to energy waste for building energy conservation is an attractive option. However, it is challenging and time-consuming to manually analyze such massive amounts of data.

Data mining technologies are very powerful approaches to extracting hidden knowledge from data [5]. In the past two decades, two common types of data mining-based methods have been widely applied in the building field: clustering-based methods, and association rule mining-based methods [6]. Clustering-based methods divide observations into different categories according to the geometrical distances between them [7]. Observations in the same category are similar, and observations in different categories are dissimilar. Clustering-based methods are capable of identifying building energy consumption patterns [8], occupant behavior patterns [9], sensor faults [10], etc. McLoughlin et al. developed a clustering-based method for identifying residential buildings' daily electricity use patterns [11]. Rhodes et al. introduced clustering algorithms to reveal the daily electricity consumption patterns in residential buildings [12]. Pan et al. [13] and Li et al. [14] extracted daily building electricity consumption patterns from hourly building operational data using clustering algorithms. Yu et al. used clustering algorithms to identify the monthly energy consumption patterns of building energy systems in residential buildings [15]. D'Oca et al. utilized clustering algorithms to discover occupant behavior patterns regarding opening/closing windows in office buildings [16]. Subsequently, they further proposed a clustering-based method for identifying the occupancy schedules of office buildings [17]. Association rule mining-based methods can extract the quantitative relationships between variables in a large dataset [18]. The relationships are represented in a text form of association rules, i.e., $A \rightarrow B$. In the building field, association rule mining algorithms have shown powerful abilities to identify control strategies, sensor faults, device faults, etc. Li et al. developed an association rule mining-based method for detecting the control strategies of heating, ventilation, and air conditioning (HVAC) systems [19]. Xue et al. introduced association rule mining algorithms for extracting the operation patterns of a district heating system from its historical operational data [20]. Control strategies, sensor faults, and

* Corresponding author.
E-mail address: youngzhao@zju.edu.cn (Y. Zhao).

https://doi.org/10.1016/j.autcon.2021.103624
Received 16 March 2020; Received in revised form 4 February 2021; Accepted 6 February 2021
Available online 24 February 2021

improper operation patterns were identified. Similarly, Zhang et al. presented an association rule mining-based method for analyzing the historical operational data of HVAC systems [21,22]. This method revealed the sensor faults and improper operation patterns in an existing HVAC system. Li et al. also discovered the device faults and improper operation patterns in a variable refrigerant flow air conditioning system using association rule mining algorithms [23]. Fan et al. proposed an association rule mining-based data mining framework for identifying sensor faults, improper operation patterns, and control strategies from the historical operational data of HVAC systems [24,25]. With the aim of discovering the dynamic anomalies in HVAC systems, they further presented a temporal association rule mining-based method [26] and gradual association rule mining-based method [27].

However, existing data mining-based methods still cannot process data automatically. It is usually necessary to adjust the parameters of the data mining algorithms until acceptable results are obtained. Furthermore, the results need to be analyzed manually to obtain valuable knowledge. There is a saying that goes, "a picture is worth a thousand words." Visualization technologies can display data mining results in a way that is easy for humans to understand [28]. As such, they can significantly improve the efficiency of data analysis.

Some visual analysis methods have been proposed in the building field for energy efficiency analysis, customer classification, energy management, etc. [29]. Yarbrough et al. presented a heat map-based analysis tool for visualizing building energy use patterns [30]. A similar visual analysis method was developed by Janetzko et al. for detecting power consumption anomalies [31]. The method was mainly based on three visualization technologies: recursive pattern visualization, spiral visualization, and line charts. Liu et al. utilized histograms, scatter plots, and line charts to visualize smart meter data [32]. However, most of the existing visual analysis methods do not combine visualization technologies with data mining technologies. In general, they aim to visualize raw data, rather than providing data mining results. Furthermore, they do not consider prior knowledge in the visualization process. As combinations of data mining and visualization technologies, visual data mining technologies have shown a powerful capacity to mine large databases visually in various fields such as medicine [33], image processing [34], and geography [35]. They are very useful for enhancing the interactions between domain experts and data mining processes, and can significantly improve the quality and efficiency of data mining [36]. In the building field, the results from existing data mining-based methods are usually displayed in text forms (such as association rules [37,38]), tree charts (such as decision trees [39]), and two-dimensional/three-dimensional diagrams (such as daily energy consumption curves [40]). However, three main challenges remain for these visualization techniques, as follows.

- First, it is challenging for these visualization techniques to represent prior knowledge, such as the topological structures of building energy systems. Different building energy systems have different designs and control strategies. Experts must know them in advance, so that they can understand the knowledge extracted from the various building energy systems.
- Second, it is very difficult to represent the relationships among multiple variables using these visualization techniques. The number of association rules will grow exponentially with an increase in the number of variables included in the association rules [22]. It takes a long time to extract valuable knowledge from many association rules. Therefore, association rules should not be utilized to represent the relationships among multiple variables. Decision trees are designed to discover the relationships between a target variable and correlated variables, and cannot discover the interrelationships among multiple variables. As for two-dimensional/three-dimensional diagrams, they cannot represent relationships among more than three variables.
- Finally, data mining is a complex process that includes a series of steps. Appropriate visualization technologies should be integrated into the steps requiring human intervention. However, previous studies in the building field have always focused on the visualization of the final results. They have generally ignored the visualization of results in the intermediate steps, such as in data preprocessing. It is of great value to propose an effective solution for integrating visualization technologies into each step of data mining; such a solution remains lacking in the building field.

To overcome the above challenges, a generic visual data mining-based framework is proposed in this study. It has three main contributions. The first contribution is that two types of prior knowledge-based graphs are constructed to visualize the system-level and device-level relationships among multiple variables. They can represent the topological structures of building energy systems and the quantitative relationships among multiple variables. The second contribution is that a novel maximal frequent subgraph mining-based approach is developed to reveal the operation patterns of building energy systems from the datasets of graphs. It has high computational efficiency, and can remove a vast majority of the redundant frequent subgraphs. These contributions overcome the first two challenges, and contribute to making the data mining process more efficient. The third contribution is that the visual data mining-based framework integrates visualization technologies into each step of data mining. It provides a generic solution for visualizing the results of each step of data mining, thereby overcoming the third challenge. This makes the interactions between domain experts and the data mining process more convenient.

Nomenclature

$f(x)$	a probability density function
$\max(f(x))$	the maximum of the probability density function
$x_1, x_2, \ldots, x_n$	observations of a variable
$K(.)$	a kernel function
h	a bandwidth
δ_1	the threshold of the probability density of outliers
α	scale factor of outlier threshold
δ_2	the threshold of the number of categories
C_{left}	a left consistency index
C_{right}	a right consistency index
v_1	a label of the first node of an edge
v_2	a label of the second node of an edge
l	a label of an edge
G	a graph
LR	load rate
S	an on-off state
N	the number of running devices
P	power
T	temperature
ΔT	temperature difference
CHW	chilled water
SCHW	supply chilled water
TRCHW	total return chilled water
CT	a cooling tower
CH	a chiller
COWP	a cooling water pump
PCHWP	a primary chilled water pump
SCHWP	a secondary chilled water pump
CHWV	a chilled water valve of a chiller
COWV	a cooling water valve of a chiller
CTV	an inlet valve of a cooling tower
WDH	a water distribution header
WCH	a water collection header

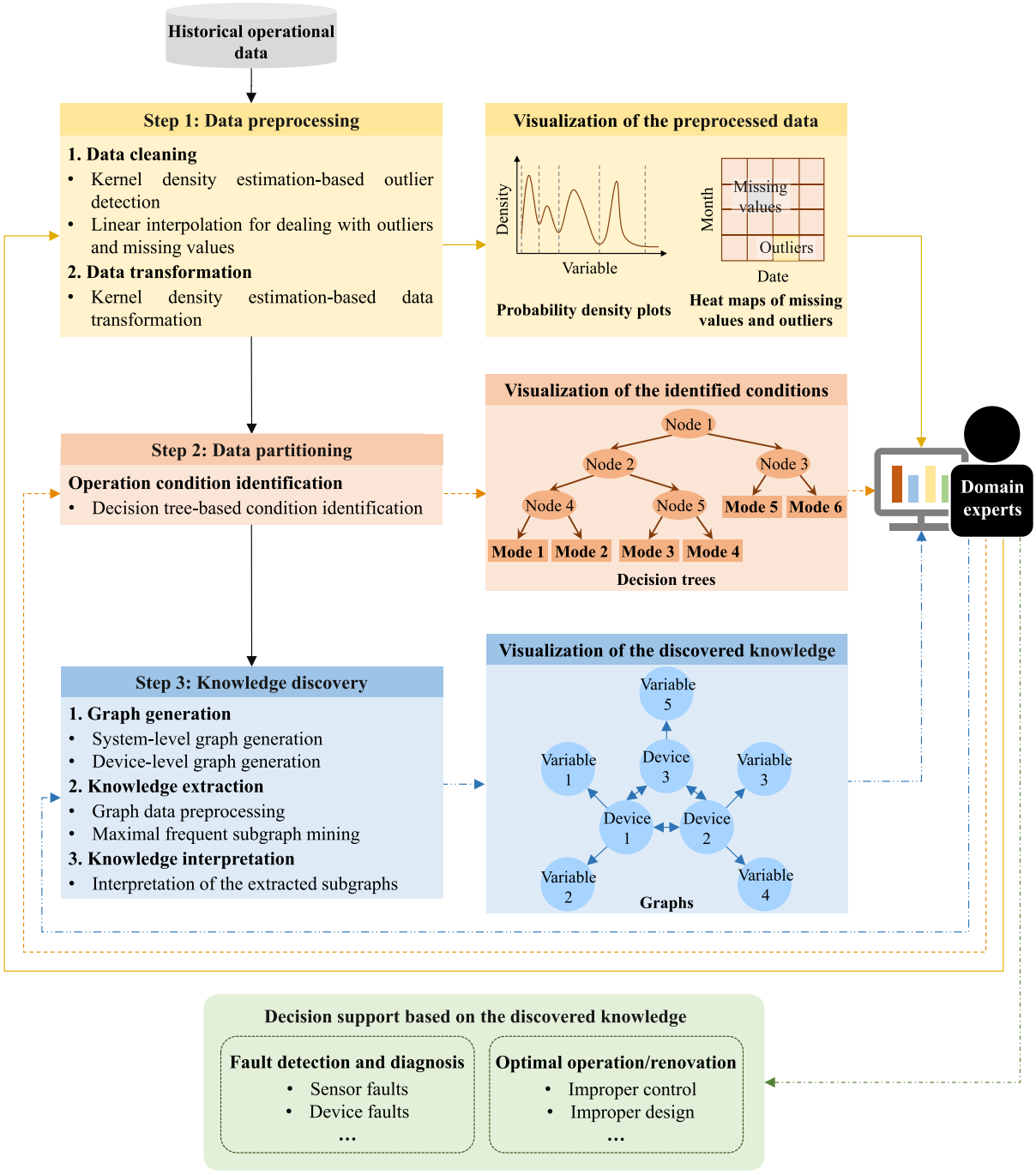

Fig. 1. Schematic of the visual data mining-based framework.

2. Methodology

A schematic of the visual data mining-based framework is shown in Fig. 1. It consists of three steps: data preprocessing, data partitioning, and knowledge discovery. In the first step, the historical operational data collected from building energy systems are preprocessed to improve the quality. In the second step, the preprocessed data are divided into different categories, with the aim of identifying the operation conditions. In the last step, the data in each category are transformed into graph data. The graph data are then processed by a top-down maximal frequent subgraph mining algorithm to discover non-redundant frequent subgraphs. Finally, the non-redundant frequent subgraphs are interpreted by domain experts, with the aim of discovering valuable operation patterns for decision support in building energy management. Visualization technologies are utilized to visualize the results of each step, so as to improve the interpretability of the results.

2.1. Kernel density estimation-based data preprocessing

2.1.1. Data preprocessing

2.1.1.1. Data cleaning. Missing values and outliers are very common in the operational data of building energy systems. The task of data cleaning is to fill in missing values and remove outliers. Outliers can be detected using unsupervised approaches, supervised approaches, and

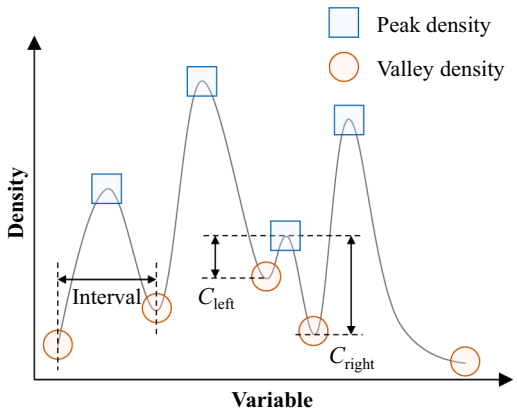

Fig. 2. Illustration of a probability density distribution of a numerical variable.

statistical approaches [24]. This study applied a kernel density estimation-based approach to detect outliers [7]. Kernel density estimation is a common approach for estimating the probability density function of the observations of a variable [41]. It is defined by Eq. (1). A probability density represents the probability of an observation occurring. According to the research of Zhang et al. [22], observations with very low densities should be regarded as outliers, as they deviate from most observations. The threshold of the probability density of the outliers is defined by Eq. (2). Observations are regarded as outliers if their probability densities are lower than the threshold.

$$f(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \quad (1)$$

$$\delta_1 = \frac{max(f(x))}{\alpha} \quad (2)$$

In the above, $f(x)$ is the probability density function; $x_1, x_2, \ldots, x_n$ are observations of a variable; $K(.)$ is a kernel function; h is a smoothing parameter denoted as the "bandwidth"; δ_1 is the threshold of the probability density of the outliers; max(f(x)) is the maximum of the probability density function; and α is a scale factor for the outlier threshold.

There are several common kernel functions such as the Gaussian kernel, Epanechnikov kernel, biweight kernel, and triweight kernel. In this study, a standard Gaussian kernel function $K(u) = \frac{1}{\sqrt{2\pi}}e^{-0.5u^2}$ is adopted, as it is widely utilized in other similar tasks and shows good performance [7,22].

The missing values and outliers can be replaced by values estimated using moving averages, imputation, etc. [24]. In this study, linear interpolation was applied to estimate the possible values of the missing values and outliers. It is difficult to process missing values and outliers that last for a long time. Therefore, missing values and outliers were addressed by linear interpolation if they lasted for a short time; and were deleted if they lasted for a long time.

2.1.1.2. Data transformation. Data transformation aims to transform numerical data into categorical data. Equal-frequency binning, equal-interval binning, and kernel density estimation-based are three common data transformation approaches [22]. A kernel density estimation-based data transformation approach was adopted in this study [22]. As shown in Fig. 2, there are usually several peak densities and several valley densities in a probability density distribution. In general, peak densities are generated owing to control strategies or unknown operation patterns in building energy systems [22]. Therefore, the numerical data can be divided into different categories according to the locations of the peak densities. Based on this concept, three steps were included in the kernel density estimation-based data transformation approach. First, observations between two adjacent valleys were classified into a category. Then, if the number of categories was greater than a given threshold of the number of categories (δ_2), the category with the minimum interval was merged with adjacent categories. The aim was to eliminate the impact of noisy data, which could potentially generate categories with very small intervals. A consistency index C, that is, the difference between adjacent valley and peak densities, was adopted to quantify the similarity between two adjacent categories. As shown in Fig. 2, there were two consistency indexes for a category: a left consistency index C_{left} and right consistency index C_{right}. The category with the minimum interval was merged with the adjacent category with a smaller consistency index. The category merging process was iterated until the number of categories was equal to the threshold of the number of

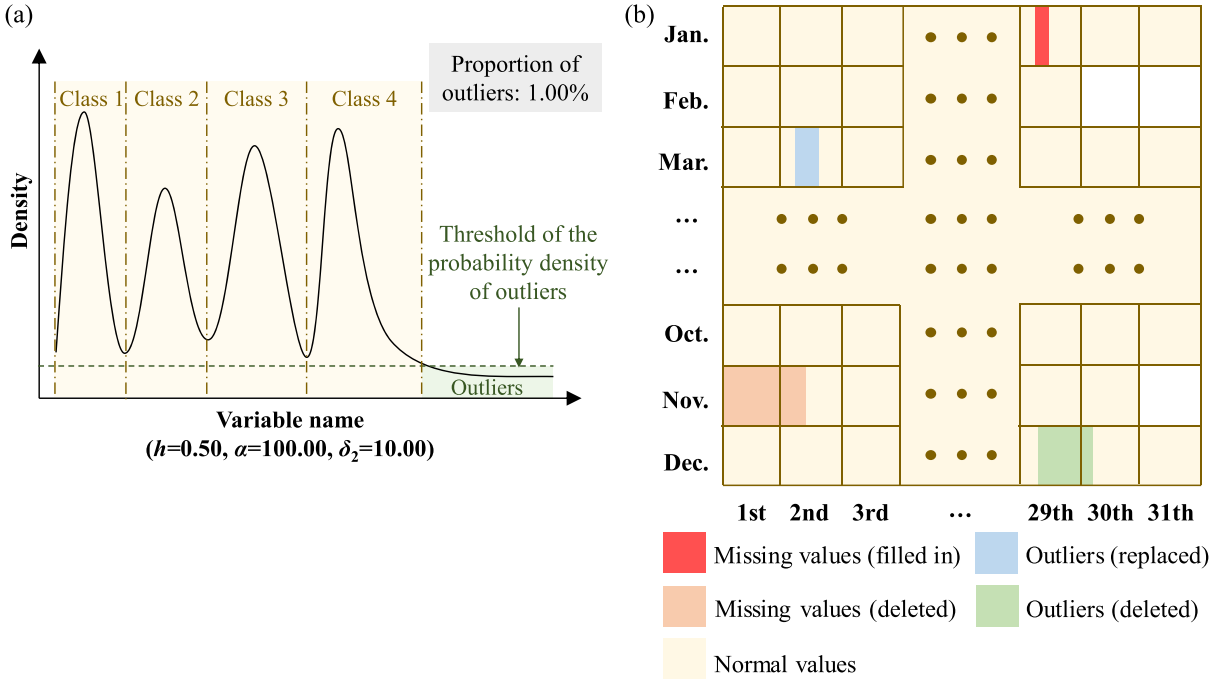

Fig. 3. Examples of (a) a probability density plot and (b) a temporal heat map.

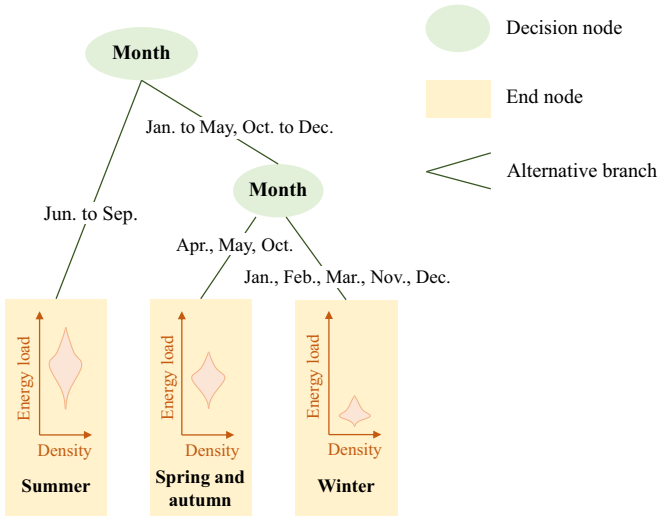

Fig. 4. Example of a decision tree of operation conditions of a building energy system.

categories. Lastly, the observations in a category were replaced by the interval of the category.

2.1.2. Visualization of the preprocessed data

Two visualization approaches were utilized to visualize the preprocessed data. The first one was a probability density plot, as shown in Fig. 3(a). It visualized the intervals of the classified data, intervals of the outliers, proportions of the outliers, and parameter values of the data preprocessing algorithms. With the help of a probability density plot, domain experts could visually understand the intervals of both classified data and outliers. Furthermore, they could adjust the bandwidth (h), scale factor of the outlier threshold (α), and threshold of the number of categories (δ_2) until the quality of the preprocessed data is sufficiently high. The second approach was a temporal heat map, as shown in Fig. 3 (b). It was a graphical representation of the temporal distribution of normal values, missing values, and outliers. The horizontal and vertical coordinates of the temporal heat map were the date and month, respectively. Thus, domain experts could easily understand the quality of the preprocessed data through the temporal heat map.

2.2. Decision tree-based data partitioning

2.2.1. Data partitioning

Building energy systems usually work under a wide range of operational conditions, owing to dynamic changes in the outdoor and indoor environment parameters. For instance, the operating conditions of an HVAC system in summer should be significantly different from those in winter. Data partitioning aims to divide a dataset into several subsets. Each subset belongs to a specific operation condition, and is mined independently.

Clustering and classification are two common data partitioning approaches [24]. Decision tree-based classification was utilized in this study. Several decision tree algorithms have been proposed, such as ID3, C4.5, classification and regression trees, and conditional inference trees [7,42]. In this study, one of the most popular decision tree algorithms, i.e., the conditional inference trees algorithm, was utilized. Compared with other decision tree algorithms, it can effectively avoid a variable selection bias during the model development process, thereby improving the interpretability of the decision trees [42]. The energy load of a building energy system is the most crucial feature for characterizing the system operation conditions. Therefore, in this study, it was selected as the output of the decision tree model.

2.2.2. Visualization of the identified conditions

The operation conditions identified by the conditional inference trees algorithm were visualized using decision tree diagrams. There are three main components in a decision tree diagram: decision nodes, end nodes, and alternative branches. A decision node indicates a decision to be made. An end node shows a possible outcome. Alternative branches represent the paths for the different choices of a decision. Fig. 4 illustrates an example of a decision tree diagram for the operating conditions of a building energy system. The probability density distributions of the energy loads for each condition are visualized using a violin plot [43]. Domain experts can easily determine whether the identified operation conditions are reliable according to the decision tree diagrams.

2.3. Maximal frequent subgraph mining-based knowledge discovery

2.3.1. Knowledge discovery

2.3.1.1. Graph generation. Graphs have a powerful ability to visualize the complex relationships among the multiple measured variables in

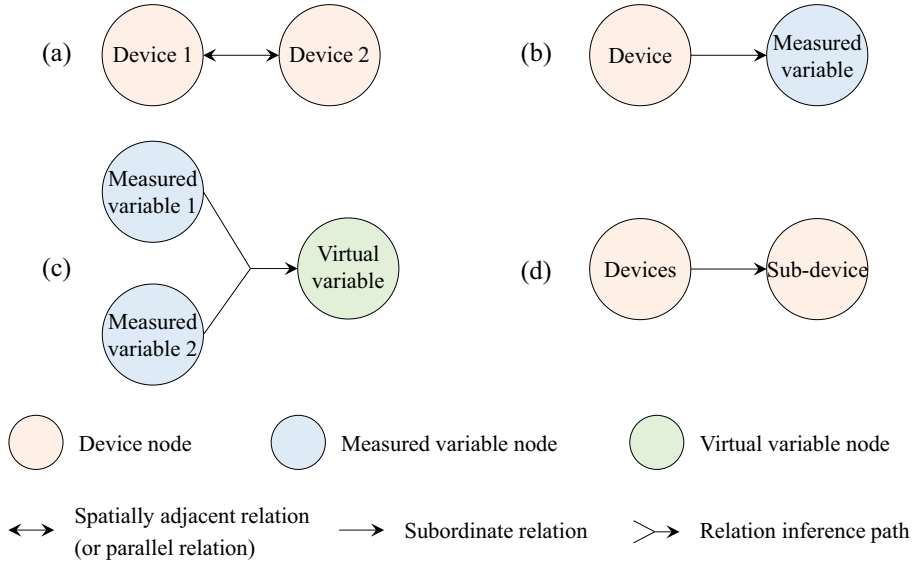

Fig. 5. Relations between nodes in a graph.

building energy systems [44]. In practice, building energy systems' operational data are always stored in a two-dimensional table where one dimension is time, and the other dimension is the measured variable. In the sub-step of graph generation, the operational data stored in a two-dimensional table are transformed into graphs (stored in a dataset of graphs).

In general, the knowledge hidden in the operational data of building energy systems can be divided into two levels: system-level knowledge, and device-level knowledge. System-level knowledge is related to system operation patterns. Device-level knowledge is related to device operation patterns. Therefore, in this study, two types of prior knowledge-based graphs were constructed to map the two-dimensional operational data into graphs, as follows.

- The first type of graph was a system-level graph representing the system-level prior knowledge. It considered all of the devices. Three types of nodes were included in the system-level graph: device nodes, measured variable nodes, and virtual variable nodes. The nodes were connected by edges, which represented the relations between nodes. Three types of relations were considered. The first type was a spatially adjacent relation between different types of devices, as shown in Fig. 5(a). For instance, chillers and chilled water pumps are always connected by pipes. They are spatially adjacent. The second one was a subordinate relation between the devices and measured variables, as shown in Fig. 5(b). For instance, the total power of all chillers is subordinate to the chillers. The third type of relation comprised the relations between the measured variables and virtual variables, as shown in Fig. 5(c). For instance, the chilled water temperature difference is a useful variable that is usually not monitored in practice. Nevertheless, it can be calculated as a virtual variable, based on the supply and return chilled water temperatures.
- The second type of graph was a device-level graph for representing device-level prior knowledge. It considered the same types of devices. Two types of nodes were considered in the device-level graph: device nodes, and measured variable nodes. Three types of relations were considered. The first type was a parallel relation between similar devices, as shown in Fig. 5(a). For instance, centrifugal chillers and screw chillers are usually connected in parallel if the two types of chillers both exist in an HVAC system. The second one was a subordinate relation between the devices and measured variables, as shown in Fig. 5(b). For instance, the supply chilled water (SCHW) temperature of a chiller is subordinate to the chiller. The third type of relation was a subordinate relation between devices and sub-devices, as shown in Fig. 5(d). For instance, each chiller is a sub-device if an HVAC system includes more than one chiller.

2.3.1.2. Knowledge extraction. The sub-step of knowledge extraction includes two parts: graph preprocessing, and maximal frequent subgraph mining. Graph preprocessing aims at removing the nodes of useless variables. Maximal frequent subgraph mining aims at extracting non-redundant operation patterns from a dataset of graphs.

In general, devices are always redundant in building energy systems. For instance, there are usually several chillers in an HVAC system, and only some of them are in operation most of the time. When devices are shut down, their operation patterns are useless for energy efficiency analyses. The power or on-off state of a device can be employed to judge whether the device is working. Moreover, some nodes, such as device nodes, are constant in the graphs. Such nodes are useful for knowledge visualization. However, they increase the computational load of graph mining algorithms, as the computational load significantly depends on the number of nodes in the graphs. Therefore, in this study, a graph preprocessing approach was proposed. It merged both the measured variable nodes of devices that never worked and nodes with constant values into several fusion nodes. It reduced the number of nodes in the graphs, without the loss of useful information.

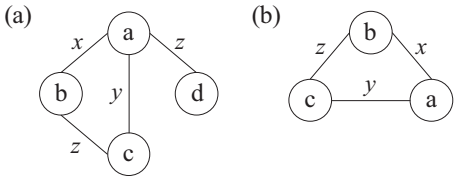

Fig. 6. Graph (a) and its subgraph (b).

As a popular graph mining technology, frequent subgraph mining can identify common structures from a dataset of graphs, and has been employed in various fields such as social networks, transportation networks, and biological networks [45]. A subgraph is regarded as frequent if its *support*, i.e., the frequency of the subgraph occurring, exceeds a prespecified threshold. In general, frequent subgraph mining algorithms can be divided into two categories according to the subgraph search strategy [46]. The first category includes breadth-first strategy-based algorithms such as the Apriori-based graph mining [47] and frequent subgraph [48]. The second category includes depth-first strategy-based algorithms such as the graph-based substructure pattern mining (gSpan) [49], MoFa [50], fast frequent subgraph mining [51], and "Graph, Sequences and Tree Extraction" [52]. However, most of the mined frequent subgraphs are redundant. They usually have supergraphs that include the same graph structures as them, and cannot provide new information relative to their supergraphs.

Mining the maximal frequent subgraphs is an effective solution for identifying non-redundant frequent subgraphs. A frequent subgraph is regarded as a maximal frequent subgraph if none of its supergraphs are frequent [53]. The knowledge hidden in maximal frequent subgraphs is the same as that hidden in frequent subgraphs, as a frequent subgraph must be a subgraph of the maximal frequent subgraph. Bottom-up mining algorithms (such as spanning tree-based maximal graph mining [53] and "MARGIN" [54]) and top-down mining algorithms (such as "Top-Down" [55]) are two common types of maximal frequent subgraph mining algorithms. According to the research results of Guo et al. [55], top-down mining algorithms have a lower computational load than other algorithms.

Therefore, in this study, a top-down maximal frequent subgraph mining algorithm was developed to extract the maximal frequent subgraphs from the dataset of preprocessed graphs. Unlike conventional top-down algorithms, a simplified judgment approach to subgraph isomorphism was presented, based on the assumption that a node in a graph had a unique label. Both the system-level graphs and device-level graphs satisfied this assumption, as the nodes of a graph corresponded to the different sensors and devices. Under this assumption, each edge of a graph could be represented by a unique triple (v_1, l, v_2), where v_1, l, and v_2 were the labels of the first node, edge, and second node, respectively. Then, a graph could be represented by a set of triples of the edges. For instance, graphs (a) and (b) shown in Fig. 6 can be represented by {(a, x, b), (a, z, d), (a, y, c), (b, z, c)}, and {(b, z, c), (b, x, a), (c, y, a)}, respectively. It can be seen that (a, x, b) and (b, x, a) represent the same structure of an edge, but they have different representations. A node ranking strategy was adopted to avoid the ambiguity in representations of the same edges in different graphs. It included two steps. First, a node label was randomly assigned a unique identification number in {1, 2, …, n}, where n was the number of all of the non-redundant labels of the nodes in all graphs. Then, an ordered triple (v_1, l, v_2) of an edge was created to replace the original triple, following the criterion that the identification number of v_1 was less than the identification number of v_2. After node ranking, the same edges in different graphs had the same representation. Moreover, a graph could be represented by a unique set of triples of edges. Graph G_1 was considered as a subgraph of another graph G_2 when the set of edges of G_1 was a proper subset of the set of edges of G_2. The two graphs G_1 and G_2 were isomorphic when the sets of edges of the two graphs were equal.

Table 1
Pseudocode of the top-down maximal frequent subgraph mining algorithm.

Algorithm Top-Down Maximal Frequent Subgraph Mining (*D*, *min_sup*)

Inputs: A dataset of preprocessed graphs $D = \{G_1, G_2, \ldots, G_m\}$ and a threshold of *support min_sup*.

Output: A set of maximal frequent subgraphs *S*.

begin

1: find all edges of preprocessed graphs;
2: sort nodes in triples of every edge;
3: replace original edges of graphs in *D* with the sorted edges;
4: count *support* of each edge;
5: find infrequent edges whose *supports* are less than *min_sup*;
6: remove the infrequent edges in each graph in *D*;
7: set *D*' to ∅;
8: **for** each graph without infrequent edges **do**
9: **if** the graph doesn't have an isomorphic graph in *D*' **then**
10: insert the graph to *D*';
11: the *support* of the graph is equal to 1;
12: **else**
13: the *support* of its isomorphic graph in *D*' plus 1;
14: set *num* to the maximal number of edges of a graph in *D*';
15: set *S* to ∅;
16: set *d* to ∅;
17: **while** *num* is not less than 1 **do**
18: **if** there exist graphs in *D*' whose number of edges is equal to *num* **then**
19: **for** each graph in *D*' whose number of edges is equal to *num* **do**
20: **if** the graph doesn't have an isomorphic graph in *d* **then**
21: insert the graph to *d*;

```
    for each graph in d do
        calculate the sum of supports of its supergraphs in D' and its isomorphic graphs in D';
        if the sum of supports is greater than min_sup then
            remove the graph from d;
            if the graph doesn't have supergraphs in S then
                insert the graph to S;
    for each remaining graph in d do
        obtain all subgraphs which have num-1 edges of the remaining graph;
        for each subgraph do
            if the subgraph doesn't have an isomorphic graph in d then
                insert the subgraph to d;
        remove the remaining graph from d;
    num minus 1
return
end
```

The pseudocode for the top-down maximal frequent subgraph mining algorithm is listed in Table 1. It has four steps. First, the nodes of an edge in each graph are sorted using a node ranking strategy. Next, the infrequent edges in each graph are removed, as graphs are infrequent if they include infrequent edges. Then, the isomorphic graphs without infrequent edges are merged. The *support* of a graph with isomorphic graphs is equal to the number of its isomorphic graphs plus one. Finally, the graphs are mined, following a top-down search strategy. The search starts from graphs with the maximal number of edges. Their *supports* are counted. If a graph is frequent, it is identified as a maximal frequent subgraph. If a graph is not frequent, its subgraphs with one edge less than it must be further mined. In the next search, graphs with one edge less than the graphs in the previous search are mined, together with the subgraphs from the previous search. The search is ended when the number of edges of graphs to be mined is less than one.

2.3.1.3. Knowledge interpretation. The maximal frequent subgraphs extracted by the top-down maximal frequent subgraph mining algorithm must be interpreted by domain experts. Abnormal operation patterns, such as device faults, sensor faults, and improper control strategies, can be detected after knowledge interpretation. They are useful for improving building energy efficiency. The visualization approach described in Section 2.3.2 was selected to visualize the extracted maximal frequent subgraphs.

2.3.2. Visualization of the discovered knowledge

A visualization approach was adopted to visualize the maximal frequent subgraphs. As shown in Fig. 7, it comprised three parts. The first part showed the maximal frequent subgraph, which was displayed in the left pane. Its fusion nodes were restored before knowledge visualization. The second part showed the temporal distribution of the maximal frequent subgraph. It was displayed in the upper-right pane, and indicated the time at which the maximal frequent subgraph occurred. The third part showed several typical days of the maximal frequent subgraph, and was displayed in the bottom-right pane. Considering that a maximal frequent subgraph might occur on many days, only the three days with the top three frequencies of the maximal frequent subgraph occurring were displayed.

3. Evaluation

3.1. Description of the data source

HVAC systems are the most typical building energy systems. The one-year historical operational data of a chiller plant of an HVAC system of a public building located in Shenzhen, China, are employed to evaluate the visual data mining-based framework. The data were collected in 2017, with a sampling interval of 10 min. In this chiller plant, there are eight chillers (CH1 to CH8), 10 primary chilled water pumps (PCHWP1 to PCHWP10), 14 secondary chilled water pumps (SCHWP1 to

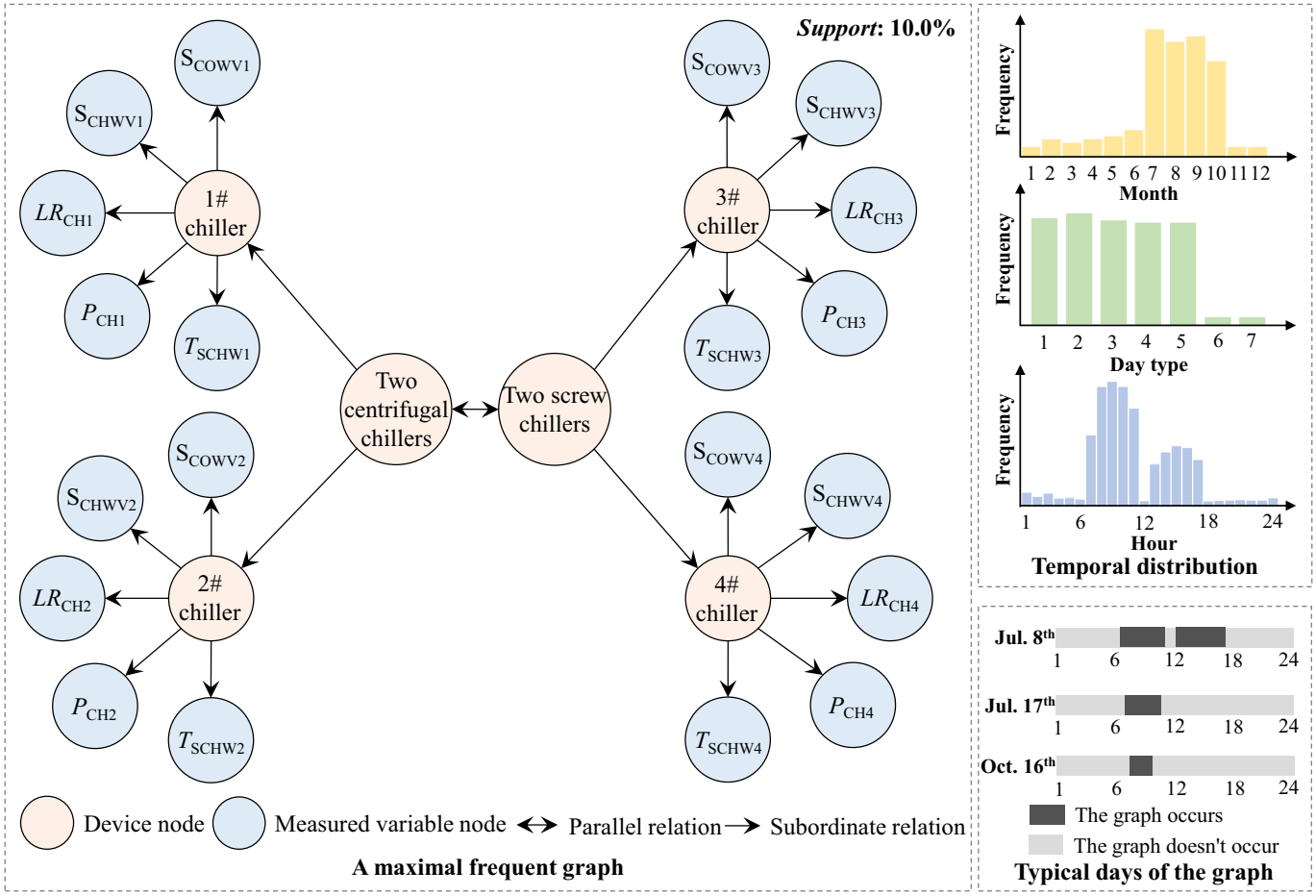

Fig. 7. Illustration of the maximal frequent subgraph visualization approach.

SCHWP14), 10 cooling water pumps (COWP1 to COWP10), and 20 cooling towers (CT1 to CT20). A total of seven types of variables were monitored, including the power (P) of the devices, the number (N) of running devices, the frequency of the pumps, the flow rate of the chilled water, the temperatures (T) of the SCHW and return chilled water, the temperatures of the supply cooling water and return cooling water, and the on-off states (S) of the devices/valves. Two virtual variables were calculated in this study, that is, the temperature difference (ΔT) of the chilled water, and the temperature difference of the cooling water.

The computations are executed in a desktop computer with a 3.4 GHz Intel Core i5 processor and 24 GB of storage memory. Two types of open-source programming languages, Python and R, are adopted to implement the visual data mining-based framework. A Python package named "statsmodels" is utilized to implement the kernel density estimation algorithm. A package of R named "party" is chosen to implement the conditional inference trees algorithm. A Python package named "gspan-mining" is selected to implement gSpan. Three Python packages (Seaborn [56], Matplotlib [57], and NetworkX [58]) are utilized for the visualizations.

3.2. Preprocessing and visualizing the raw data

A kernel density estimation-based outlier detection approach is adopted to identify the outliers. A kernel density estimation-based data transformation approach is applied to transform the numerical data into categorical data. The standard Gaussian kernel function is selected as the kernel function for the kernel density estimation. Three crucial parameters are initialized based on engineering experience: the bandwidth (h), scale factor of the outlier threshold (α), and threshold of the number of categories (δ_2). The initial bandwidth (h) is 0.50. The initial scale factor of the outlier threshold (α) is 100.00. The initial threshold of the number of categories (δ_2) is 10.00. The results from the outlier detection and data transformation are visualized using probability density plots. Based on the probability density plots, the three parameters are adjusted if the results are not as good as expected.

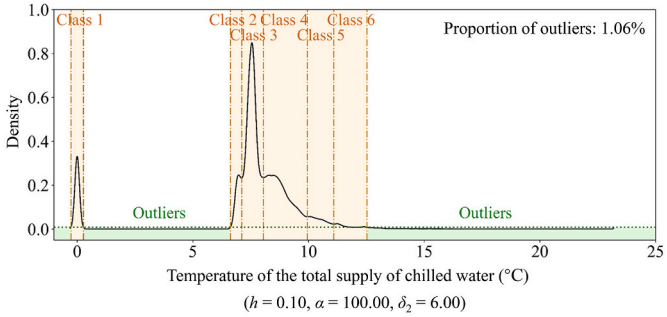

Fig. 8. Visual preprocessing results of the temperature of the total supply of chilled water.

The visual preprocessing results for the temperature of the total supply of chilled water are shown in Fig. 8. The values of h, α, and δ_2 are 0.10, 100.00, and 6.00, respectively. A total of six classes are obtained, and the interval of each class is suitable. The first class should be regarded as sensor faults, as the temperature of the total supply of chilled water cannot be 0.0 °C. The intervals of the second and third classes are [6.6 °C, 7.1 °C] and [7.1 °C, 8.1 °C], respectively. This is

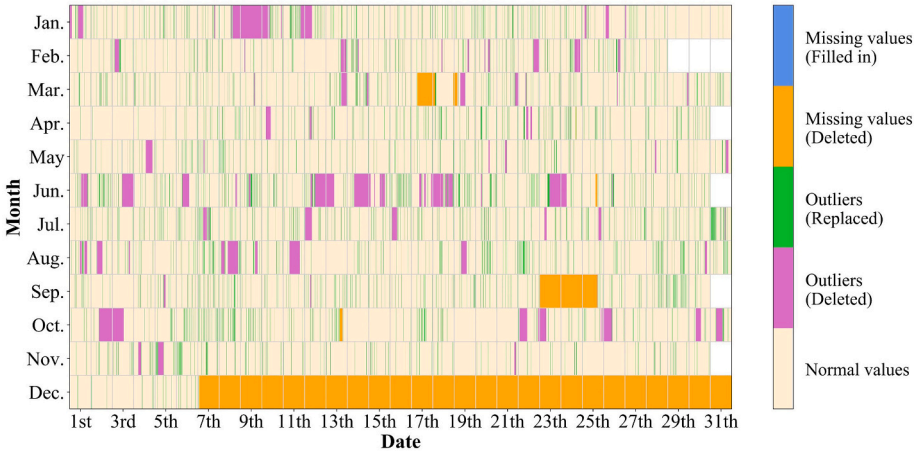

Fig. 9. Temporal heat map of the missing values and outliers.

reasonable, as the temperature of the total supply of chilled water is usually controlled to be approximately 7.0 °C in practice. The fourth, fifth, and sixth classes show that the temperature of the total supply of chilled water is controlled to be high. This might result from an energy conservation strategy, as raising the SCHW temperature of a chiller can improve the coefficient of performance of the chiller. Observations are identified as outliers if they are between 0.3 °C and 6.6 °C, or are higher than 12.5 °C. This is reasonable, as the temperature of the total supply of chilled water cannot be too low or too high in practice. Ultimately, a total of 1.06% of the observations of this variable are regarded as outliers.

Both outliers and missing values are managed by the linear interpolation algorithm if they last for less than 1 h. Otherwise, they are discarded. A temporal heat map of the missing values and outliers is shown in Fig. 9. A total of 7.91% of the raw data are missing for more than 1 h. Moreover, a total of 4.98% of the raw data are identified as outliers lasting for more than 1 h. Therefore, 12.89% of the raw data are deleted. According to Fig. 9, it can be seen that the quality of the data is very poor in December. The data are missing from December 7th, 2017 forward, accounting for 6.83% of the raw data. However, the quality of the data in the other months is acceptable.

3.3. Identifying and visualizing operation conditions of the chiller plant

The conditional inference trees algorithm is utilized to identify the operation conditions of the chiller plant. The hourly cooling load of the chiller plant is selected as the output of this algorithm, and is calculated based on the chilled water flow rate and chilled water temperature difference. Three time variables (*Month*, *Day Type*, and *Hour*) are selected as the inputs of this algorithm for identifying the seasonal conditions, weekly conditions, and daily conditions, respectively. Three decision trees are obtained, as shown in Figs. 10(a), (b), and (c). As shown in Fig. 10(a), four operation conditions are identified based on *Month*. The cooling loads of the first and second conditions are significantly higher than those of the third and fourth conditions. They are typical operation conditions in summer. The cooling loads of the fourth condition are the lowest. This is a typical operation condition in winter. The third condition is a typical operation condition for spring and autumn. As shown in Fig. 10(b), two operation conditions are identified based on *Day Type*. They represent operation conditions on weekdays and weekends, respectively. As shown in Fig. 10(c), two conditions are identified based on *Hour*. They represent the operation conditions in the daytime and nighttime, respectively. The results show that the cooling loads in the daytime, i.e., from 8:00 to 19:00, are usually significantly higher than those in the nighttime, i.e., from 20:00 on a day to 7:00 on the next day. This is reasonable, as the work time for occupants in this building is approximately from 8:00 to 19:00. Finally, the entire dataset is divided into 12 datasets, corresponding to the 12 operation conditions listed in Table 2.

3.4. Discovering and visualizing abnormal operation patterns of the chiller plant

A system-level graph is applied to transform the preprocessed data related to the system-level knowledge in the 12 operation conditions into 12 datasets of graphs. Four device-level graphs are utilized to transform the preprocessed data related to the device-level knowledge in the 12 operation conditions into 48 datasets of graphs. The four device-level graphs are related to chillers, cooling towers, cooling water pumps, and chilled water pumps, respectively. The minimum threshold of *support* is 20.00% multiplied by the size of the dataset.

3.4.1. Performance comparisons between the top-down maximal frequent subgraph mining algorithm and gSpan

Previous studies discovered that gSpan has better performance in terms of memory usage and computational load than other common frequent subgraph mining algorithms [59]. Therefore, this study compares the performance of the top-down maximal frequent subgraph mining algorithm with that of gSpan. Two indexes, i.e., the computational load and number of discovered subgraphs, are considered. The performances of the two algorithms on the 60 datasets of graphs are listed in Tables 3, 4, and 5. The total number of maximal frequent subgraphs accounts for only 0.03% of that of the frequent subgraphs mined by gSpan. Moreover, the total computational load of the top-down maximal frequent subgraph mining algorithm accounts for only 3.73% of that of gSpan. This indicates that, overall, the top-down maximal frequent subgraph mining algorithm significantly outperforms gSpan.

In some cases, the computational load of the top-down maximal frequent subgraph mining algorithm is higher than that of gSpan. This is mainly because the subgraph search strategies are different. The top-down maximal frequent subgraph mining algorithm is based on a top-down search strategy. It starts the search from subgraphs with the maximum number of edges, and proceeds to subgraphs with the minimum number of edges. The computational load significantly depends on the number of edges of the maximal frequent subgraphs. The greater the number of edges of the maximal frequent subgraphs, the lower the computational load. Unlike the top-down maximal frequent subgraph mining algorithm, gSpan is based on a bottom-up search strategy. It starts the search from frequent edges, and proceeds to the frequent subgraphs with the maximum number of edges. The smaller the number of edges in the maximal frequent subgraphs, the lower the

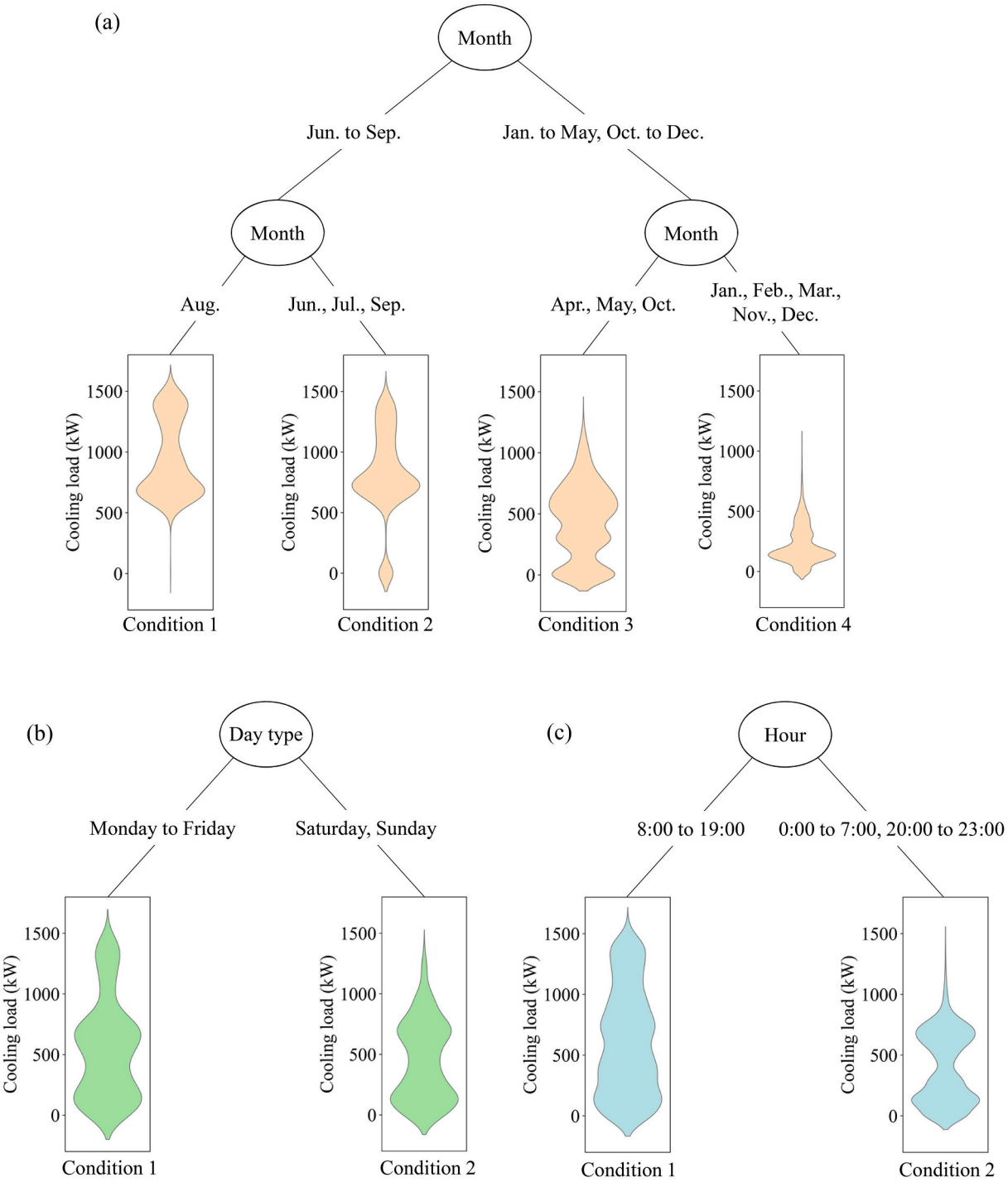

Fig. 10. Decision trees for the operation conditions of the chiller plant.

computational load. Therefore, the computational load of the top-down maximal frequent subgraph mining algorithm might be higher than that of gSpan if the number of edges in the maximal frequent subgraphs is relatively small.

3.4.2. Abnormal operation patterns revealed from the extracted maximal frequent subgraphs

According to Table 3, a total of 362 maximal frequent subgraphs are discovered from the datasets of the graphs in summer conditions. According to Table 4, a total of 620 maximal frequent subgraphs are discovered from the datasets of the graphs in winter conditions. According to Table 5, a total of 471 maximal frequent subgraphs are discovered from the datasets of the graphs in spring and autumn conditions. All of the maximal frequent subgraphs are visualized using the approach described in Section 2.3.2. Then, the domain experts check them manually to detect abnormal operation patterns. Two cases are taken as examples to illustrate two typical abnormal operation patterns, as described in Sections 3.4.2.1 and 3.4.2.2.

3.4.2.1. Case 1: Abnormal operation patterns of chillers. Two system-level maximal frequent subgraphs of abnormal operation patterns are discovered under conditions 5 and 6, as shown in Figs. 11 and 12, respectively. According to Fig. 11, the number of running chillers (N_{CH})

Table 2
Operation conditions of the chiller plant.

No.	Month	Day type	Hour
1	Summer	Weekday	Daytime
2			Nighttime
3		Weekend	Daytime
4			Nighttime
5	Winter	Weekday	Daytime
6			Nighttime
7		Weekend	Daytime
8			Nighttime
9	Spring and autumn	Weekday	Daytime
10			Nighttime
11		Weekend	Daytime
12			Nighttime

is usually only one in the daytime on weekdays from January to March. According to Fig. 12, the number of running chillers (N_{CH}) is usually two during the nighttime on weekdays from January to March. It is abnormal that the number of running chillers in the nighttime is greater than that during the daytime, as the building cooling loads in the nighttime should be lower than those in the daytime. It is also discovered that the number of running cooling water pumps (N_{COWP}) increases with the number of running chillers (N_{CH}). This results in additional cooling water pumps working in the nighttime.

To further explain this abnormal operation pattern, the chiller-level maximal frequent subgraphs discovered under condition 6 are further investigated. An abnormal operation pattern of the #7 chiller is discovered. As shown in Fig. 13, the #7 chiller (CH7) and #8 chiller (CH8) usually work simultaneously on weekdays from January to March. Furthermore, the chilled water valve of the #7 chiller (CHWV7) is closed when the #7 chiller works. However, the valve cannot be closed tightly, so a certain amount of chilled water flows through the #7 chiller. This results in a very low temperature of the SCHW of the #7 chiller (T_{SCHW7}), i.e., 3.1–5.0 °C. After consulting with the building operation staff, it was found that the control strategies of the #7 chiller were wrong during the nighttime. The chilled water valve was not opened when the #7 chiller worked. According to Fig. 11, a chiller and cooling water pump are sufficient during the daytime. Therefore, it is suggested to turn on a chiller and cooling water pump in the nighttime, so as to improve the energy efficiency of this chiller plant.

3.4.2.2. Case 2: Abnormal operation patterns of cooling towers. Abnormal patterns are discovered in the input valves of the #5 and #6 cooling towers under condition 1 according to two maximal frequent subgraphs, as shown in Figs. 14 and 15. According to Figs. 14 and 15, the input valve of the #6 cooling tower (CTV6) is usually closed during the daytime on weekdays from June to September when the #6 cooling tower (CT6) works. According to Fig. 15, the input valve of the #5 cooling tower (CTV5) is usually not closed during the daytime on weekdays from July to September when the #5 cooling tower (CT5) does not work. Both operation patterns are abnormal, as the on-off state of the cooling tower and that of its input valve should be consistent. After consulting with the building operation staff, it was determined that the input valves of the #5 and #6 cooling towers might be stuck. Such abnormal operation patterns resulted in energy waste, although they did not significantly affect the temperature of the total supply of cooling water. For instance, the #6 cooling tower worked invalidly when its input valve was closed, resulting in energy waste. However, other cooling towers still worked to adjust the temperature of the total supply of cooling water.

4. Discussions

4.1. Comparisons between the visual data mining-based framework and conventional methods

The visual data mining-based framework can present the results of data preprocessing, data partitioning, and knowledge discovery in a visual way. Moreover, it can consider prior knowledge for knowledge discovery and knowledge visualization. The framework has two main advantages over conventional data mining methods.

In particular, domain experts can easily understand the results from data mining using this framework. The prior knowledge-based graphs can indicate the quantitative relationships among multiple variables and system topological structures, whereas conventional data mining methods cannot. For instance, the results from clustering and association rule mining cannot consider prior knowledge. Their results will be difficult to understand if there are many variables considered.

In addition, the amount of worthless knowledge mined by the maximal frequent subgraph mining-based knowledge discovery approach could be significantly smaller than that mined by conventional data mining methods. This is mainly because prior knowledge-based

Table 3
Performances of gSpan and the top-down maximal frequent subgraph mining algorithm in summer conditions.

Algorithm	Performance	Condition 1	Condition 2	Condition 3	Condition 4
Dataset of system-level graphs					
gSpan	Computational load (s)	1662.37	902.76	26.21	649.27
	Number of discovered subgraphs	30,905	15,865	1159	23,179
Top-down	Computational load (s)	149.89	200.50	15.35	139.65
	Number of maximal frequent subgraphs	21	66	48	38
Dataset of chilled water pump-level graphs					
gSpan	Computational load (s)	2.31	4.66	0.82	1.16
	Number of frequent subgraphs	35	83	33	57
Top-down	Computational load (s)	0.88	7.44	3.85	4.83
	Number of maximal frequent subgraphs	10	11	10	11
Dataset of cooling tower-level graphs					
gSpan	Computational load (s)	25,187.11	21,990.90	0.58	10,209.98
	Number of frequent subgraphs	196,607	196,607	23	196,607
Top-down	Computational load (s)	1.18	4.58	1.33	2.09
	Number of maximal frequent subgraphs	2	2	2	2
Dataset of cooling water pump-level graphs					
gSpan	Computational load (s)	81.59	83.53	97.75	101.02
	Number of frequent subgraphs	1908	1831	4407	4869
Top-down	Computational load (s)	16.61	18.74	8.78	6.22
	Number of maximal frequent subgraphs	23	31	6	4
Dataset of chiller-level graphs					
gSpan	Computational load (s)	2182.12	752.61	126.04	248.95
	Number of frequent subgraphs	36,807	13,661	5141	9403
Top-down	Computational load (s)	80.25	212.18	46.73	24.50
	Number of maximal frequent subgraphs	15	23	24	13

Table 4
Performances of gSpan and the top-down maximal frequent subgraph mining algorithm in winter conditions.

Algorithm	Performance	Condition 5	Condition 6	Condition 7	Condition 8
Dataset of system-level graphs					
gSpan	Computational load (s)	72.37	66.92	43.15	46.36
	Number of frequent subgraphs	1815	1447	1981	2393
Top-down	Computational load (s)	1691.73	1172.26	149.91	367.04
	Number of maximal frequent subgraphs	102	36	39	43
Dataset of chilled water pump-level graphs					
gSpan	Computational load (s)	7.38	11.65	3.86	5.59
	Number of frequent subgraphs	211	285	161	273
Top-down	Computational load (s)	0.79	0.73	1.41	0.68
	Number of maximal frequent subgraphs	14	4	11	9
Dataset of cooling tower-level graphs					
gSpan	Computational load (s)	184,355.71	23,768.83	12,668.04	17,166.61
	Number of frequent subgraphs	1,509,635	336,241	315,679	409,659
Top-down	Computational load (s)	3892.87	396.60	265.40	264.61
	Number of maximal frequent subgraphs	33	29	23	25
Dataset of cooling water pump-level graphs					
gSpan	Computational load (s)	77.25	110.03	88.36	91.60
	Number of frequent subgraphs	1567	2183	4071	4089
Top-down	Computational load (s)	2.78	10.10	1.02	10.28
	Number of maximal frequent subgraphs	43	32	8	13
Dataset of chiller-level graphs					
gSpan	Computational load (s)	418.94	586.20	71.84	69.85
	Number of frequent subgraphs	9344	12,339	3476	3179
Top-down	Computational load (s)	801.38	1232.53	147.34	123.10
	Number of maximal frequent subgraphs	41	39	39	37

Table 5
Performances of gSpan and the top-down maximal frequent subgraph mining algorithm in spring and autumn conditions.

Algorithm	Performance	Condition 9	Condition 10	Condition 11	Condition 12
Dataset of system-level graphs					
gSpan	Computational load (s)	144.00	28.04	12.24	10.91
	Number of frequent subgraphs	4189	852	702	657
Top-down	Computational load (s)	254.76	401.47	203.41	184.17
	Number of maximal frequent subgraphs	71	57	36	38
Dataset of chilled water pump-level graphs					
gSpan	Computational load (s)	3.19	5.01	3.88	4.18
	Number of frequent subgraphs	72	163	262	282
Top-down	Computational load (s)	0.91	1.03	0.92	0.61
	Number of maximal frequent subgraphs	10	8	5	5
Dataset of cooling tower-level graphs					
gSpan	Computational load (s)	15,759.90	17,867.23	8535.53	8215.14
	Number of frequent subgraphs	245,797	246,269	260,141	260,079
Top-down	Computational load (s)	14.78	18.27	39.60	21.72
	Number of maximal frequent subgraphs	4	3	7	5
Dataset of cooling water pump-level graphs					
gSpan	Computational load (s)	103.41	64.99	57.12	36.99
	Number of frequent subgraphs	3507	2002	3493	2200
Top-down	Computational load (s)	4.60	7.31	1.97	2.59
	Number of maximal frequent subgraphs	22	26	15	26
Dataset of chiller-level graphs					
gSpan	Computational load (s)	396.72	112.08	144.83	87.05
	Number of frequent subgraphs	11,297	2945	8490	4706
Top-down	Computational load (s)	282.17	201.49	79.42	82.68
	Number of maximal frequent subgraphs	46	38	28	21

graphs constrain the relationships among the variables to be mined. The relationships among the unrelated variables are not included in the prior knowledge-based graphs. Therefore, they will be ignored in the process of knowledge discovery. Considering that association rule mining was always utilized to perform the same task in previous studies, a common association rule mining algorithm, named frequent pattern growth [60], is applied to mine knowledge from the 12 datasets corresponding to the 12 operation conditions listed in Table 2. The threshold of *support* for the association rule mining is set to 20.00%. A total of 213,950 two-variable association rules are mined. However, there are only 1453 maximal frequent subgraphs according to Tables 3, 4, and 5. The number of association rules is significantly greater than the number of maximal frequent subgraphs. This means that the maximal frequent subgraph mining-based knowledge discovery approach can reduce the amount of time experts spend interpreting the discovered knowledge.

4.2. Application scenarios for visual data analysis in other types of building energy systems

The visual data mining-based framework was applied to analyze the operational data of a chiller plant. The abnormal operation patterns were successfully revealed, and were valuable for improving the energy efficiency of this chiller plant. Apart from the chiller plant, the framework also has great potential for visual data analysis for other types of building energy systems, such as lighting systems, and heating systems. In essence, building energy systems can be regarded as networks composed of devices, sensors, etc. Therefore, the relationships among the variables of these energy systems can also be represented using

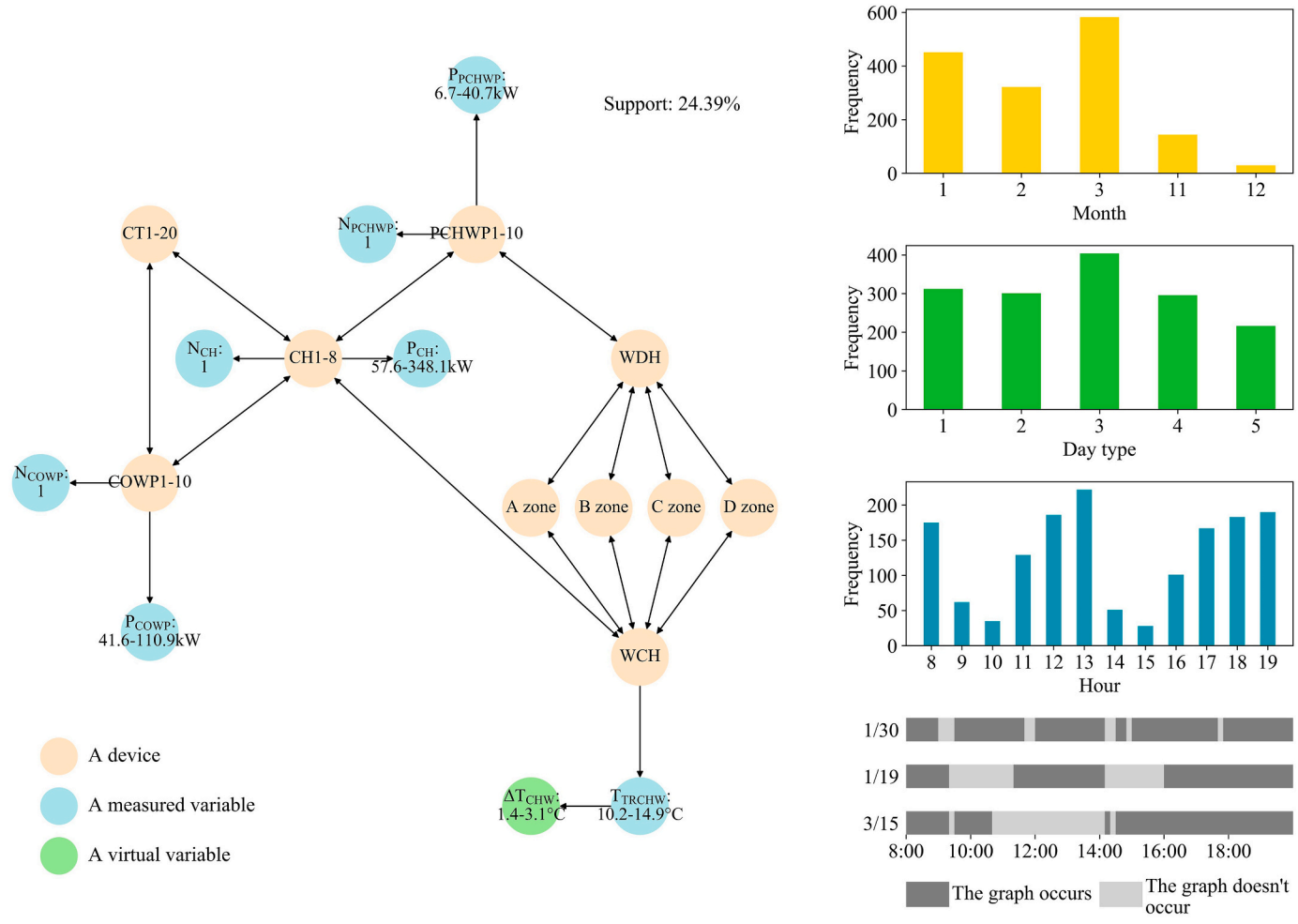

Fig. 11. Maximal frequent subgraph of system-level knowledge under condition 5.

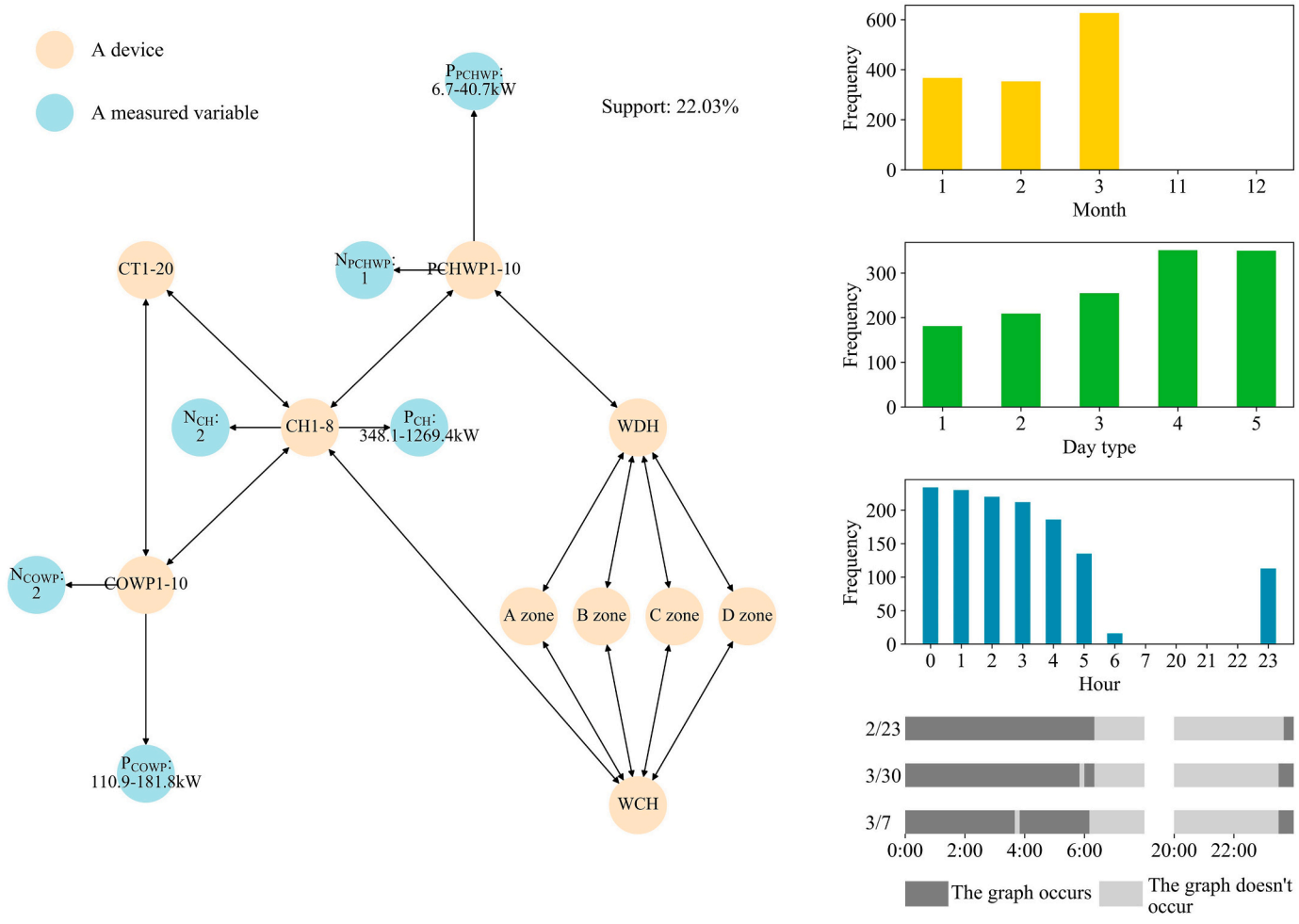

Fig. 12. Maximal frequent subgraph of system-level knowledge under condition 6.

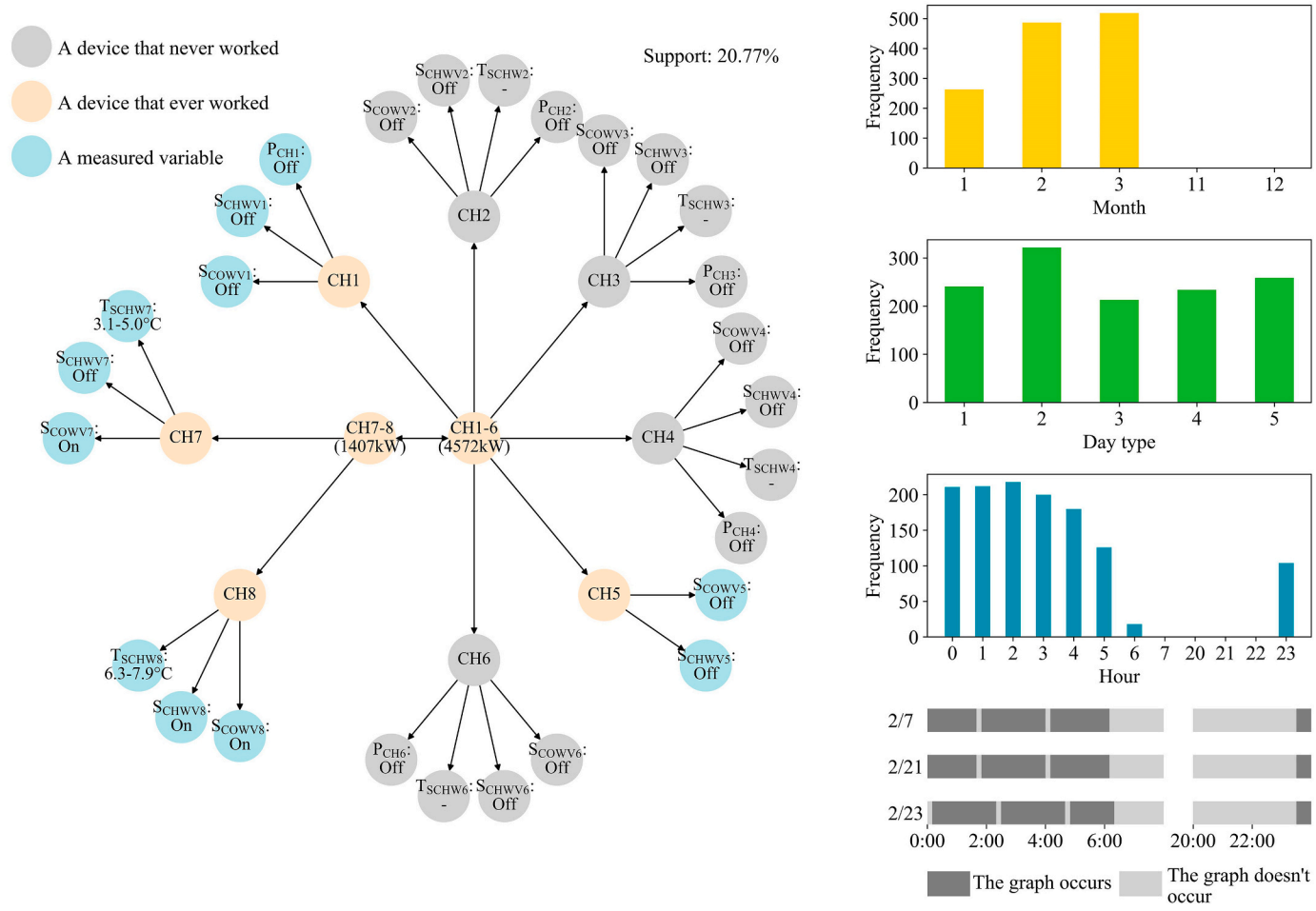

Fig. 13. Maximal frequent subgraph of an abnormal pattern of the #7 chiller under condition 6.

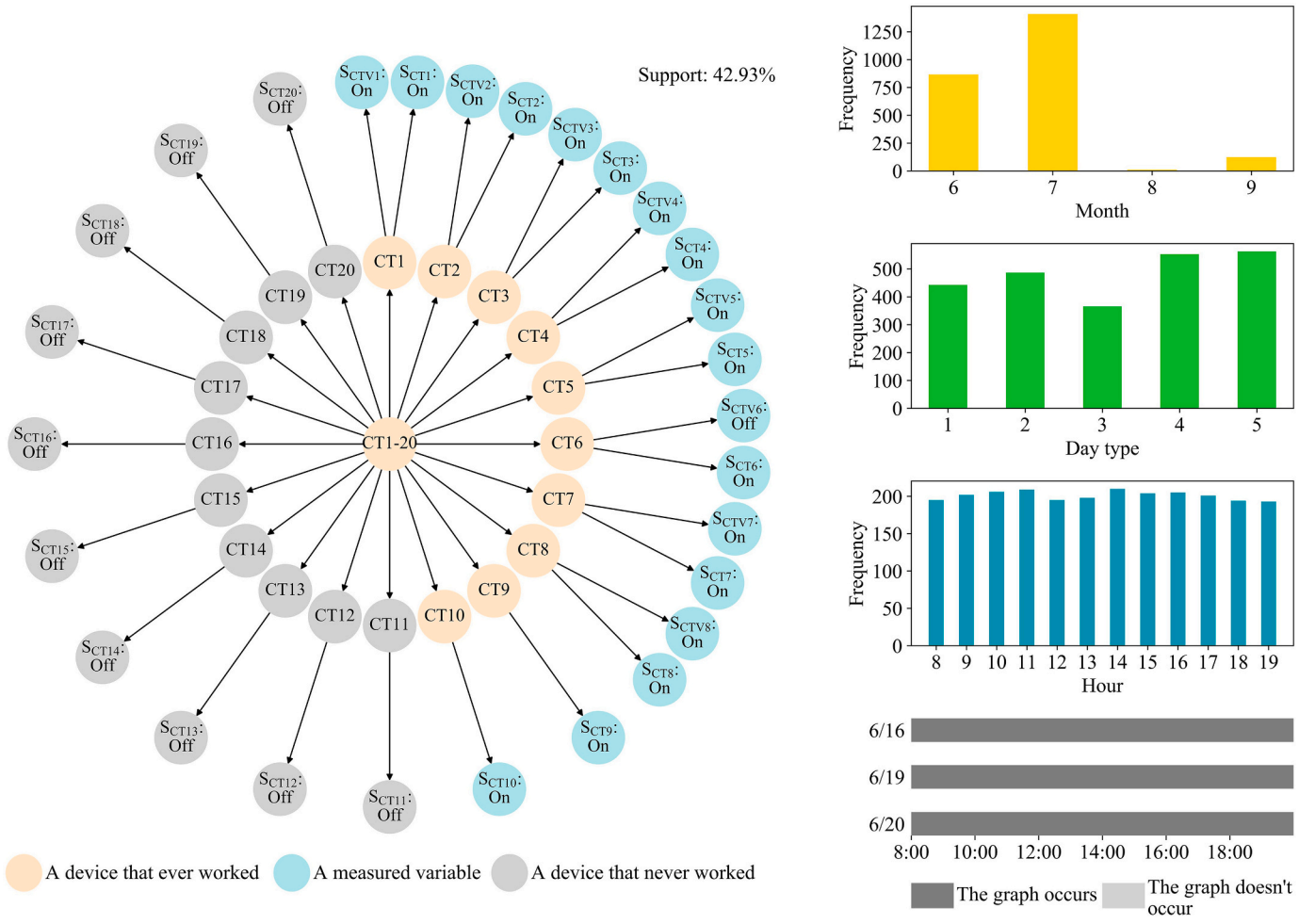

Fig. 14. Maximal frequent subgraph of an abnormal pattern of the #6 cooling tower under condition 1.

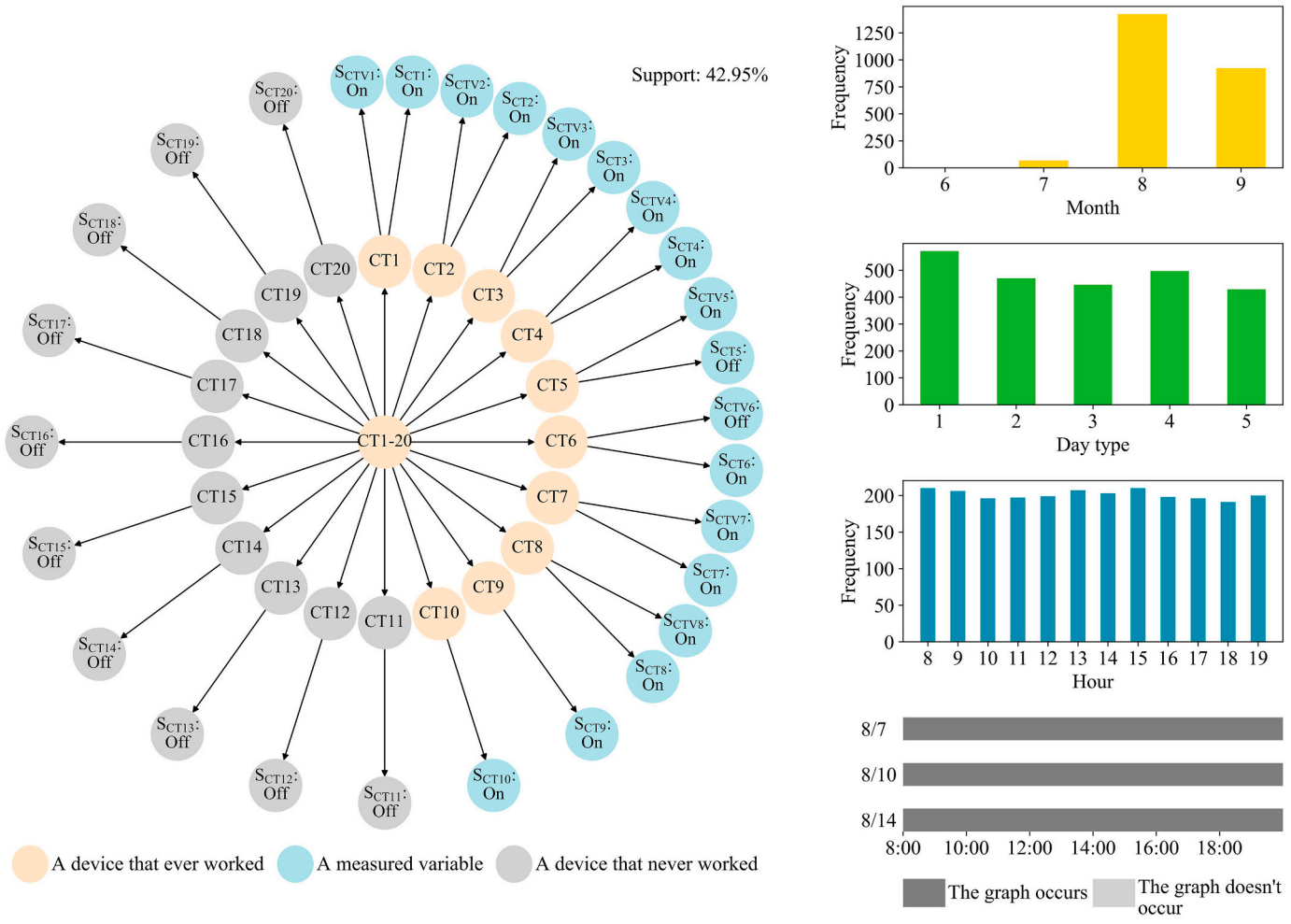

Fig. 15. Maximal frequent subgraph of abnormal patterns of the #5 and #6 cooling towers under condition 1.

graphs. The framework helps to analyze the operational data of these energy systems in a graph-based visual way. It provides a solution for extracting value from the massive amounts of operational data of building energy systems to improve their energy efficiency.

4.3. Limitations and future works

The maximal frequent subgraphs were interpreted manually in this study. In general, the maximal frequent subgraphs of abnormal and normal operation patterns should have different statistical characteristics. It might be possible to distinguish them automatically. It is of great value to study how to distinguish between the maximal frequent subgraphs of abnormal and normal operation patterns, as it could greatly improve the efficiency of knowledge interpretation.

Moreover, other graph mining algorithms, for example, link prediction, graph classification, and graph clustering, also have great potential to be utilized in the building field. Additional studies are encouraged to investigate the potential application scenarios of these algorithms in the building field.

5. Conclusions

The interpretability of visual data mining technologies is better than that of conventional data mining technologies. This study proposes a generic visual data mining-based framework for revealing abnormal operation patterns in building energy systems in a deep visual understanding way. A kernel density estimation-based approach is utilized for data preprocessing. The preprocessed data are visualized using probability density plots and temporal heat maps. A decision tree-based approach is adopted to identify the operating conditions of the building energy systems. Two types of prior knowledge-based graphs are constructed, i.e., to visualize system-level and device-level knowledge of the building energy systems, respectively. A top-down maximal frequent subgraph mining algorithm is developed to extract the non-redundant operation patterns of the building energy systems from the datasets of the graphs.

The one-year operational data of a chiller plant in a public building located in Shenzhen, China, are analyzed using the visual data mining-based framework. The results validate the great potential of the framework. With the help of the visualization technologies, it is convenient to adjust the parameters of the kernel density estimation-based data preprocessing approach, and to evaluate the quality of the preprocessed data. A total of 12 operation conditions are identified by the decision tree-based approach, and it is easy to explain them using decision tree diagrams. The results also show that the top-down maximal frequent subgraph mining algorithm has higher computational efficiency and less redundant knowledge than the most common frequent subgraph mining algorithm (gSpan). The total computational load of the top-down maximal frequent subgraph mining algorithm accounts for only 3.73% of that of gSpan. Moreover, the total number of maximal frequent subgraphs mined by the top-down maximal frequent subgraph mining algorithm accounts for only 0.03% of that of the frequent subgraphs mined by gSpan. Abnormal operation patterns are successfully discovered, such as abnormal control of chillers, and input valve faults in cooling towers.

They are valuable for improving the energy efficiency of the chiller plant.

Two future research directions are considered. The first is to study how to distinguish between the maximal frequent subgraphs of abnormal and normal operation patterns. The second is to investigate potential application scenarios for other graph mining algorithms, for example, link prediction, graph classification, and graph clustering. All of the codes are available for download within a GitHub repository (https://github.com/iEnergyX-lab/A-visual-data-mining-framework-for-building-energy-systems).

Declaration of Competing interest

None.

Acknowledgements

The authors gratefully acknowledge the support of the National Key Research and Development Program of China (No. 2018YFE0116300) and the National Natural Science Foundation of China (No. 51978601).

References

[1] T. Hong, L. Yang, D. Hill, W. Feng, Data and analytics to inform energy retrofit of high performance buildings, Appl. Energy 126 (2014) 90–106, https://doi.org/10.1016/j.apenergy.2014.03.052.
[2] Y. Zhao, T. Li, X. Zhang, C. Zhang, Artificial intelligence-based fault detection and diagnosis methods for building energy systems: advantages, challenges and the future, Renew. Sust. Energ. Rev. 109 (2019) 85–101, https://doi.org/10.1016/j.rser.2019.04.021.
[3] S. Katipamula, M.R. Brambley, Review article: methods for fault detection, diagnostics, and prognostics for building systems—a review, part I, HVAC&R Research 11 (1) (2005) 3–25, https://doi.org/10.1080/10789669.2005.10391123.
[4] Z. Yu, B.C.M. Fung, F. Haghighat, Extracting knowledge from building-related data — a data mining framework, Build. Simul. 6 (2013) 207–222, https://doi.org/10.1007/s12273-013-0117-8.
[5] C. Fan, F. Xiao, Z. Li, J. Wang, Unsupervised data analytics in mining big building operational data for energy efficiency enhancement: a review, Energy Build. 159 (2018) 296–308, https://doi.org/10.1016/j.enbuild.2017.11.008.
[6] Y. Zhao, C. Zhang, Y. Zhang, Z. Wang, J. Li, A review of data mining technologies in building energy systems: load prediction, pattern identification, fault detection and diagnosis, Energy Built Environ. 1 (2) (2019) 149–164, https://doi.org/10.1016/j.enbenv.2019.11.003.
[7] J. Han, M. Kamber, J. Pei, Data Mining: Concepts and Techniques, 3rd ed., Morgan Kaufmann, Waltham, 2012 (ISBN:978-0-12-381479-1).
[8] C. Wang, Y. Du, H. Li, F. Wallin, G. Min, New methods for clustering district heating users based on consumption patterns, Appl. Energy 251 (2019) 113373, https://doi.org/10.1016/j.apenergy.2019.113373.
[9] X. Ren, C. Zhang, Y. Zhao, G. Boxem, W. Zeiler, T. Li, A data mining-based method for revealing occupant behavior patterns in using mechanical ventilation systems of Dutch dwellings, Energy Build. 193 (2019) 99–110, https://doi.org/10.1016/j.enbuild.2019.03.047.
[10] R. Yan, Z. Ma, G. Kokogiannakis, Y. Zhao, A sensor fault detection strategy for air handling units using cluster analysis, Autom. Constr. 70 (2016) 77–88, https://doi.org/10.1016/j.autcon.2016.06.005.
[11] F. McLoughlin, A. Duffy, M. Conlon, A clustering approach to domestic electricity load profile characterisation using smart metering data, Appl. Energy 141 (2015) 190–199, https://doi.org/10.1016/j.apenergy.2014.12.039.
[12] J.D. Rhodes, W.J. Cole, C.R. Upshaw, T.F. Edgar, M.E. Webber, Clustering analysis of residential electricity demand profiles, Appl. Energy 135 (2014) 461–471, https://doi.org/10.1016/j.apenergy.2014.08.111.
[13] S. Pan, X. Wang, Y. Wei, X. Zhang, C. Gal, G. Ren, D. Yan, Y. Shi, J. Wu, L. Xia, J. Xie, J. Liu, Cluster analysis for occupant-behavior based electricity load patterns in buildings: a case study in Shanghai residences, Build. Simul. 10 (2017) 889–898, https://doi.org/10.1007/s12273-017-0377-9.
[14] K. Li, Z. Ma, D. Robinson, J. Ma, Identification of typical building daily electricity usage profiles using Gaussian mixture model-based clustering and hierarchical clustering, Appl. Energy 231 (2018) 331–342, https://doi.org/10.1016/j.apenergy.2018.09.050.
[15] Z. Yu, B.C.M. Fung, F. Haghighat, H. Yoshino, E. Morofsky, A systematic procedure to study the influence of occupant behavior on building energy consumption, Energy Build. 43 (6) (2011) 1409–1417, https://doi.org/10.1016/j.enbuild.2011.02.002.
[16] S. D'Oca, T. Hong, A data-mining approach to discover patterns of window opening and closing behavior in offices, Build. Environ. 82 (2014) 726–739, https://doi.org/10.1016/j.buildenv.2014.10.021.
[17] S. D'Oca, T. Hong, Occupancy schedules learning process through a data mining framework, Energy Build. 88 (2015) 395–408, https://doi.org/10.1016/j.enbuild.2014.11.065.
[18] Y. Cheng, W. Yu, Q. Li, GA-based multi-level association rule mining approach for defect analysis in the construction industry, Autom. Constr. 51 (2015) 78–91, https://doi.org/10.1016/j.autcon.2014.12.016.
[19] S. Qiu, F. Feng, Z. Li, G. Yang, P. Xu, Z. Li, Data mining based framework to identify rule based operation strategies for buildings with power metering system, Build. Simul. 12 (2019) 195–205, https://doi.org/10.1007/s12273-018-0472-6.
[20] P. Xue, Z. Zhou, X. Fang, X. Chen, L. Liu, Y. Liu, J. Liu, Fault detection and operation optimization in district heating substations based on data mining techniques, Appl. Energy 205 (2017) 926–940, https://doi.org/10.1016/j.apenergy.2017.08.035.
[21] C. Zhang, Y. Zhao, X.J. Zhang, An improved association rule mining-based method for discovering abnormal operation patterns of HVAC systems, Energy Procedia 158 (2019) 2701–2706, https://doi.org/10.1016/j.egypro.2019.02.025.
[22] C. Zhang, X. Xue, Y. Zhao, X. Zhang, T. Li, An improved association rule mining-based method for revealing operational problems of building heating, ventilation and air conditioning (HVAC) systems, Appl. Energy 253 (2019) 113492, https://doi.org/10.1016/j.apenergy.2019.113492.
[23] G. Li, Y. Hu, H. Chen, H. Li, M. Hu, Y. Guo, J. Liu, S. Sun, M. Sun, Data partitioning and association mining for identifying VRF energy consumption patterns under various part loads and refrigerant charge conditions, Appl. Energy 185 (2017) 846–861, https://doi.org/10.1016/j.apenergy.2016.10.091.
[24] C. Fan, F. Xiao, C. Yan, A framework for knowledge discovery in massive building automation data and its application in building diagnostics, Autom. Constr. 50 (2015) 81–90, https://doi.org/10.1016/j.autcon.2014.12.006.
[25] C. Fan, F. Xiao, Mining big building operational data for improving building energy efficiency: a case study, Build. Serv. Eng. Res. Technol. 39 (1) (2018) 117–128, https://doi.org/10.1177/0143624417704977.
[26] C. Fan, F. Xiao, H. Madsen, D. Wang, Temporal knowledge discovery in big BAS data for building energy management, Energy Build. 109 (2015) 75–89, https://doi.org/10.1016/j.enbuild.2015.09.060.
[27] C. Fan, Y. Sun, K. Shan, F. Xiao, J. Wang, Discovering gradual patterns in building operations for improving building energy efficiency, Appl. Energy 224 (2018) 116–123, https://doi.org/10.1016/j.apenergy.2018.04.118.
[28] T. Soukup, I. Davidson, Visual Data Mining: Techniques and Tools for Data Visualization and Mining, 1st ed., John Wiley & Sons, New York, 2002 (ISBN:0-471-14999-3).
[29] C. Miller, Z. Nagy, A. Schlueter, A review of unsupervised statistical learning and visual analytics techniques applied to performance analysis of non-residential buildings, Renew. Sust. Energ. Rev. 81 (2018) 1365–1377, https://doi.org/10.1016/j.rser.2017.05.124.
[30] I. Yarbrough, Q. Sun, D.C. Reeves, K. Hackman, R. Bennett, D.S. Henshel, Visualizing building energy demand for building peak energy analysis, Energy Build. 91 (2015) 10–15, https://doi.org/10.1016/j.enbuild.2014.11.052.
[31] H. Janetzko, F. Stoffel, S. Mittelstädt, D.A. Keim, Anomaly detection for visual analytics of power consumption data, Comput. Graph. 38 (2014) 27–37, https://doi.org/10.1016/j.cag.2013.10.006.
[32] X. Liu, L. Golab, I.F. Ilyas, SMAS: A smart meter data analytics system, in: Proceedings of the 2015 IEEE 31st International Conference on Data Engineering, 2015, pp. 1476–1479, https://doi.org/10.1109/ICDE.2015.7113405.
[33] A. Rosado-Muñoz, J.M. Martínez-Martínez, P. Escandell-Montero, E. Soria-Olivas, Visual data mining with self-organising maps for ventricular fibrillation analysis, Comput. Methods Prog. Biomed. 111 (2) (2013) 269–279, https://doi.org/10.1016/j.cmpb.2013.02.011.
[34] Q. Zhang, X. Song, Y. Yang, H. Ma, R. Shibasaki, Visual graph mining for graph matching, Comput. Vis. Image Underst. 178 (2019) 16–29, https://doi.org/10.1016/j.cviu.2018.11.002.
[35] U. Demšar, Investigating visual exploration of geospatial data: an exploratory usability experiment for visual data mining, Comput. Environ. Urban. Syst. 31 (5) (2007) 551–571, https://doi.org/10.1016/j.compenvurbsys.2007.08.006.
[36] S.J. Lee, K. Siau, A review of data mining techniques, Ind. Manag. Data Syst. 101 (1) (2001) 41–46, https://doi.org/10.1108/02635570110365989.
[37] F. Xiao, C. Fan, Data mining in building automation system for improving building operational performance, Energy Build. 75 (2014) 109–118, https://doi.org/10.1016/j.enbuild.2014.02.005.
[38] D.F.M. Cabrera, H. Zareipour, Data association mining for identifying lighting energy waste patterns in educational institutes, Energy Build. 62 (2013) 210–216, https://doi.org/10.1016/j.enbuild.2013.02.049.
[39] J. Liu, J. Wang, G. Li, H. Chen, L. Shen, L. Xing, Evaluation of the energy performance of variable refrigerant flow systems using dynamic energy benchmarks based on data mining techniques, Appl. Energy 208 (2017) 522–539, https://doi.org/10.1016/j.apenergy.2017.09.116.
[40] C. Miller, Z. Nagy, A. Schlueter, Automated daily pattern filtering of measured building performance data, Autom. Constr. 49 (2015) 1–17, https://doi.org/10.1016/j.autcon.2014.09.004.
[41] E. Parzen, On estimation of a probability density function and mode, Ann. Math. Stat. 33 (1962) 1065–1076, https://doi.org/10.1214/aoms/1177704472.
[42] T. Hothorn, K. Hornik, A. Zeileis, Unbiased recursive partitioning: a conditional inference framework, J. Comput. Graph. Stat. 15 (3) (2006) 651–674, https://doi.org/10.1198/106186006X133933.
[43] J.L. Hintze, R.D. Nelson, Violin plots: a box plot-density trace synergism, Am. Stat. 52 (2) (1998) 181–184, https://doi.org/10.1080/00031305.1998.10480559.
[44] C. Fan, F. Xiao, M. Song, J. Wang, A graph mining-based methodology for discovering and visualizing high-level knowledge for building energy management, Appl. Energy 251 (2019) 113395, https://doi.org/10.1016/j.apenergy.2019.113395.

[45] S.H. Farhi, D. Boughaci, Two bi-objective hybrid approaches for the frequent subgraph mining problem, Appl. Soft Comput. 72 (2018) 291–297, https://doi.org/10.1016/j.asoc.2018.07.058.
[46] B. Güvenoglu, B.E. Bostanoglu, A qualitative survey on frequent subgraph mining, Open Comput. Sci. 8 (1) (2018) 194–209, https://doi.org/10.1515/comp-2018-0018.
[47] A. Inokuchi, T. Washio, H. Motoda, An Apriori-based algorithm for mining frequent substructures from graph data, in: D.A. Zighed, J. Komorowski, J. Żytkow (Eds.), Principles of Data Mining and Knowledge Discovery, Springer, Berlin, 2000, pp. 13–23, https://doi.org/10.1007/3-540-45372-5_2.
[48] M. Kuramochi, G. Karypis, Frequent subgraph discovery, in: Proceedings of the 2001 IEEE International Conference on Data Mining, 2001, pp. 313–320, https://doi.org/10.1109/ICDM.2001.989534.
[49] X. Yan, J. Han, gSpan: graph-based substructure pattern mining, in: Proceedings of the 2002 IEEE International Conference on Data Mining, 2002, pp. 721–724, https://doi.org/10.1109/ICDM.2002.1184038.
[50] C. Borgelt, M.R. Berthold, Mining molecular fragments: finding relevant substructures of molecules, in: Proceedings of the 2002 IEEE International Conference on Data Mining, 2002, pp. 51–58, https://doi.org/10.1109/ICDM.2002.1183885.
[51] J. Huan, W. Wang, J. Prins, Efficient mining of frequent subgraphs in the presence of isomorphism, in: Proceedings of the Third IEEE International Conference on Data Mining, 2003, pp. 549–552, https://doi.org/10.1109/ICDM.2003.1250974.
[52] S. Nijssen, J.N. Kok, A quickstart in frequent structure mining can make a difference, in: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2004, pp. 647–652, https://doi.org/10.1145/1014052.1014134.
[53] J. Huan, W. Wang, J. Prins, J. Yang, SPIN: Mining maximal frequent subgraphs from graph databases, in: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2004, pp. 581–586, https://doi.org/10.1145/1014052.1014123.
[54] L.T. Thomas, S.R. Valluri, K. Karlapalem, MARGIN: Maximal Frequent Subgraph Mining, in: Proceedings of the Sixth International Conference on Data Mining, 2006, pp. 1097–1101, https://doi.org/10.1109/ICDM.2006.102.
[55] J.F. Guo, R. Chai, J. Li, Top-down algorithm for mining maximal frequent subgraph, Adv. Mater. Res. 204–210 (2011) 1472–1476, https://doi.org/10.4028/www.scientific.net/AMR.204-210.1472.
[56] M. Waskom, O. Botvinnik, P. Hobson, J. Warmenhoven, J.B. Cole, Y. Halchenko, J. Vanderplas, S. Hoyer, S. Villalba, E. Quintero, A. Miles, T. Augspurger, T. Yarkoni, C. Evans, D. Wehner, L. Rocher, T. Megies, L.P. Coelho, E. Ziegler, T. Hoppe, S. Seabold, S. Pascual, P. Cloud, M. Koskinen, C. Hausler, D. Milajevs Kjemmett, A. Qalieh, D. Allan, K. Meyer, Seaborn: v0.6.0, Zenodo, 2015, https://doi.org/10.5281/zenodo.19108.
[57] J.D. Hunter, Matplotlib: a 2D graphics environment, Comput. Sci. Eng. 9 (3) (2007) 90–95, https://doi.org/10.1109/MCSE.2007.55.
[58] A.A. Hagberg, D.A. Schult, P.J. Swart, Exploring network structure, dynamics, and function using NetworkX, in: Proceedings of the 7th Python in Science Conference, 2008, pp. 11–15, in: https://conference.scipy.org/proceedings/scipy2008/paper_2/.
[59] M. Wörlein, T. Meinl, I. Fischer, M. Philippsen, A quantitative comparison of the subgraph miners MoFa, gSpan, FFSM, and Gaston, in: Proceedings of the 9th European Conference on Principles of Data Mining and Knowledge Discovery, 2005, pp. 392–403, https://doi.org/10.1007/11564126_39.
[60] J. Han, J. Pei, Y. Yin, Mining frequent patterns without candidate generation, ACM SIGMOD Rec. 29 (2000) 1–12, https://doi.org/10.1145/335191.335372.